Proof-theoretic contributions to explicit mathematics

Habilitationsschrift

Thomas Strahm

Institut für Informatik und
angewandte Mathematik
Universität Bern

2001

Contents

Introduction

Proof theory came into being in the twenties of the last century, when it was inaugurated by David Hilbert in order to secure the foundations of mathematics. It was Hilbert's aim to overcome the foundational crisis, in which mathematics found itself at the beginning of the 20th century. This crisis had its origin in the new set-theoretic viewpoint, which was based on Cantor's "paradise" of infinite sets and which led to new non-constructive proof methods in mathematics. Hence, a foundational justification of these new principles was in order, and this need was confirmed by the discovery of various contradictions in the first formulations of set theory, most famously Russell's paradox.

It was the aim of *Hilbert's program* to (i) axiomatize the whole of mathematics in a "big" axiom system M and, subsequently, (ii) show the consistency of M in a small fragment F of M, which is based on finitistic principles only. As is well-known, Kurt Gödel [44] proved in 1930 that Hilbert's program must fail in the above form. His famous incompleteness results show, in particular, that in each consistent axiom system S containing a modificium of arithmetic, the consistency of S is not derivable in S itself.

Gödel's results did not at all destroy proof theory, but they showed that proof-theoretic research had to transgress the strict finitistic standpoint proposed by Hilbert. The crucial first step in this direction was made by Gerhard Gentzen [41] in 1936, who gave a completely new consistency proof for the system of Peano arithmetic PA. Gentzen's proof used, apart from finitistic methods, only induction along a wellordering of ordertype ε_0, the first fixed point of the ordinal function $\alpha \mapsto \omega^\alpha$. Significant aspects of Gentzen's proof are that (i) the wellordering of ordertype ε_0 can be coded as a decidable, even primitive recursive relation on the natural numbers, and moreover, (ii) induction along that ordering is only needed for decidable, quantifier-free statements. Thus, ε_0 characterizes the infinitary content of first-order number theory PA, and it also measures the initial segment of the true arithmetic sentences which are provable in PA.

Gentzen's treatment of first-order arithmetic constitutes the paradigmatic example of a so-called *ordinal analysis* of a formal system S, whose aim is to attach a specific informative recursive ordinal to S, the so-called *proof-theoretic ordinal of* S, in symbols $|\mathsf{S}|$. For example, $|\mathsf{S}|$ can be defined to be the order type of the least wellordering which is used to derive the consistency of S. Since Gentzen's treatment of PA, an ordinal analysis has been accomplished for a large number of formal systems, ranging from subsystems of second order arithmetic to various systems of admissible set theory, cf. Pohlers [93] and Rathjen [96] for recent surveys. Indeed, such analyses do not only give us precise information about the consistency strength of a given system S, but usually one obtains as a byproduct further results, such as e.g. an exact characterization of the computational power of S in terms of its provably total number-theoretic functions. Summing up, ordinal analysis can be seen as a contribution to what is sometimes called the *modified Hilbert program* nowadays.

A serious opponent of Hilbert was the Dutch mathematician L. E. J. Brouwer, who was a convinced critic of the set-theoretic viewpoint and of all non-constructive arguments in mathematics. According to his opinion, mathematics had to be developed on purely constructive grounds; in particular, Brouwer did not accept the law of excluded middle. Brouwer's *intuitionism* was later formalized by Heyting and led to the evolvement of intuitionistic formal systems. The development of computability theory in the thirties of the last century provided an adequate interpretation of intuitionism.

The study of formal systems for constructive mathematics and, more generally, constructive aspects of proofs has always been an important subject area for proof theory. It was particularly stimulated by the publication of Errett Bishop's book in 1967 on the constructive foundations of analysis, which initiated the development of a host of formal systems adequate for the representation of Bishop-style constructive mathematics (BCM); among them are Martin-Löf's theory of transfinite types [80, 81] and Feferman's systems of explicit mathematics [27, 29]. Bishop's understanding of constructive mathematics was in the sense of a high-level programming language: a constructive proof of an assertion gives immediate rise to an algorithm, which realizes the proof. Bishop's ideas substantially influenced work in computer science, more specifically research on the so-called proofs-as-programs paradigm.

Explicit mathematics was introduced by Feferman loc. cit. in the early seventies. Beyond its original aim to provide a basis for Bishop-style constructivism, the explicit framework has gained considerable importance in proof theory in connection with the proof-theoretic analysis of subsystems of second order arithmetic and set theory. In particular, it was possible to reduce prima-facie non-constructive systems to a constructively justifiable framework. The most famous example in this connection is the reduction of the subsystem of second order arithmetic based on Δ^1_2 comprehension and bar induction to the most prominent framework of explicit mathematics, T_0, achieved by Jäger [56] and Jäger and Pohlers [65]. Corresponding reductions in the framework of Martin-Löf type theory have been later obtained independently by Griffor and Rathjen [45] and Setzer [112]. Justifications of non-constructive in terms of constructive systems by means of proof-theoretic reductions can be seen as a further important step in the modified or extended Hilbert program.

More recently, systems of explicit mathematics have been used to study various forms of *abstract computation*, especially from a proof-theoretic perspective. It has turned out that already the operational core of explicit mathematics, so-called *applicative theories*, are of significant interest for this purpose. A lot of work in this connection has been devoted to the analysis of higher type functionals from generalized recursion theory, e.g. the non-constructive μ operator or the Suslin operator E_1, cf. Feferman and Jäger [35, 36], Glass and Strahm [43], Jäger and Strahm [68, 69, 70], Marzetta and Strahm [84], and Strahm [123]. For a comprehensive survey of many of these results cf. Jäger, Kahle, and Strahm [63]. We will be examining the abstract computations perspective of explicit mathematics in Part II of this habilitation thesis in the context of computational complexity, more precisely, time- and space bounded notions of computability.

Systems of explicit mathematics have also been employed to develop a general logical framework for functional programming and type theory, where it is possible to derive properties of functional programs such as termination and correctness. The programs considered are taken from functional programming languages, which are either based on the untyped λ calculus or the polymorphic typed λ calculus. For more details, cf. the references Feferman [31, 32, 34] and Jäger [60]. A discussion of the call-by-value and call-by-name point of view in the applicative setting of explicit mathematics is given in Stärk [116]. Moreover, for frameworks closer to actual programming languages, cf. Hayashi and Nakano [47] and Talcott [125]. The former reference contains, among other things, the description of an experimental implementation for extracting programs from constructive proofs in a Feferman-style explicit mathematics setting. Most recently, Studer [124] has shown how to use explicit mathematics as a logical foundation of object-oriented programming.

Let us now give a short explanation of the two basic kinds of objects which are present in explicit mathematics. These are *operations* or *rules* and *classifications* or *types*. The former may be thought of as mechanical rules of computation, which can freely be applied to each other: self-application is meaningful, though not necessarily total. The basic axioms concerning operations are those of a partial combinatory algebra, thus giving immediate rise to explicit definitions (lambda abstraction) and a form of the recursion theorem. The standard interpretation of the operations is the domain of the partial recursive functions. Classifications or types, on the other hand, are collections of operations and must be thought of as being generated successively from preceding ones. In contrast to the restricted character of operations, types can have quite complicated defining properties. What is essential in the whole explicit mathematics approach, however, is the fact that types are again represented by operations or, as we will call them in this case, *names*. Thus each type U is named or represented by a name u; in general, U may have many different names or representations. It is exactly this interplay between operations and types on the level of names which makes explicit mathematics extremely powerful and, in fact, witnesses its explicit character.

The main emphasis of our contributions in the first part of this habilitation thesis is on types and certain type existence principles, whereas in the second part of the thesis, we will work in the pure operational core of explicit mathematics. In both parts we will primarily be concerned with a thorough analysis of the proof-theoretic and computational content of the relevant systems. It is our aim in the sequel to give a very informal and general overview of this thesis. We try to avoid any technicalities at this stage and sketch the main results and ideas. Let us now start with Part I.

⋆ ⋆ ⋆

We begin with a very simple example of a type existence axiom in explicit mathematics. Given two types U and V with associated names u and v, respectively, we have the type U intersected with V, in symbols $U \cap V$, whose name is given *uniformly* by an operation int as $\mathsf{int}(u, v)$. Apart from axioms claiming the existence of certain ground types such as e.g. the identity type or the type of the natural numbers, there are further very simple type generating axioms, e.g. the one for forming complements of a type or the closure under quantification over operations. All these axioms have in common that they do not refer to the *general notion of type* or name in their defining condition: this is the reason why they are called *elementary*. In particular, quantification over types is not allowed in an elementary type existence principle. There is a further very natural principle of a rather different character, the so-called *Join* axiom for generating types. It applies to any type U with name u and an operation f for which fx is the name of a type V_x for each x in U, and produces (the name of) a new type $\mathsf{j}(u, f)$, the *disjoint union* of the types V_x for $x \in U$. The elementary type existence principles together with the join operation can be considered as the very basic type generating axioms of explicit mathematics.

The crucial notion in our considerations in Part I of this thesis is the one of a *universe*. Roughly speaking, a universe is a type of types which is closed under previously recognized type existence principles, the idea being that the acceptance of those principles leads to the acceptance of a corresponding universe. The proper way to understand the notion "types of types" in our explicit mathematics framework is in the form of "types of names (of types)", of course. The type existence principles which are relevant as closure conditions for our universes are those described above, namely the elementary closure properties of types together with the join principle.

Universes are a frequently studied concept in the constructive context at least since the work of Martin-Löf, cf. e.g. Martin-Löf [80, 81] or Palmgren [88] for a survey. Their justification is very closely related to reflection principles in classical and admissible set theory, leading to the existence of large cardinals and their recursive analogues, respectively. Universes were first discussed in the framework of explicit mathematics in Feferman [30] in connection with his proof of Hancock's conjecture. Later they have been reconsidered by Marzetta [83, 82] in the context of a non-uniform limit axiom, cf. also Marzetta and Strahm [84]. For a survey of some of the relevant previous results cf. Jäger, Kahle, and Studer [64]. Further references will be given in the course of this thesis.

Coming back to the notion of universe described above, i.e. types of types closed under the elementary type existence principles and join, the question arises: what are natural principles for generating such universes? In this thesis we will study two fundamental notions for generating universes, namely the limit axiom (L) and the Mahlo axiom (M). The axiom (L) is in a sense the simplest way for constructing universes: given a name a of a type, (L) claims the existence of a universe named $\mathsf{u}(a)$ such that a is an element of $\mathsf{u}(a)$. Here u is an operation which produces the name $\mathsf{u}(a)$ *uniformly* in a given name a. The Mahlo axiom (M), on the other hand, features a more sophisticated way for generating universes: given a name a and an operation f mapping (names of) types to (names of) types, there is a (name of) a universe $\mathsf{m}(a, f)$ which contains a and is closed under f; once more this universe is named by the operation m uniformly in a and f. In the following we are interested in the two systems of explicit mathematics EIN and EMA. Both are based on explicit elementary type theory with join plus the limit axiom (L) and the Mahlo axiom (M), respectively, as well as complete induction on the natural numbers with respect to types.

There is a strong connection between universes in explicit mathematics and regular cardinals in classical set theory as well as admissible ordinals in admissible set theory. On the basis of this correspondence, the universe of discourse of EIN clearly resembles the notion of an inaccessible cardinal in classical set theory or a recursively inaccessible ordinal in admissible set theory. Accordingly, EMA's universe of discourse is closely related to Mahlo cardinals in impredicative set theory and recursively Mahlo ordinals in admissible set theory. These analogies are further witnessed by the fact that the corresponding large cardinals and large admissible ordinals give immediate rise to models of EIN and EMA, which are based on classical and admissible set theory, respectively. Thus EIN and EMA provide further natural examples of Feferman's marriage of convenience between generalized recursion theory and classical set theory, cf. Feferman [28].

The main results in the first part of this habilitation thesis concern the determination of the exact proof-theoretic strength of EIN and EMA. In addition, we are also interested in the extension of EIN and EMA by the schema $(\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$ of complete induction on the natural numbers with respect to *all formulas* in the underlying language $\mathbb{L}$; recall that in EIN and EMA, induction on the natural numbers is available with respect to types only. We will characterize the proof-theoretic ordinal of all these systems by means of the so-called ternary Veblen of φ function, which is a straightforward generalization of the well-known binary φ function, cf. Section 2.1 for details. More precisely, we will establish the following proof-theoretic ordinals:

$$|\mathsf{EIN}| = \varphi 100 \qquad |\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})| = \varphi 1 \varepsilon_0 0$$
$$|\mathsf{EMA}| = \varphi \omega 00 \qquad |\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I_N})| = \varphi \varepsilon_0 00$$

The determination of the proof-theoretic strength of EIN is previously due to Kahle [74], cf. also Strahm [121]. The relevant argument used is a refinement of Marzetta's [82, 83] treatment of a *non-uniform* version of EIN, cf. also Marzetta and Strahm [84]. The first analysis of $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ is given in Strahm [121]. Finally, the proof theory of explicit metapredicative Mahloness EMA and $\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I_N})$ has been developed in Jäger and Strahm [66] (upper bounds) and Strahm [118] (lower bounds).

The ordinal $\varphi 100$ in fact equals the famous Feferman-Schütte ordinal Γ_0, the limiting number of predicative provability. It was found independently by Feferman and Schütte in the early sixties of the last century as a result of the foundational program to study the principles and ordinals which are implicit in a predicative conception of the universe of sets of natural numbers. Since then numerous theories have been found which are not prima facie predicatively justifiable, but nevertheless have predicative strength in the sense that Γ_0 is an upper bound to their proof-theoretic ordinal. It is common to all these predicative theories that their analysis requires methods from predicative proof theory only, in contrast to the present proof-theoretic treatment of stronger impredicative systems. On the other hand, it has long been known that there are natural systems which have proof-theoretic ordinal greater than Γ_0 and whose analysis makes use just as well of methods which every proof-theorist would consider to be predicative. Nevertheless, not many theories of the latter kind have been known until recently.

Metapredicativity is a new area in proof theory which is concerned with the analysis of formal systems whose proof-theoretic ordinal is beyond the Feferman-Schütte ordinal Γ_0, but which can be given a proof-theoretic analysis that uses *methods* from predicative proof theory only. It has recently been discovered that the world of metapredicativity is extremely rich and that it includes many natural and foundationally interesting formal systems, cf. e.g. Jäger [53], Jäger, Kahle, Setzer, and Strahm [62], Jäger and Strahm [66, 71], Kahle [73], Rathjen [97, 99], Rüede [106, 105, 103, 104], and Strahm [118, 121, 122]. A short discussion of this recent research work is given in Chapter 5 of this thesis.

The analysis of the systems of explicit mathematics EIN and EMA, possibly augmented by the schema of full induction on the natural numbers ($\mathbb{L}$-$\mathsf{I_N}$), clearly makes use of methods of predicative proof theory only. In particular, the wellordering proofs given in Chapter 2 of this thesis are predicative in spirit in the sense that they resemble and in fact generalize well-known wellordering proofs in systems of ramified analysis below Γ_0. Further, our upper bound computations are based on techniques from predicative proof theory such as asymmetric interpretation as well as generalized forms of the second cut elimination theorem of predicative proof theory. Thus, the results presented in the first part of this thesis substantially extend the realm of metapredicative proof theory.

Feferman's most famous system of explicit mathematics $\mathsf{T_0}$ also includes the so-called principle of *inductive generation*. It asserts the existence of (least definable) accessible parts of binary relations. It is well-known that the presence of the axiom of inductive generation greatly raises the proof-theoretic strength of a given system. For a detailed analysis of EIN and EMA augmented by inductive generation, the reader is referred to Jäger and Studer [72] and Tupailo [129]. For corresponding work in the context of Martin-Löf type theory, see Setzer [113] and Rathjen [98].

This concludes our introductory remarks on our work in the first part of this thesis. For a more detailed overview we refer the reader to the Plan of Part I as well as to the introductory texts at the beginning of each chapter.

$\star\star\star$

One of the central aims of theoretical computer science is to classify algorithms with respect to their computational complexity. Traditionally, the complexity of computing a function is measured by means of various kinds of time- and space bounded Turing- or register machines. Numerous important complexity classes have been identified, among them are the polynomial time computable functions, the functions computable on various levels of the polynomial time hierarchy, or the polynomial space computable functions; of crucial interest is the notion of polynomial time computability, which is the most widely studied mathematical model of feasible computability.

In the past years, intense research efforts have been made in order to find machine-independent characterizations of various classes of computational complexity by means of methods and concepts from mathematical logic. This led to new approaches to complexity in terms of finite model theory, subrecursion theory, lambda calculi, proof theory, and so on. The results obtained shed new light on the nature and structure of complexity classes and their mutual relationships.

In Part II of this habilitation thesis we focus on a proof-theoretic approach to computational complexity. In particular, we are interested in studying various questions related to bounded complexities in the expressively powerful *applicative core of explicit mathematics*, thus leading to generalized notions of complexity on abstract structures. These new investigations are hoped to provide a better understanding of bounded complexities from a proof-theoretic and abstract computability point of view.

The first-order applicative fragment of explicit mathematics is perfectly apt for studying and representing algorithms. Indeed, the axioms of a partial combinatory algebra guarantee the representability of all recursive functions, though in general the proof of totality or convergence in a given applicative setting heavily depends on the forms of induction allowed. Hence, it is natural to measure the strength of an applicative theory in terms of its provably total functions. A further distinguished advantage of this applicative approach is the fact that higher types arise very naturally and, hence, it also makes sense to consider the class of higher type functionals which provably converge in a specific axiomatic framework.

We will be mainly interested in this thesis in the classes of functions on the binary words $\mathbb{W} = \{0,1\}^\star$ which are computable on a multitape Turing machine in *polynomial time*, *simultaneously polynomial time and linear space*, *polynomial space*, and *linear space*. In the sequel we abbreviate these classes as FPtime, FPtimeLinspace, FPspace, and FLinspace, respectively. It is our aim to develop four applicative systems, one for each of the four complexity classes above. Our theories can be seen as natural applicative analogues of well-known systems of bounded arithmetic, cf. the monographs Buss [15], Hájek and Pudlák [46], and Krajíček [76].

All our systems are based on the theory B of operations and words. Its principal axioms are those of a partial combinatory algebra plus a unary predicate W for the binary words. Possibly, B is augmented by the (total) operations of word concatenation $*$ and word multiplication $\times$; here $x \times y$ denotes the word x, length of y times concatenated with itself. The crucial formulas in the language of B are the so-called $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formulas: those are formulas having a leading *bounded existential quantifier* followed by a positive and W free condition, see Section 7.2 for the exact definition. There are two forms of induction with respect to $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formulas which will be at the heart of our delineation of complexity classes. The first one is usual notation induction along the branches of the full binary tree; it is denoted by $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$. The second induction principle is simply induction along the lexicographic ordering of the full binary tree, which orders words by their length and words of the same length lexicographically; we call this principle with respect to $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formulas $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell})$.

Depending on whether we have $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ or $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell})$, and whether we assume as given only word concatenation or both concatenation and word multiplication, we can now distinguish four natural applicative theories. The theory B augmented by $*$, $\times$ and $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ is denoted by PT. If we replace in this system $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ by $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell})$, we use the name PS for the resulting theory. Moreover, the theories PTLS and LS are obtained from PT and PS by simply dropping the axioms about word multiplication $\times$. For a given applicative system S, let us denote by $\mathrm{ProvTot}(\mathsf{S})$ the class of its provably total word functions. In this thesis we will establish the following crucial results concerning the provably total functions of our four applicative systems, cf. our paper Strahm [117]:

$$\mathrm{ProvTot}(\mathsf{PT}) = \mathrm{FP\textsc{time}} \qquad \mathrm{ProvTot}(\mathsf{PTLS}) = \mathrm{FP\textsc{time}L\textsc{inspace}}$$

$$\mathrm{ProvTot}(\mathsf{PS}) = \mathrm{FP\textsc{space}} \qquad \mathrm{ProvTot}(\mathsf{LS}) = \mathrm{FL\textsc{inspace}}$$

In order to prove lower bounds for our applicative theories we will make use of suitable function algebra characterizations of the above complexity classes. The latter can be very pleasantly represented in the applicative framework, thereby making direct use of the recursion or fixed point theorem as well as our bounded induction principles. The corresponding upper bounds will be obtained by (partial) cut elimination in a Gentzen-style reformulation of our systems combined with a suitable realizability interpretation in their standard open term model.

Turning to the system PT whose provably total functions coincide with the polynomial time computable functions, we will see that in PT we can also embed well-known systems of bounded arithmetic such as e.g. Buss' S^1_2 (cf. Buss [15]) or, equivalently, Ferreira's system of polynomial time computable arithmetic PTCA^+ (cf. Ferreira [37, 38]). In contrast to the tedious "bootstrapping" of say S^1_2, which requires heavy coding machinery, the introduction of the polynomial time computable functions in our system PT is very smooth and coding-free.

As we have already indicated above, we will also be interested in higher type aspects of our applicative systems, in particular PT. The general background is the relatively new area of *higher type complexity theory* and, in particular, work on *feasible functionals*. The fundamental aim in this field is to identify a sensible higher type analogue of the polynomial time computable functions. Most prominent in the previous research is the class of so-called *basic feasible functionals* BFF, which has proved to be a very robust class with various kinds of machine and programming language characterizations.

The basic feasible functionals go back to Melhorn [85] and Cook and Urquhart [26]. In the latter paper they are introduced via a typed formal system PV^ω in order to establish functional and realizability interpretations of an intuitionistic version of Buss' S^1_2. The terms of PV^ω define exactly the BFF's. We will show that Cook and Urquhart's system PV^ω is directly contained in our applicative theory PT; thus, the basic feasible functionals are proof-theoretically justified in a type-free applicative setting.

Finally, we will present further natural applicative systems for various classes of computable functions. In particular, we will study a system PH which is closely related to the *polynomial time hierarchy*; the crucial principle of PH is a very uniform type two functional π for bounded quantification. Further investigations concern applicative theories whose provably total functions are exactly the primitive recursive functions.

For a more extensive overview of Part II of this habilitation thesis, we ask the reader to consult the Plan of Part II as well as the introductions to the various chapters.

⋆ ⋆ ⋆

Remarks

Part I and Part II of this thesis can be read entirely independently of each other. Moreover, the devoted reader who reads both parts of this thesis will realize that there is only a tiny overlap between the two parts with respect to the formulation and standard consequences of the basic applicative axioms of explicit mathematics.

Finally, let us mention that throughout these investigations we have made free use of the papers Strahm [121], Jäger and Strahm [66], Strahm [118], and Strahm [117].

Acknowledgements

I am deeply thankful to my academic teacher and friend Professor Gerhard Jäger. His steady support and his enthusiasm for logic and theoretical computer science have greatly influenced the genesis of this work and my personal scientific development.

I have benefited immensely from the close contact and collaboration with Professor Solomon Feferman. His numerous writings on explicit mathematics have always been very challenging. My two visits to Stanford in 1995 and 1998 were most enjoyable.

The frequent discussions with Professor Andrea Cantini on our common research interests have been very stimulating for my work. I particularly well remember the Castiglioncello workshop in 1998 and my short visit to Florence last spring.

Many thanks to all my friends from the Bern research group for lots of help and discussions and, especially, for the great atmosphere they have shared with me in all these years.

Last but not least, I would like to thank my wife Pascale for all her understanding and support and my parents for their steady encouragement over the years.

Bern, January 2001, Thomas Strahm

Part I

Metapredicative systems of explicit mathematics

Plan of Part I

In the following we give a short informal plan of the first part of this habilitation thesis.

In the first chapter we will define the basic theory of explicit mathematics ETJ. We will introduce the notion of *universe* and spell out in detail the fundamental universe generating axioms, i.e. the limit axiom (L) and the Mahlo axiom (M), resulting in the two extensions EIN and EMA of ETJ. We will briefly address some of the basic consequences of these systems.

Chapter 2 is devoted to detailed wellordering proofs for the systems EIN and EMA, possibly augmented by the schema of complete induction on the natural numbers $(\mathbb{L}\text{-}\mathsf{I_N})$ for arbitrary formulas in the underlying language $\mathbb{L}$. More precisely, we will establish the lower bounds Γ_0, $\varphi 1 \varepsilon_0 0$, $\varphi \omega 00$ and $\varphi \varepsilon_0 00$ for the systems of explicit mathematics EIN, $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$, EMA and $\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I_N})$ with φ denoting the ternary Veblen function. The methodology of our wellordering proofs is related to well-known wellordering proofs for predicative analysis.

In Chapter 3 we will introduce specific theories with ordinals over Peano arithmetic suitable for dealing with certain non-monotone inductive definitions. Central are the two theories OIN and OMA which describe a recursively inaccessible and recursively Mahlo universe of discourse. A crucial feature of our theories is the fact that induction on the ordinals is not permitted. The systems of explicit mathematics EIN and EMA will be embedded into OIN and OMA by means of formalized inner model constructions.

Upper bounds for OIN and OMA, possibly augmented by full induction on the natural numbers $(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$, will be established in Chapter 4 of this thesis, thus showing that the lower bounds obtained in Chapter 2 are indeed the best possible ones. In the treatment of our ordinal theories we will only use methods from predicative proof theory, thus showing the metapredicativity of our theories with ordinals and, hence, systems of explicit mathematics.

In Chapter 5, finally, we will provide a short and informal guided tour through the landscape of (metapredicative) systems which are closely related to EIN and EMA. We will address subsystems of second order arithmetic, admissible set theories without foundation, and (iterated) fixed point theories, among others.

Chapter 1

Systems of explicit mathematics with universes

In this chapter we will introduce the various systems of explicit mathematics which will be relevant in Part I of this habilitation thesis. We start by describing the basic theory of explicit mathematics ETJ, which is based on elementary comprehension and the axiom about join as well as induction on the natural numbers with respect to types; some of the crucial and well-known consequences of ETJ are mentioned without proof.

Next we will define the fundamental notion of a type being a *universe* in our explicit framework. Basically, a universe is a type of (names of) types which reflects the type generating principles of the theory ETJ. Two fundamental principles claiming the existence of universes will be spelled out, namely the *limit axiom* (L) and the *Mahlo axiom* (M), leading to the two extensions EIN and EMA of ETJ, respectively. EIN is called the theory for *explicit inaccessibility*, whereas EMA is our formal system for *explicit Mahloness*. Both systems contain, in addition, certain natural ordering principles for universes, one of whose basic consequences will be briefly addressed.

The central aim in Part I of this thesis is to give a complete ordinal analysis of EIN and EMA, possibly augmented by the *schema* of complete induction on the natural numbers with respect to all formulas in the underlying language of explicit mathematics.

1.1 Elementary explicit type theory with join

This section is devoted to the exact definition of the system ETJ[1] of explicit elementary type theory plus join as well as the discussion of some of its basic consequences.

All systems of explicit mathematics considered in Part I of this thesis are formulated in the second order language $\mathbb{L}$ for individuals and types. It comprises individual variables $a, b, c, f, g, h, u, v, w, x, y, z, \dots$ as well as type variables $U, V, W, X, Y, Z, \dots$ (both possibly with subscripts). $\mathbb{L}$ also includes the individual constants k, s (combinators), $\mathsf{p}, \mathsf{p_0}, \mathsf{p_1}$ (pairing and projections), 0 (zero), $\mathsf{s_N}$ (successor), $\mathsf{p_N}$ (predecessor), $\mathsf{d_N}$ (definition by numerical cases) and additional individual constants, called *generators*, which will be used for the uniform naming of types, namely nat (natural numbers), id (identity), co (complement), int (intersection), dom (domain), inv (inverse image), j (join), as well as u and m (universe generators). There is one binary function symbol $\cdot$ for (partial) application of individuals to individuals. Further, $\mathbb{L}$ has unary relation symbols $\downarrow$ (defined) and N (natural numbers) as well as three binary relation symbols $\in$ (membership), $=$ (equality) and $\Re$ (naming, representation).[2]

For a uniform definition of the notion of proof-theoretic ordinal (cf. Definition 4 below) it is convenient that $\mathbb{L}$ also includes an anonymous unary relation symbol Q and a corresponding generator q. The relation Q plays the role of an anonymous predicate on the natural numbers with no specific mathematical meaning.

The *individual terms* $(r, s, t, r_1, s_1, t_1, \dots)$ of $\mathbb{L}$ are built up from individual variables and individual constants by means of our function symbol $\cdot$ for application. In the following we often abbreviate $(s \cdot t)$ simply as (st), st or sometimes also $s(t)$; the context will always ensure that no confusion arises. We further adopt the convention of association to the left so that $s_1 s_2 \dots s_n$ stands for $(\dots(s_1 \cdot s_2)\dots s_n)$. We also set $t' := \mathsf{s_N} t$. Finally, we define general n tupling by induction on $n \geq 2$ as follows:

$$(s_1, s_2) := \mathsf{p} s_1 s_2, \qquad (s_1, \dots, s_{n+1}) := ((s_1, \dots, s_n), s_{n+1}).$$

[1] ETJ is basically the theory EETJ of Jäger, Kahle and Studer [64] plus induction on the natural numbers for types.

[2] The use of a binary naming relation $\Re$ for formalizing explicit mathematics is due to Jäger [59].

The positive literals of $\mathbb{L}$ are of the form $\mathsf{N}(s)$, $s\downarrow$, $s = t$, $U = V$, $s \in U$ and $\Re(s, U)$. Since we work with a logic of partial terms, it is not guaranteed that all terms have values, and $s\downarrow$ is read as s *is defined*. Moreover, $\mathsf{N}(s)$ says that s is a natural number, and the formula $\Re(s, U)$ is used to express that the individual s *represents* the type U or is a *name* of U.

The *formulas* $(A, B, C, A_1, B_1, C_1, \ldots)$ of $\mathbb{L}$ are generated from the positive literals by closing under the usual propositional connectives, as well as existential and universal quantification for individuals and types. The following table contains a useful list of abbreviations:

$$
\begin{array}{rcl}
s \simeq t & := & s\downarrow \vee\, t\downarrow \to s = t, \\
s \in \mathsf{N} & := & \mathsf{N}(s), \\
(\exists x \in \mathsf{N})A(x) & := & (\exists x)(x \in \mathsf{N} \wedge A(x)), \\
(\forall x \in \mathsf{N})A(x) & := & (\forall x)(x \in \mathsf{N} \to A(x)), \\
U \subset V & := & (\forall x)(x \in U \to x \in V), \\
s \,\dot{\in}\, t & := & (\exists X)(\Re(t, X) \wedge s \in X), \\
(\exists x \,\dot{\in}\, s)A(x) & := & (\exists x)(x \,\dot{\in}\, s \wedge A(x)), \\
(\forall x \,\dot{\in}\, s)A(x) & := & (\forall x)(x \,\dot{\in}\, s \to A(x)), \\
s \doteq t & := & (\exists X)[\Re(s, X) \wedge \Re(t, X)], \\
s \,\dot{\subset}\, t & := & (\exists X, Y)[\Re(s, X) \wedge \Re(t, Y) \wedge X \subset Y], \\
\Re(s) & := & (\exists X)\Re(s, X).
\end{array}
$$

The vector notation $\vec{U}$ and $\vec{s}$ is sometimes used to denote finite sequences of type variables $U_1, \ldots, U_m$ and individual terms $s_1, \ldots, s_n$, respectively, whose length is given by the context.

The logic of ETJ is Beeson's *classical logic of partial terms* (cf. Beeson [8] or Troelstra and Van Dalen [127]) for the individuals and *classical* logic with equality for the types. Observe that Beeson's formalization includes the usual strictness axioms.

The nonlogical axioms of the theory ETJ for elementary explicit types with join are divided into the following groups I–V:

I. Applicative axioms. These axioms formalize that the individuals form a partial combinatory algebra, that we have pairing and projection and the usual closure conditions on the natural numbers plus definition by numerical cases.

(1) $\mathsf{k}ab = a$,

(2) $\mathsf{s}ab\downarrow \;\wedge\; \mathsf{s}abc \simeq ac(bc)$,

(3) $\mathsf{p}_0(a,b) = a \;\wedge\; \mathsf{p}_1(a,b) = b$,

(4) $0 \in \mathsf{N} \;\wedge\; (\forall x \in \mathsf{N})(x' \in \mathsf{N})$,

(5) $(\forall x \in \mathsf{N})(x' \neq 0 \wedge \mathsf{p}_{\mathsf{N}}(x') = x)$,

(6) $(\forall x \in \mathsf{N})(x \neq 0 \;\rightarrow\; \mathsf{p}_{\mathsf{N}}x \in \mathsf{N} \wedge (\mathsf{p}_{\mathsf{N}}x)' = x)$,

(7) $a \in \mathsf{N} \wedge b \in \mathsf{N} \;\wedge\; a = b \rightarrow \mathsf{d}_{\mathsf{N}}xyab = x$,

(8) $a \in \mathsf{N} \wedge b \in \mathsf{N} \;\wedge\; a \neq b \rightarrow \mathsf{d}_{\mathsf{N}}xyab = y$.

II. Explicit representation and extensionality. The following axioms state that each type has a name, that there are no homonyms and that equality of types is extensional.

(1) $(\exists x)\Re(x, U)$,

(2) $\Re(a, U) \wedge \Re(a, V) \rightarrow U = V$,

(3) $(\forall x)(x \in U \leftrightarrow x \in U) \rightarrow U = V$.

III. Basic type existence axioms. In the following we provide a finite axiomatization of uniform elementary comprehension plus join.

Natural numbers

$\Re(\mathsf{nat}) \;\wedge\; (\forall x)(x \,\dot{\in}\, \mathsf{nat} \leftrightarrow \mathsf{N}(x))$.

Representation of Q

$\Re(\mathsf{q}) \;\wedge\; (\forall x)(x \,\dot{\in}\, \mathsf{q} \leftrightarrow \mathsf{Q}(x)) \;\wedge\; \mathsf{q} \,\dot{\subset}\, \mathsf{nat}$.

Identity

$\Re(\mathsf{id}) \;\wedge\; (\forall x)(x \,\dot{\in}\, \mathsf{id} \leftrightarrow (\exists y)(x = (y,y)))$.

Complements

$\Re(a) \;\rightarrow\; \Re(\mathsf{co}(a)) \wedge (\forall x)(x \,\dot{\in}\, \mathsf{co}(a) \leftrightarrow x \,\dot{\notin}\, a)$.

Intersections

$\Re(a) \wedge \Re(b) \;\rightarrow\; \Re(\mathsf{int}(a,b)) \wedge (\forall x)(x \,\dot{\in}\, \mathsf{int}(a,b) \leftrightarrow x \,\dot{\in}\, a \wedge x \,\dot{\in}\, b)$.

Domains

$\Re(a) \ \rightarrow \ \Re(\mathsf{dom}(a)) \wedge (\forall x)(x \mathrel{\dot{\in}} \mathsf{dom}(a) \leftrightarrow (\exists y)((x, y) \mathrel{\dot{\in}} a))$.

Inverse images

$\Re(a) \ \rightarrow \ \Re(\mathsf{inv}(a, f)) \wedge (\forall x)(x \mathrel{\dot{\in}} \mathsf{inv}(a, f) \leftrightarrow fx \mathrel{\dot{\in}} a)$.

Joins

$\Re(a) \wedge (\forall x \mathrel{\dot{\in}} a)\Re(fx) \ \rightarrow \ \Re(\mathsf{j}(a, f)) \wedge \Sigma(a, f, \mathsf{j}(a, f))$.

In this last axiom the formula $\Sigma(a, f, b)$ expresses that b names the disjoint union of f over a, i.e.

$$\Sigma(a, f, b) \ := \ (\forall x)(x \mathrel{\dot{\in}} b \leftrightarrow (\exists y, z)(x = (y, z) \wedge y \mathrel{\dot{\in}} a \wedge z \mathrel{\dot{\in}} fy)).$$

IV. Uniqueness of generators. These axioms essentially guarantee that different generators create different names. To achieve this, we have for syntactically different generators r_0 and r_1 and arbitrary generators s and t:

(1) $r_0 \neq r_1$,

(2) $(\forall x)(sx \neq t)$,

(3) $(\forall x, y)(sx = ty \ \rightarrow \ s = t \wedge x = y)$.

V. Type induction on the natural numbers. This axiom provides complete induction on the natural numbers for types.

$$(\mathsf{T}\text{-}\mathsf{I}_\mathsf{N}) \quad (\forall X)(0 \in X \wedge (\forall x \in \mathsf{N})(x \in X \rightarrow x' \in X) \ \rightarrow \ (\forall x \in \mathsf{N})(x \in X)).$$

This completes the description of our basic explicit framework ETJ for elementary type theory with join. We will also be interested in the strengthening of ETJ (and its extensions) by induction on the natural numbers for all formulas of the language $\mathbb{L}$. Accordingly, formula induction ($\mathbb{L}$-I_N) is the schema

$$(\mathbb{L}\text{-}\mathsf{I}_\mathsf{N}) \qquad A(0) \wedge (\forall x \in \mathsf{N})(A(x) \rightarrow A(x')) \ \rightarrow \ (\forall x \in \mathsf{N})A(x)$$

for each $\mathbb{L}$ formula A. We will write $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I}_\mathsf{N})$ for ETJ augmented by the schema $(\mathbb{L}\text{-}\mathsf{I}_\mathsf{N})$.

We conclude this section with mentioning some crucial consequences of ETJ. As usual, the axioms of a partial combinatory algebra allow one to define λ abstraction and to prove a recursion or fixed point theorem. For proofs of these standard results the reader is referred to [8, 27].

Lemma 1 (Abstraction and recursion) *1. For each $\mathbb{L}$ term t and all variables x there exists an $\mathbb{L}$ term $(\lambda x.t)$ whose variables are those of t, excluding x, so that* ETJ *proves*

$$(\lambda x.t)\downarrow \wedge (\lambda x.t)x \simeq t.$$

2. There exists a closed $\mathbb{L}$ term rec *so that* ETJ *proves*

$$\mathsf{rec} f\downarrow \wedge \mathsf{rec} f x \simeq f(\mathsf{rec} f)x.$$

In the original formulation of explicit mathematics, elementary comprehension is not dealt with by a finite axiomatization but directly as an infinite axiom scheme. An $\mathbb{L}$ formula is called *elementary* if it contains neither the relation symbol $\Re$ nor bound type variables. The following result of Feferman and Jäger [36] shows that this scheme of uniform elementary comprehension is provable from our finite axiomatization. Join and uniqueness of generators are not needed for this argument.

Lemma 2 (Elementary comprehension) *For every elementary formula $A(u, \vec{v}, W_1, \ldots, W_n)$ with at most the indicated free variables there exists a closed term t of $\mathbb{L}$ so that* ETJ *proves:*

1. $\bigwedge_{i=1}^{n} \Re(w_i, W_i) \rightarrow \Re(t(\vec{v}, w_1, \ldots, w_n))$,
2. $\bigwedge_{i=1}^{n} \Re(w_i, W_i) \rightarrow (\forall x)(x \,\dot{\in}\, t(\vec{v}, w_1, \ldots, w_n) \leftrightarrow A(x, \vec{v}, W_1, \ldots, W_n))$.

1.2 Introducing universes: the limit and Mahlo axioms

Important types in explicit mathematics are the so-called *universes*. In this section we introduce two important axioms for generating such universes, namely the limit and the Mahlo axioms, resulting in the crucial extensions EIN and EMA of ETJ, respectively. In addition, we briefly address some natural ordering principles for universes.

First, we want to introduce the concept of a *universe* into explicit mathematics. In a nutshell, a universe is supposed to be a type which consists of names only and reflects the type existence principles of the theory ETJ.

For the detailed definition of a universe we introduce some auxiliary notation and let $\mathcal{C}(W, a)$ be the closure condition which is the disjunction of the following $\mathbb{L}$ formulas:

(1) $a = \mathsf{nat} \vee a = \mathsf{q} \vee a = \mathsf{id}$,

(2) $(\exists x)(a = \mathsf{co}(x) \wedge x \in W)$,

(3) $(\exists x, y)(a = \mathsf{int}(x, y) \wedge x \in W \wedge y \in W)$,

(4) $(\exists x)(a = \mathsf{dom}(x) \wedge x \in W)$,

(5) $(\exists x, f)(a = \mathsf{inv}(x, f) \wedge x \in W)$,

(6) $(\exists x, f)[a = \mathsf{j}(x, f) \wedge x \in W \wedge (\forall y \mathrel{\dot\in} x)(fy \in W)]$.

Thus the formula $(\forall x)(\mathcal{C}(W, x) \to x \in W)$ states that W is a type which is closed under the type constructions of ETJ, i.e., elementary comprehension and join. If, in addition, all elements of W are names, we call W a universe, in symbols, $\mathsf{U}(W)$. Moreover, we write $\mathcal{U}(a)$ to express that the individual a is the name of a universe.

$$\begin{aligned} \mathsf{U}(W) &:= (\forall x)(\mathcal{C}(W, x) \to x \in W) \wedge (\forall x \in W)\Re(x), \\ \mathcal{U}(a) &:= (\exists X)(\Re(a, X) \wedge \mathsf{U}(X)). \end{aligned}$$

We are now ready to state the crucial universe existence principles studied in this thesis, namely the limit and Mahlo axioms, leading to two systems of explicit mathematics EIN and EMA, respectively.

A very simple way to claim the existence of universes is by adding an operation u which assigns to each name a a universe $\mathsf{u}(a)$ containing a. This is easily expressed by the following limit axiom (L):

$$(\mathsf{L}) \qquad \Re(a) \to \mathcal{U}(\mathsf{u}(a)) \wedge a \mathrel{\dot\in} \mathsf{u}(a).$$

This axiom can be seen as an explicit analogue of the corresponding limit axiom well-known from theories for iterated admissible sets, cf. Jäger [58].

A more sophisticated and stronger form of generating universes is given by the so-called Mahlo axioms in explicit mathematics. For their formulation the following shorthand notations are useful.

$$\begin{aligned} (f : \Re \to \Re) &:= (\forall x)(\Re(x) \to \Re(fx)), \\ (f : s \to s) &:= (\forall x \mathrel{\dot\in} s)(fx \mathrel{\dot\in} s). \end{aligned}$$

Obviously, $(f : \Re \to \Re)$ and $(f : s \to s)$ means that f maps names to names and elements of (the type named by) s to elements of (the type named by) s, respectively.

We can now introduce the Mahlo axiom for explicit mathematics. Given a name a and an operation f from names to names one simply claims that there exists (a name of) a universe $\mathsf{m}(a,f)$ which contains a and reflects f. Taking up the analogy that regular cardinals in classical set theory correspond to universes in explicit mathematics, our formulation of Mahlo in explicit mathematics may be regarded as a uniform version of Mahlo in set theory.

$$(\mathsf{M}.1) \qquad \Re(a) \land (f : \Re \to \Re) \;\to\; \mathcal{U}(\mathsf{m}(a,f)) \land a \,\dot{\in}\, \mathsf{m}(a,f),$$

$$(\mathsf{M}.2) \qquad \Re(a) \land (f : \Re \to \Re) \;\to\; (f : \mathsf{m}(a,f) \to \mathsf{m}(a,f)).$$

It is readily clear that the universe generator u can be defined in terms of m by means of the operation $\lambda a.\mathsf{m}(a, \lambda x.x)$. In fact, we will see in the next chapter that m even allows for much stronger “iterations” of u.

It is an interesting topic to see what kind of ordering principles for universes can be consistently added to the previous axioms. This question is discussed at full length in Jäger, Kahle and Studer [64], and it is shown there that one must not be too liberal. As a consequence of these considerations we do not claim linearity and connectivity for arbitrary universes, but only for so-called *normal universes*, i.e. universes which are named by means of the type generator u and m,

$$\mathcal{U}_{no}(a) \;:=\; \mathcal{U}(a) \;\land\; [(\exists x)(a = \mathsf{u}(x)) \;\lor\; (\exists x, f)(a = \mathsf{m}(x,f))].$$

The principles of *linearity* and *connectivity* of normal universes are then given by the following two axioms:

$$(\mathcal{U}_{no}\text{-}\mathsf{Lin}) \qquad (\forall x,y)[\mathcal{U}_{no}(x) \land \mathcal{U}_{no}(y) \;\to\; x \,\dot{\in}\, y \lor x \doteq y \lor y \,\dot{\in}\, x],$$

$$(\mathcal{U}_{no}\text{-}\mathsf{Con}) \qquad (\forall x,y)[\mathcal{U}_{no}(x) \land \mathcal{U}_{no}(y) \;\to\; x \,\dot{\subset}\, y \lor y \,\dot{\subset}\, x].$$

It is shown in [64] that connectivity of normal universes also implies transitivity of normal universes in its most general form. For the reader's convenience we briefly sketch the relevant argument.

Lemma 3 (Strong transitivity) *We have that* $\mathsf{ETJ} + (\mathcal{U}_{\mathsf{m}}\text{-}\mathsf{Con})$ *proves*

$$\mathcal{U}_{no}(a) \land \mathcal{U}_{no}(b) \land c \doteq a \land c \,\dot{\in}\, b \;\to\; a \,\dot{\subset}\, b.$$

Proof. Assume the premise of the implication to be proved. Then c is also a name of the universe named by a. Since universes never contain their names (cf. e.g. Marzetta [82]) we have $c \mathrel{\dot{\notin}} a$, thus $b \mathrel{\dot{\not\subset}} a$. But now connectivity of normal universes ($\mathcal{U}_{no}$-Con) yields $a \mathrel{\dot{\subset}} b$ as desired. □

We are now in a position to define the two central systems of explicit mathematics EIN and EMA for *explicit inaccessibility* and *explicit Mahloness*, respectively. According to our discussion above, EIN is contained in EMA.

$$\begin{aligned}
\mathsf{EIN} &:= \mathsf{ETJ} + (\mathsf{L}) + (\mathcal{U}_{no}\text{-}\mathsf{Lin}) + (\mathcal{U}_{no}\text{-}\mathsf{Con}), \\
\mathsf{EMA} &:= \mathsf{ETJ} + (\mathsf{M.1}) + (\mathsf{M.2}) + (\mathcal{U}_{no}\text{-}\mathsf{Lin}) + (\mathcal{U}_{no}\text{-}\mathsf{Con}).
\end{aligned}$$

In the sequel we will be interested in a detailed proof-theoretic analysis of EIN and EMA, possibly augmented by the schema $(\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$ of complete induction on the natural numbers N with respect to all formulas in $\mathbb{L}$. In particular, we will establish the following proof-theoretic ordinals:

$$\begin{array}{ll}
|\mathsf{EIN}| = \Gamma_0 & |\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})| = \varphi 1 \varepsilon_0 0 \\
|\mathsf{EMA}| = \varphi \omega 0 0 & |\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})| = \varphi \varepsilon_0 0 0
\end{array}$$

We will also briefly address the (well-known) proof theory of ETJ as well as $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$. Special emphasis will be put in the following chapters on very *uniform* proofs of lower as well as upper proof-theoretic bounds.

Chapter 2

Wellordering proofs

In this chapter we will provide detailed wellordering proofs for the systems introduced in Chapter 1, cf. Strahm [118]. We will start with some ordinal-theoretic preliminaries including, in particular, a detailed definition of the ternary Veblen or φ function $\varphi\alpha\beta\gamma$.

In a first step we will briefly review the well-known lower bounds for the two systems ETJ as well as $\mathsf{ETJ}+(\mathbb{L}\text{-}\mathsf{I_N})$. Then we will turn to the system EIN for explicit inaccessibility and show that in the presence of type induction, transfinite induction is available along initial segments of the Feferman-Schütte ordinal Γ_0; this result is previously due to Feferman [30]. The extension of EIN by the full induction schema $(\mathbb{L}\text{-}\mathsf{I_N})$ signifies a dramatic increase in proof-theoretic strength. In particular, $(\mathbb{L}\text{-}\mathsf{I_N})$ allows for the build up of hierarchies of universes of length bounded by a fixed ordinal α less than ε_0. As a consequence, this will yield that transfinite induction is derivable in $\mathsf{EIN}+(\mathbb{L}\text{-}\mathsf{I_N})$ for initial segments of the ordinal $\varphi 1\varepsilon_0 0$.

The system EMA for explicit Mahloness features, in addition, the existence of hierarchies of hyperuniverses, 2-hyperuniverses and so on. For example, a hyperuniverse is a universe which is closed under the universe generating operation u. We will show that the ordinal $\varphi\omega 00$ is a lower bound to the proof-theoretic strength of the theory EMA. Further, the stronger system $\mathsf{EMA}+(\mathbb{L}\text{-}\mathsf{I_N})$ derives the existence of α-hyperuniverses for each α less than ε_0 and, therefore, it proves induction along initial segments of the ordinal $\varphi\varepsilon_0 00$.

The term *metapredicative* indeed applies to the wellordering proofs for EIN and EMA given in this chapter. First of all, the *notation system* used is based on the ternary φ function, which is a straightforward generalization of the well-known binary φ function; in particular, no collapsing is used in this notation system. Secondly and most importantly, the general *methodology* of the wellordering proofs given below is very much in the spirit of the wellordering proofs for predicative systems due to Feferman and Schütte, cf. e.g. [30, 33, 110]. For example, instead of working in initial segments of the ramified analytic hierarchy or the ordinary jump hierarchy one considers hierarchies of universes, n-hyperuniverses, or α-hyperuniverses.

2.1 Ordinal-theoretic preliminaries

In the following we will measure the proof-theoretic strength of formal theories in terms of their proof-theoretic ordinals. As usual, for all primitive recursive relations $\sqsubset$[1] and all $\mathbb{L}$ formulas $A(a)$ we set:

$$\begin{aligned} \mathsf{Prog}(\sqsubset, A) &:= (\forall x \in \mathsf{N})[(\forall y \in \mathsf{N})(y \sqsubset x \rightarrow A(y)) \rightarrow A(x)], \\ \mathsf{TI}(\sqsubset, A) &:= \mathsf{Prog}(\sqsubset, A) \rightarrow (\forall x \in \mathsf{N})A(x). \end{aligned}$$

Thus $\mathsf{TI}(\sqsubset, A)$ expresses transfinite induction along the relation $\sqsubset$ for the formula $A(a)$. The proof-theoretic ordinal of a theory T is defined by referring to transfinite induction for the anonymous relation Q.

Definition 4 *1. An ordinal* α is provable in a theory T*, if there is a primitive recursive wellordering* $\sqsubset$ *of order type* α *so that* $\mathsf{T} \vdash \mathsf{TI}(\sqsubset, \mathsf{Q})$*.*

2. The least ordinal which is not provable in T *is called the* proof-theoretic ordinal of T *and is denoted by* $|\mathsf{T}|$*.*

The ordinals which are relevant for the theories considered in this thesis are most easily expressed by making use of a *ternary* Veblen or φ function which we are going to define now. The usual Veblen hierarchy generated by the *binary* function φ, starting off with the function $\varphi 0 \beta = \omega^\beta$ is well known from the literature, cf. Pohlers [92] or Schütte [110]. The *ternary* φ function is obtained as a straightforward generalization of the binary case by defining $\varphi\alpha\beta\gamma$ inductively as follows:

[1] Of course, by making use of the recursion theorem and a little amount of complete induction on the natural numbers one can easily represent primitive recursive functions and relations in ETJ.

(i) $\varphi 0\beta\gamma$ is just $\varphi\beta\gamma$;

(ii) if $\alpha > 0$, then $\varphi\alpha 0\gamma$ denotes the γth ordinal which is strongly critical with respect to all functions $\lambda\xi,\eta.\varphi\delta\xi\eta$ for $\delta < \alpha$.

(iii) if $\alpha > 0$ and $\beta > 0$, then $\varphi\alpha\beta\gamma$ denotes the γth common fixed point of the functions $\lambda\xi.\varphi\alpha\delta\xi$ for $\delta < \beta$.

For example, $\varphi 10\alpha$ is Γ_α, and more generally, $\varphi 1\alpha\beta$ denotes a Veblen hierarchy over $\lambda\alpha.\Gamma_\alpha$. It is straightforward how to extend these ideas in order to obtain φ functions of all finite arities, and even further to Schütte's Klammersymbole [109].

We let Λ_3 denote the least ordinal greater than 0 which is closed under the ternary φ function. In the following we confine ourselves to the standard notation system which is based on this function. Since the exact definition of such a system is a straightforward generalization of the notation system for Γ_0 (cf. [92, 110]), we do not go into details here. We write $\prec$ for the corresponding primitive recursive wellordering of order type Λ_3 and assume without loss of generality that 0 is the least element with respect to $\prec$. Further, we let Lim denote the primitive recursive set of limit notations and we presuppose a primitive recursively given fundamental sequence $(\ell[n] : n \in \mathbb{N})$ for each limit notation ℓ; we will assume that $\ell[0] > 0$. As the definition of fundamental sequences is easy in the setting of φ functions we do not give it here and refer the reader to the relevant proofs in the next sections.

There exist primitive recursive functions acting on the codes of our notation system which correspond to the usual operations on ordinals. In the sequel it is often convenient in order to simplify notation to use ordinals and ordinal operations instead of their codes and primitive recursive analogues. Then (for example) ω and $\omega + \omega$ stand for the natural numbers whose order type with respect to $\prec$ are ω and $\omega + \omega$.

When working in the systems ETJ*,* EIN *and* EMA *in this chapter, we let* $a, b, c, d, e, \ldots$ *range over the field of* $\prec$ *and* ℓ *denote limit notations.* In addition, we write $\mathsf{Prog}(A)$ instead of $\mathsf{Prog}(\prec, A)$ and $\mathsf{TI}(A, a)$ for the formula $\mathsf{Prog}(A) \to (\forall b \prec a)A(b)$. Further, $\mathsf{I}(a)$ abbreviates $(\forall X)\mathsf{TI}(X, a)$. If we want to stress the relevant induction variable of a formula A, we sometimes write $\mathsf{Prog}(\lambda x.A(x))$ and $\mathsf{TI}(\lambda x.A(x), a)$ instead of $\mathsf{Prog}(A)$ and $\mathsf{TI}(A, a)$, respectively. Finally, we let $\mathsf{Prog}(U)$ and $\mathsf{Prog}(u)$ stand for $\mathsf{Prog}(\lambda x.x \in U)$

and $\mathsf{Prog}(\lambda x.x \mathrel{\dot\in} u)$, respectively; the formulas $\mathsf{TI}(U, a)$ and $\mathsf{TI}(u, a)$ are understood analogously.

2.2 Lower bounds for ETJ

The proof theory of ETJ and $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I}_\mathsf{N})$ is well-known and due to Feferman [29]. Nevertheless, in favor of a uniform presentation, let us briefly recapitulate the relevant results and ideas.

Of course, it is trivial to observe that Peano arithmetic PA is contained in ETJ. Indeed, it is even possible to get a direct embedding of the subsystem ACA_0 of second order arithmetic based on arithmetic comprehension and induction on the natural numbers for sets; this embedding uses the obvious translation of the language of second order arithmetic into the language $\mathbb{L}$ of types and names. Thus we have the following theorem, which can also be obtained by carrying through the usual wellordering proof directly in ETJ.

Theorem 5 *We have for all ordinals α less than ε_0 that ETJ proves $\mathsf{I}(\alpha)$. Thus, $\varepsilon_0 \leq |\mathsf{ETJ}|$.*

The crucial observation to be made in establishing the lower bound of ETJ augmented by $(\mathbb{L}\text{-}\mathsf{I}_\mathsf{N})$ is that in $\mathsf{ETJ}+(\mathbb{L}\text{-}\mathsf{I}_\mathsf{N})$ we can prove that the arithmetical jump hierarchy iterated along each initial segment of ε_0 is well-defined. To see this, assume that we are given an arithmetical formula $A(X, u)$. Then we can use the recursion theorem in order to find an operation jump_A satisfying the following recursion equations:

$$\begin{aligned}
\mathsf{jump}_A x0 &\simeq x, \\
\mathsf{jump}_A x(a{+}1) &\simeq \{u \in \mathsf{N} : A(\mathsf{jump}_A xa, u)\}, \\
\mathsf{jump}_A x\ell &\simeq \mathsf{j}(\{a : a \prec \ell\}, \mathsf{jump}_A x).
\end{aligned}$$

The crucial task is to show the well-definedness of this A jump hierarchy below ε_0, i.e., one must derive for each fixed α less that ε_0 the statement

$$(\star) \qquad \Re(x) \rightarrow (\forall a \prec \alpha)\Re(\mathsf{jump}_A xa).$$

It is readily seen that $(\star)$ can be proved by transfinite induction along initial segments of ε_0, which is available in $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I}_\mathsf{N})$ for arbitrary formulas of $\mathbb{L}$, due to the presence of $(\mathbb{L}\text{-}\mathsf{I}_\mathsf{N})$. Hence, the standard subsystem of analysis

Π^1_0-$\mathsf{CA}_{<\varepsilon_0}$ (cf. e.g. [35]) is contained in $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})$. Thus we have e.g. by Feferman [33] or Schütte [110] that $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})$ proves transfinite induction with respect to types along each initial segment of the ordinal $\varphi\varepsilon_0 0$.

Theorem 6 *We have for all ordinals α less than $\varphi\varepsilon_0 0$ that* $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})$ *proves* $\mathsf{I}(\alpha)$. *Thus,* $\varphi\varepsilon_0 0 \leq |\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})|$.

We will see in Chapter 4 that the lower bounds sketched in this section are indeed best possible.

2.3 Lower bounds for EIN

In this section we establish proof-theoretic lower bounds for the two systems EIN and $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$. More specifically, we show that EIN and $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ prove $\mathsf{I}(\alpha)$ for all ordinals α less than Γ_0 and $\varphi 1 \varepsilon_0 0$, respectively.

Let us start with the system EIN which includes the limit axiom (L) and induction on the natural numbers for types only. It has been shown by Marzetta [83, 82] that (a non-uniform version of) EIN interprets Friedman's subsystem of second order arithmetic ATR_0, which is based on the schema of arithmetical transfinite recursion and induction on the natural numbers for sets, cf. [115]. Using the well-known fact that $|\mathsf{ATR}_0| = \Gamma_0$ (cf. [40, 57]), Marzetta's embedding yields the desired lower bound for EIN, of course.

Nevertheless, in the following let us briefly sketch a direct wellordering proof for EIN, which will also be relevant in the wellordering proofs for the stronger systems treated below. The following lemma constitutes the crucial step in deriving transfinite induction below Γ_0.

Lemma 7 *We have that* EIN *proves*

$$\Re(x) \wedge (\forall y \,\dot{\in}\, \mathsf{u}(\mathsf{u}(x)))\mathsf{TI}(y,a) \;\rightarrow\; (\forall y \,\dot{\in}\, \mathsf{u}(x))\mathsf{TI}(y,\varphi a 0).$$

Proof. (Sketch) Let us indicate the key steps of this argument: given a name x and assuming $(\forall y \,\dot{\in}\, \mathsf{u}(\mathsf{u}(x)))\mathsf{TI}(y,a)$, we also have $(\forall y \,\dot{\in}\, \mathsf{u}(\mathsf{u}(x)))\mathsf{TI}(y,\omega^{a+1})$, due to the fact that universes are closed under elementary (and hence arithmetical) comprehension. Further, given an arbitrary name y in $\mathsf{u}(x)$ we can now set up the ordinary (arithmetical) jump hierarchy starting with y below ω^{a+1} in $\mathsf{u}(x)$; this hierarchy can be described by making use of the recursion theorem and using join at limit stages, cf. the previous section. The fact that the hierarchy is total or well-defined in $\mathsf{u}(x)$ is shown by induction up to ω^{a+1} and indeed this is possible since the relevant statement to be established, expressing that the levels of the hierarchy belong to $\mathsf{u}(x)$, defines a type in $\mathsf{u}(\mathsf{u}(x))$, a universe above $\mathsf{u}(x)$, by closure of $\mathsf{u}(\mathsf{u}(x))$ under elementary comprehension. But the existence of the jump hierarchy starting from y below ω^{a+1} immediately entails $\mathsf{TI}(y, \varphi a0)$, for example by Lemma 5.3.1 in Feferman [33] or Lemma 10 on p. 187 in Schütte [110]. $\Box$

A straightforward iterated application of the above lemma yields the following crucial theorem about the proof-theoretic lower bound of the theory EIN.

Theorem 8 *We have for all ordinals α less than Γ_0 that EIN proves $\mathsf{I}(\alpha)$. Thus, $\Gamma_0 \leq |\mathsf{EIN}|$.*

Proof. We inductively define the fundamental sequence $(\alpha_j : j \in \mathbb{N})$ for Γ_0 by $\alpha_0 := 1$ and $\alpha_{j+1} := \varphi\alpha_j 0$. We further use the notation $\mathsf{u}^{(j)}(x)$ for the j-fold application of u to x, i.e. $\mathsf{u}^{(0)}(x) := x$ and $\mathsf{u}^{(j+1)}(x) := \mathsf{u}(\mathsf{u}^{(j)}(x))$. We have to show that EIN proves $(\forall X)\mathsf{TI}(X, \alpha_k)$ for each natural number k. Towards that aim one makes straightforward use of the previous lemma in order to show by induction on $j \leq k$ that EIN proves

$$\Re(x) \ \rightarrow \ (\forall y \,\dot{\in}\, \mathsf{u}^{(k+1-j)}(x))\mathsf{TI}(y, \alpha_j).$$

If we choose $j = k$ in this last assertion, then we obtain that EIN derives

$$\Re(x) \ \rightarrow \ (\forall y \,\dot{\in}\, \mathsf{u}(x))\mathsf{TI}(y, \alpha_k).$$

In particular, this entails that $(\forall x)(\Re(x) \rightarrow \mathsf{TI}(x, \alpha_k))$ is provable in EIN. Since we have an axiom saying that each type has a name, we have thus shown in EIN the assertion $(\forall X)\mathsf{TI}(X, \alpha_k)$. $\Box$

This ends our discussion of the lower bound of EIN. In conclusion, EIN allows for the build up of *finite* towers of universes, but nothing essentially stronger. Using the techniques sketched above, it is also not difficult to establish sharp lower bounds for the systems ETJ and ETJ plus $(\mathbb{L}\text{-}\mathsf{I_N})$ augmented by an axiom claiming the existence of exactly n universes on top of each other. For details see Feferman [30] and Marzetta and Strahm [84].

We now turn to the wellordering proof of the system $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$. The crucial difference in strength between EIN and $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ consists in the fact that the presence of ($\mathbb{L}$-$\mathsf{I_N}$) allows us to build *transfinite* hierarchies of universes of length α for each fixed α less than ε_0.

Towards that aim, we introduce an operation h via the recursion theorem in order to satisfy the following recursion equations:

$$\begin{aligned} \mathsf{h}x0 &\simeq \mathsf{u}(x), \\ \mathsf{h}x(a{+}1) &\simeq \mathsf{u}(\mathsf{h}xa), \\ \mathsf{h}x\ell &\simeq \mathsf{u}(\mathsf{j}(\{a : a \prec \ell\}, \mathsf{h}x)). \end{aligned}$$

Hence, the hierarchy starts with a universe containing (the name) x, at successor stages one puts a universe on top of the hierarchy defined so far, and at limit stages a universe above the disjoint union of the previously defined hierarchy is taken. The well-definedness of h below ε_0 is asserted in the following lemma.

Lemma 9 *We have for all ordinals α less than ε_0 that* $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ *proves*

1. $(\forall x)[\Re(x) \rightarrow (\forall a \prec \alpha)\mathcal{U}(\mathsf{h}xa)]$,
2. $(\forall x)[\Re(x) \rightarrow (\forall a \prec \alpha)(\forall b \prec a)(\mathsf{h}xb \mathrel{\dot\in} \mathsf{h}xa)]$.

Proof. For the proof of this lemma it is crucial to observe that we have transfinite induction up to each α less than ε_0 available in $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ with respect to *arbitrary* statements of $\mathbb{L}$. This is due to the presence of the full schema of induction on the natural numbers. Hence, both claims can be proved by transfinite induction up to an $\alpha \prec \varepsilon_0$. For the first assertion this is immediate. For the second claim one has to show that the statement $(\forall b \prec a)(\mathsf{h}xb \mathrel{\dot\in} \mathsf{h}xa)$ is progressive with respect to $\prec$. This is straightforward if a is not a limit notation. If a is limit and $b \prec a$, then also $b{+}1 \prec a$ and $\mathsf{h}xb \mathrel{\dot\in} \mathsf{h}x(b{+}1)$. On the other hand, one easily sees that there is a name of the universe denoted by $\mathsf{h}x(b{+}1)$ which belongs to $\mathsf{h}xa$, since we have by definition $\mathsf{j}(\{b : b \prec a\}, \mathsf{h}x) \mathrel{\dot\in} \mathsf{h}xa$. But then $\mathsf{h}xb \mathrel{\dot\in} \mathsf{h}xa$ is immediate by strong transitivity (Lemma 3). This concludes our argument. □

Crucial for the wellordering proof below is the notion $\mathsf{I}^c_x(a)$ of *transfinite induction up to a for all types (respectively names) belonging to a universe* $\mathsf{h}xb$ *for* $b \prec c$, which is given as follows:

$$\mathsf{I}^c_x(a) \;:=\; (\forall b \prec c)(\forall y \mathrel{\dot\in} \mathsf{h}xb)\mathsf{TI}(y, a).$$

The next lemma tells us that $\mathsf{I}^\ell_x(a)$ can be represented by a type in $\mathsf{h}x\ell$.

Lemma 10 *We have for all ordinals α less than ε_0 that* $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ *proves*

$$(\forall x, \ell)[\Re(x) \wedge \ell \prec \alpha \;\rightarrow\; (\exists y \mathrel{\dot\in} \mathsf{h}x\ell)(\forall a)(a \mathrel{\dot\in} y \leftrightarrow \mathsf{I}^{\ell}_{x}(a))].$$

Proof. We work in $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ and assume $\Re(x)$ as well as $\ell \prec \alpha \prec \varepsilon_0$. Then we know by the definition of $\mathsf{h}x\ell$ that $\mathsf{j}(\{b : b \prec \ell\}, \mathsf{h}x) \mathrel{\dot\in} \mathsf{h}x\ell$. By closure of $\mathsf{h}x\ell$ under join this readily entails that also (a name of) the type

$$\{(b, u, v) : b \prec \ell \wedge u \mathrel{\dot\in} \mathsf{h}xb \wedge v \mathrel{\dot\in} u\}$$

belongs to $\mathsf{h}x\ell$. Therefore, by closure of $\mathsf{h}x\ell$ under elementary comprehension, there exists a type (name) y in $\mathsf{h}x\ell$ which satisfies the condition claimed by the lemma. □

The following lemma is crucial for the base case of Lemma 12 below and makes important use of Lemma 7 above.

Lemma 11 *We have for all ordinals α less than ε_0 that* $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ *proves*

$$(\forall x, \ell)[\Re(x) \wedge \ell \prec \alpha \;\rightarrow\; \mathsf{Prog}(\lambda a.\mathsf{I}^{\ell}_{x}(\Gamma_a))].$$

Proof. Assuming that x is a name and $\ell \prec \alpha$ is a limit notation, we aim at showing that $\lambda a.\mathsf{I}^{\ell}_{x}(\Gamma_a)$ is progressive. This claim is immediate by an easy inductive argument from

$$(\forall b)[\mathsf{I}^{\ell}_{x}(b) \;\rightarrow\; \mathsf{I}^{\ell}_{x}(\varphi b0)]. \tag{1}$$

Towards a proof of (1) assume $\mathsf{I}^{\ell}_{x}(b)$ and fix a $c \prec \ell$. We have to show $(\forall y \mathrel{\dot\in} \mathsf{h}xc)\mathsf{TI}(y, \varphi b0)$. Since ℓ is limit we also have $c{+}1 \prec \ell$ and, hence, our assumption yields $(\forall y \mathrel{\dot\in} \mathsf{h}x(c{+}1))\mathsf{TI}(y, b)$. Further, since $\mathsf{h}x(c{+}1) = \mathsf{u}(\mathsf{h}xc)$ and $\mathsf{h}xc = \mathsf{u}(w)$ for a suitable name w, we are now in a position to apply Lemma 7 and obtain $(\forall y \mathrel{\dot\in} \mathsf{h}xc)\mathsf{TI}(y, \varphi b0)$. Since c was an arbitrary notation less than ℓ we thus have shown $\mathsf{I}^{\ell}_{x}(\varphi b0)$. This ends our proof of (1).

Now in order to establish $\mathsf{Prog}(\lambda a.\mathsf{I}^{\ell}_{x}(\Gamma_a)$, it is clearly enough to show the following claims (2)–(4):

$$\mathsf{I}^{\ell}_{x}(\Gamma_0), \tag{2}$$

$$\mathsf{I}^{\ell}_{x}(\Gamma_a) \;\rightarrow\; \mathsf{I}^{\ell}_{x}(\Gamma_{a+1}), \tag{3}$$

$$\mathsf{Lim}(a) \wedge (\forall a' \prec a)\mathsf{I}^{\ell}_{x}(\Gamma_{a'}) \;\rightarrow\; \mathsf{I}^{\ell}_{x}(\Gamma_a). \tag{4}$$

For (2), observe that we are given a fundamental sequence $z_v = \Gamma_0[v]$ for Γ_0, where $z_0 = 1$ and $z_{v+1} = \varphi z_v 0$. Hence, (2) follows from (1) by ordinary induction. The argument for (3) is completely analogous by using the fundamental sequence $z_v = \Gamma_{b+1}[v]$ for Γ_{b+1} with $z_0 = \Gamma_b + 1$ and $z_{v+1} = \varphi z_v 0$. Finally, for (4) just observe that if $\mathsf{Lim}(a)$, then Γ_a is the supremum over $a' \prec a$ of $\Gamma_{a'}$, so that the claim is immediate in this case. All together this completes the proof of our lemma. □

An important tool in the proof of Lemma 12 below is the formula $\mathsf{Main}_\alpha(a)$. It is the natural adaptation to our setting of similar formulas employed in a wellordering proof below Γ_0 in Feferman [33] and the metapredicative wellordering proof in Jäger, Kahle, Setzer and Strahm [62]. Its definition makes use of the binary relation $\uparrow$ on the field of $\prec$,

$$a \uparrow b \ := \ (\exists c, \ell)(b = c + a \cdot \ell).$$

Here of course $+$ and $\cdot$ are the primitive recursive operations corresponding to ordinal addition and multiplication on the field of $\prec$. The formula $\mathsf{Main}_\alpha(a)$ now has the following definition:

$$\mathsf{Main}_\alpha(a) \ := \ (\forall x, b, c)[\Re(x) \wedge c \preceq \alpha \wedge \omega^{1+a} \uparrow c \wedge \mathsf{I}_x^c(b) \ \rightarrow \ \mathsf{I}_x^c(\varphi 1 a b)].$$

We are now ready to turn to the crucial lemma concerning $\mathsf{Main}_\alpha(a)$. It corresponds to Main Lemma I in [62], formulated in the framework of explicit mathematics with universes.

Lemma 12 *We have for all ordinals α less than ε_0 that* $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I}_\mathsf{N})$ *proves the statement* $\mathsf{Prog}(\lambda a.\mathsf{Main}_\alpha(a))$.

Proof. In order to show $\mathsf{Prog}(\lambda a.\mathsf{Main}_\alpha(a))$ it is enough to verify the following claims (1)–(3):

(1) $\mathsf{Main}_\alpha(0)$,

(2) $\mathsf{Main}_\alpha(a) \ \rightarrow \ \mathsf{Main}_\alpha(a{+}1)$,

(3) $\mathsf{Lim}(a) \wedge (\forall w \in \mathsf{N})\mathsf{Main}_\alpha(a[w]) \ \rightarrow \ \mathsf{Main}_\alpha(a)$.

Towards a proof of (1) we fix a name x and assume that $c = c_0 + \omega \cdot \ell$ for a limit notation ℓ. We have to show

(4) $\mathsf{I}_x^{c_0+\omega\cdot\ell}(b) \ \rightarrow \ \mathsf{I}_x^{c_0+\omega\cdot\ell}(\Gamma_b)$.

Towards this aim, assume $\mathsf{I}_x^{c_0+\omega\cdot\ell}(b)$. It is sufficient to verify $\mathsf{I}_x^{c_0+\omega\cdot\ell[u]}(\Gamma_b)$ for each natural number u. Since $\ell[u] > 0$ we have that $c_0 + \omega \cdot \ell[u]$ is always limit, and hence we obtain by Lemma 11,

$$(5) \qquad \mathsf{Prog}(\lambda a.\mathsf{I}_x^{c_0+\omega\cdot\ell[u]}(\Gamma_a))$$

for each u in N. But we have by Lemma 10 that $\{a : \mathsf{I}_x^{c_0+\omega\cdot\ell[u]}(a)\}$ forms a type in the universe $\mathsf{h}x(c_0 + \omega \cdot \ell[u])$, so that $\mathsf{I}_x^{c_0+\omega\cdot\ell}(b)$ and (5) immediately imply $\mathsf{I}_x^{c_0+\omega\cdot\ell[u]}(\Gamma_b)$ for each natural number u. This finishes the proof of (4) and, hence, assertion (1) of our lemma has been verified.

For the proof of (2), let us assume $\mathsf{Main}_\alpha(a)$, i.e.

$$(6) \qquad (\forall x, d, e)[\Re(x) \wedge d \preceq \alpha \wedge \omega^{1+a} \uparrow d \wedge \mathsf{I}_x^d(e) \rightarrow \mathsf{I}_x^d(\varphi 1ae)].$$

We want to establish $\mathsf{Main}_\alpha(a{+}1)$. So fix a name x and assume

$$(7) \qquad c \preceq \alpha \wedge c = c_0 + \omega^{1+a+1} \cdot \ell \wedge \mathsf{I}_x^c(b).$$

We have to show $\mathsf{I}_x^c(\varphi 1(a{+}1)b)$. Again it is sufficient to establish

$$(8) \qquad \mathsf{I}_x^{c_0+\omega^{1+a+1}\cdot\ell[u]}(\varphi 1(a{+}1)b)$$

for each natural number u. We set $c[u] := c_0 + \omega^{1+a+1} \cdot \ell[u]$ and readily observe that $\omega^{1+a} \uparrow c[u]$ for each u in N. Hence, we can derive from (6)

$$(9) \qquad (\forall u \in \mathsf{N})(\forall e)[\mathsf{I}_x^{c[u]}(e) \rightarrow \mathsf{I}_x^{c[u]}(\varphi 1ae)].$$

Our next immediate task is to establish the assertion

$$(10) \qquad (\forall u \in \mathsf{N})\mathsf{Prog}(\lambda e.\mathsf{I}_x^{c[u]}(\varphi 1(a{+}1)e)).$$

The proof of (10) requires the verification of the three claims

$$(11) \qquad \mathsf{I}_x^{c[u]}(\varphi 1(a{+}1)0),$$

$$(12) \qquad \mathsf{I}_x^{c[u]}(\varphi 1(a{+}1)e) \rightarrow \mathsf{I}_x^{c[u]}(\varphi 1(a{+}1)(e{+}1)),$$

$$(13) \qquad \mathsf{Lim}(e) \wedge (\forall e' \prec e)\mathsf{I}_x^{c[u]}(\varphi 1(a{+}1)e') \rightarrow \mathsf{I}_x^{c[u]}(\varphi 1(a{+}1)e).$$

In order to verify (11), observe that we are given a fundamental sequence $z_w = \varphi 1(a{+}1)0[w]$ for $\varphi 1(a{+}1)0$, where $z_0 = 1$ and $z_{w+1} = \varphi 1az_w$. Hence, (11) follows from (9) by ordinary induction. As to (12), we have a fundamental sequence $z_w = \varphi 1(a{+}1)(e{+}1)[u]$ for $\varphi 1(a{+}1)(e{+}1)$ with $z_0 = \varphi 1(a{+}1)e{+}1$

and $z_{w+1} = \varphi 1 a z_w$. Again the claim follows from (9) by ordinary induction. Finally, for (13), we observe that if $\mathsf{Lim}(e)$, then $\varphi 1(a+1)e$ is the supremum over $e' \prec e$ of $\varphi 1(a{+}1)e'$, hence there is nothing to prove in this case. Thus we have finished the verification of (10). Since $\mathsf{I}_x^{c[u]}(\varphi 1(a{+}1)e)$ can be represented by a type in the universe $\mathsf{h}x(c[u])$ for each u (Lemma 10), and we know $\mathsf{I}_x^c(b)$ by (7), we are now in a position to conclude from (10),

$$(\forall u \in \mathsf{N})\mathsf{I}_x^{c[u]}(\varphi 1(a{+}1)b).$$

But this is exactly (8) and, hence, the verification of (2) is concluded.

Let us finally turn to a proof of assertion (3). For a given name x we assume

$$(14) \qquad \mathsf{Lim}(a) \,\wedge\, (\forall w \in \mathsf{N})\mathsf{Main}_\alpha(a[w]),$$

$$(15) \qquad c \preceq \alpha \,\wedge\, c = c_0 + \omega^{1+a} \cdot \ell \,\wedge\, \mathsf{I}_x^c(b).$$

Observe that we have $\omega^{1+a} = \omega^a$ since a is limit. We have to show $\mathsf{I}_x^c(\varphi 1 a b)$. Indeed, it is enough to establish

$$(16) \qquad \mathsf{I}_x^{c_0+\omega^a\cdot\ell[u]}(\varphi 1 a b)$$

for each natural number u. If we set $c[u] := c_0 + \omega^a \cdot \ell[u]$, then one readily sees that for each w in N, we can write $c[u]$ in the form

$$(17) \qquad c[u] = c_0 + \omega^{1+a[w]} \cdot \ell_w$$

for a suitable limit notation ℓ_w depending on w (and $\ell[u]$). Hence, we have that $\omega^{1+a[w]} \uparrow c[u]$ for each natural number w and, therefore, we can derive from (14) that

$$(18) \qquad (\forall u, w \in \mathsf{N})(\forall e)[\mathsf{I}_x^{c[u]}(e) \;\rightarrow\; \mathsf{I}_x^{c[u]}(\varphi 1 a[w] e)].$$

In a further step we now want to establish the assertion

$$(19) \qquad (\forall u \in \mathsf{N})\mathsf{Prog}(\lambda e.\mathsf{I}_x^{c[u]}(\varphi 1 a e)).$$

Again this breaks into three subcases (20)–(22), namely

$$(20) \qquad \mathsf{I}_x^{c[u]}(\varphi 1 a 0),$$

$$(21) \qquad \mathsf{I}_x^{c[u]}(\varphi 1 a e) \;\rightarrow\; \mathsf{I}_x^{c[u]}(\varphi 1 a(e{+}1)),$$

$$(22) \qquad \mathsf{Lim}(e) \wedge (\forall e' \prec e)\mathsf{I}_x^{c[u]}(\varphi 1 a e') \;\rightarrow\; \mathsf{I}_x^{c[u]}(\varphi 1 a e).$$

As to (20), we have a fundamental sequence $z_w = \varphi 1a0[w]$ for $\varphi 1a0$ so that $z_0 = 1$ and $z_{w+1} = \varphi 1a[w]z_w$, hence, (20) is an immediate consequence of (18). The proof of (21) runs similarly. Finally, (22) is straightforward as above. All together we have concluded the verification of (19). Again, we can represent $\mathsf{I}_x^{c[u]}(\varphi 1ae)$ by a type of level $c[u]$, so that we can now derive from (19) and our assumption $\mathsf{I}_x^c(b)$ in (15),

$$(\forall u \in \mathsf{N})\mathsf{I}_x^{c[u]}(\varphi 1ab).$$

This is literally (16) and, hence, we are done with (3). In fact, this also finishes the proof of our crucial lemma. □

Using the main lemma above we are now in a position to derive the following main theorem of this section, thus establishing the desired lower bound for $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$.

Theorem 13 *We have for all ordinals α less than $\varphi 1\varepsilon_0 0$ that* $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ *proves* $\mathsf{I}(\alpha)$. *Thus,* $\varphi 1\varepsilon_0 0 \leq |\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})|$.

Proof. It is enough to show that $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ proves $(\forall X)\mathsf{TI}(X, \varphi 1\alpha 0)$ for each α less than ε_0. For that purpose, fix an arbitrary $\alpha < \varepsilon_0$. Then we also have $\omega^{1+\alpha} \cdot \omega < \varepsilon_0$ and, hence, we have $\mathsf{Prog}(\lambda a.\mathsf{Main}_{\omega^{1+\alpha}\cdot\omega}(a))$ as a theorem of $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ by our main lemma above. Since transfinite induction below ε_0 is available in $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ with respect to arbitrary statements of $\mathbb{L}$, we obtain that $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ proves $\mathsf{Main}_{\omega^{1+\alpha}\cdot\omega}(\alpha)$, i.e. the statement

$$(\forall x, b, c)[\Re(x) \wedge c \preceq \omega^{1+\alpha} \cdot \omega \wedge \omega^{1+\alpha} \uparrow c \wedge \mathsf{I}_x^c(b) \rightarrow \mathsf{I}_x^c(\varphi 1\alpha b)].$$

By choosing c as $\omega^{1+\alpha} \cdot \omega$ and b as 0 in this assertion, one derives the following as a theorem of $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$:

$$(\forall x)[\Re(x) \rightarrow \mathsf{I}_x^{\omega^{1+\alpha}\cdot\omega}(\varphi 1\alpha 0)].$$

But now we can readily conclude that $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ proves $(\forall X)\mathsf{TI}(X, \varphi 1\alpha 0)$. This is as desired and concludes the proof of the theorem. □

2.4 Lower bounds for EMA

In this section we turn to the wellordering proof for the system EMA and show that $\varphi\omega 00 \leq |\mathsf{EMA}|$; we also briefly indicate the lower bound $\varphi\varepsilon_0 00$ of $\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I_N})$.

The wellordering proofs for EIN and $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ given in all detail in the previous section have been presented in such a way that they easily generalize to provide the wellordering proof for EMA outlined in this section. As a consequence, large parts of the proofs of lemmas and theorems in this section are simply relativized versions of similar proofs given above, so that in many cases we can confine ourselves to simply formulating the relevant results and refer to the previous section for detailed proofs.

The wellordering proof for EMA can be most perspicuously presented by making use of certain subsystems S_n of EMA. The crucial type existence axiom of S_n claims the existence of *n-hyperuniverses*, which can be seen as an analogue of n-(hyper)inaccessible sets. We will see that the existence of n-hyperuniverses for each natural number n is an immediate consequence of the Mahlo axioms (M.1) and (M.2). Moreover, the proof-theoretic strength of EMA is already exhausted by its subsystems S_n for each $n \in \mathbb{N}$.

For the formulation of S_n we augment our language $\mathbb{L}$ by a generator constant u_n for each natural number n. Below we define the notion of a *type W being an n-hyperuniverse*, n-$\mathsf{U}(W)$; accordingly, n-$\mathcal{U}(u)$ expresses that u is the name of an n-hyperuniverse,

$$\begin{aligned}
0\text{-}\mathsf{U}(W) &:= \mathsf{U}(W), \\
(n{+}1)\text{-}\mathsf{U}(W) &:= \mathsf{U}(W) \wedge (\forall x \in W)(\mathsf{u}_n(x) \in W), \\
n\text{-}\mathcal{U}(u) &:= (\exists X)(\Re(u, X) \wedge n\text{-}\mathsf{U}(X)).
\end{aligned}$$

The defining axiom for the constant u_n claims for each name x that $\mathsf{u}_n(x)$ is the name of an n-hyperuniverse containing x,

$$\Re(x) \;\rightarrow\; n\text{-}\mathcal{U}(\mathsf{u}_n(x)) \wedge x \mathrel{\dot{\in}} \mathsf{u}_n(x).$$

The theory S_n now extends elementary explicit type theory with join ETJ by (i) the defining axioms for the constants u_m ($m \leq n$), as well as (ii) linearity and connectivity axioms for universes which are normal with respect to the generators u_m ($m \leq n$). Hence, if we identify u_0 with u, then the system S_0 is simply the theory EIN based on the limit axiom (L) and type induction on the natural numbers ($\mathsf{T}\text{-}\mathsf{I_N}$).

We observe that due to the presence of the linearity and connectivity axioms for normal universes in S_n, we also have *strong transitivity* for such universes according to (the proof of) Lemma 3 above.

Lemma 14 (n-hyperuniverses in EMA) *We have for all natural numbers n that S_n is contained in EMA.*

Proof. The type generators u_n can be defined in EMA by means of m,

$$\mathsf{u}_0 \;=\; \lambda x.\mathsf{m}(x, \lambda y.y), \qquad \mathsf{u}_{n+1} \;=\; \lambda x.\mathsf{m}(x, \mathsf{u}_n).$$

One readily shows by induction on n and by making use of the Mahlo axioms ($\mathsf{M}.1$) and ($\mathsf{M}.2$) that the so-defined u_n's satisfy their defining axioms in S_n. In the case $n = 0$ we have that $\mathsf{u}_0(x)$ is a universe containing x for each name x, since trivially $(\lambda y.y)$ is a total operation from $\Re$ to $\Re$. For the induction step we assume that the defining axiom for u_n has been derived in EMA; in particular, this yields that $\mathsf{u}_n : \Re \to \Re$ and, hence, by the Mahlo axioms we have for each name x that (i) $\mathsf{u}_{n+1}(x)$ is a universe containing x and (ii) $\mathsf{u}_{n+1}(x)$ is closed under u_n, thus showing that indeed $(n{+}1)$-$\mathcal{U}(\mathsf{u}_{n+1}(x))$. This concludes our inductive argument.

Further, the linearity and connectivity axioms ($\mathcal{U}_\mathsf{m}$-Lin) and ($\mathcal{U}_\mathsf{m}$-Con) of EMA entail the corresponding axioms of S_n. We have established that S_n is a subsystem of EMA for each natural number n. □

In sequel we will establish that S_n proves $\mathsf{I}(\alpha)$ for each ordinal α less than $\varphi(n{+}1)00$. This shows in particular that $\varphi\omega00$ is a lower bound for the proof-theoretic ordinal of EMA. The key lemma to be proved in the sequel says that if x is a name and we know that transfinite induction holds below a with respect to all types (names) in $\mathsf{u}_n(\mathsf{u}_n(x))$ (i.e. a universe containing a universe that contains x), then transfinite induction holds even below $\varphi na0$ for all types (names) in $\mathsf{u}_n(x)$.

Main Lemma 15 *We have for all natural numbers n that S_n proves*

$$(\star) \qquad \Re(x) \wedge (\forall y \,\dot{\in}\, \mathsf{u}_n(\mathsf{u}_n(x)))\mathsf{TI}(y, a) \;\to\; (\forall y \,\dot{\in}\, \mathsf{u}_n(x))\mathsf{TI}(y, \varphi na0).$$

The proof of this main lemma is by (meta) induction on n. The assertion of the lemma in case $n = 0$ literally coincides with our previous Lemma 7 for EIN and, indeed, our main lemma is just the natural generalization of Lemma 7.

Let us now turn to the induction step. For that purpose we fix a natural number n and assume that $(\star)$ is true for n, aiming at a proof of the assertion of our main lemma for $n{+}1$. I.e. we want to show in the theory S_{n+1} that for all names x,

$$(\star\star) \qquad (\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x)))\mathsf{TI}(y,a) \;\rightarrow\; (\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(x))\mathsf{TI}(y,\varphi(n{+}1)a0).$$

A crucial ingredient in the proof of $(\star\star)$ are (uniform) *transfinite hierarchies of n-hyperuniverses* within an $(n{+}1)$-hyperuniverse; these are the natural generalizations of the h hierarchy discussed in the previous section. In particular, using the recursion theorem, we let h_n be a closed term of $\mathbb{L}$ so that we have provably in ETJ:

$$\begin{aligned} \mathsf{h}_n x0 &\simeq \mathsf{u}_n(x), \\ \mathsf{h}_n x(a{+}1) &\simeq \mathsf{u}_n(\mathsf{h}_n xa), \\ \mathsf{h}_n x\ell &\simeq \mathsf{u}_n(\mathsf{j}(\{a : a \prec \ell\}, \mathsf{h}_n x)). \end{aligned}$$

Of course, in general, one needs some amount of transfinite induction in order to show that h_n is well-defined in an $(n{+}1)$-hyperuniverse y. Therefore, in order to express the well-definedness of h_n below a in y, we let $\mathsf{Hier}_n(y,a)$ denote the conjunction of the following three formulas:

(i) $(\forall x \,\dot{\in}\, y)(\forall b \prec a)(\mathsf{h}_n xb \,\dot{\in}\, y)$,

(ii) $(\forall x \,\dot{\in}\, y)(\forall b \prec a)n\text{-}\mathcal{U}(\mathsf{h}_n xb)$,

(iii) $(\forall x \,\dot{\in}\, y)(\forall b \prec a)(\forall c \prec b)(\mathsf{h}_n xc \,\dot{\in}\, \mathsf{h}_n xb)$.

The following lemma is a generalization of Lemma 9 above. It expresses that h_n is well-defined below a in an $(n{+}1)$-hyperuniverse $\mathsf{u}_{n+1}(x)$ provided that transfinite induction below a is available with respect to all types (names) in $\mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x))$.

Lemma 16 *We have that S_{n+1} proves*

$$\Re(x) \wedge (\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x))\mathsf{TI}(y,a) \;\rightarrow\; \mathsf{Hier}_n(\mathsf{u}_{n+1}(x),a).$$

Proof. Reasoning in S_{n+1} we assume that x is a name and for all types (names) y in $\mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x))$ transfinite induction is available below a. We have to show $\mathsf{Hier}_n(\mathsf{u}_{n+1}(x),a)$, i.e. for all $z \,\dot{\in}\, \mathsf{u}_{n+1}(x)$,

$$(1) \qquad (\forall b \prec a)(\mathsf{h}_n zb \,\dot{\in}\, \mathsf{u}_{n+1}(x)),$$

$$(2) \qquad (\forall b \prec a)n\text{-}\mathcal{U}(\mathsf{h}_n zb),$$

$$(3) \qquad (\forall b \prec a)(\forall c \prec b)(\mathsf{h}_n zc \,\dot{\in}\, \mathsf{h}_n zb).$$

Since $\{b \prec a : \mathsf{h}_n zb \,\dot{\in}\, \mathsf{u}_{n+1}(x)\}$ defines a type in $\mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x))$ by elementary comprehension, (1) follows by a straightforward transfinite induction. Moreover, (2) is immediate from (1) by the definition of h_n, the fact that universes consist of names only, and the defining axioms for the u_n's.

As to (3), we first observe that $\{b \prec a : (\forall c \prec b)(\mathsf{h}_n zc \,\dot{\in}\, \mathsf{h}_n zb)\}$ defines a type in $\mathsf{u}_{n+1}(x)$ (and hence in $\mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x))$ by transitivity): to see this one basically applies join to (1) and subsequently uses an obvious instance of elementary comprehension. Given our general assumption, we can now derive (3) by an inductive argument in literally the same manner as in the proof of Lemma 9. □

The formulas $n\mathsf{I}^c_x(a)$, expressing *transfinite induction up to a for all types (respectively names) belonging to a n-hyperuniverse* $\mathsf{h}_n xb$ *for* $b \prec c$, are the straightforward generalizations of the formula $\mathsf{I}^c_x(a)$ used above.

$$n\mathsf{I}^c_x(a) \;:=\; (\forall b \prec c)(\forall y \,\dot{\in}\, \mathsf{h}_n xb)\mathsf{TI}(y, a).$$

According to the next lemma, we have that $n\mathsf{I}^\ell_x(a)$ forms a type in $\mathsf{h}_n x\ell$. This is the generalization of Lemma 10 and the proof is literally the same.

Lemma 17 *We have that* S_{n+1} *proves*

$$\Re(x) \wedge \mathsf{Hier}_n(\mathsf{u}_{n+1}(x), a) \;\rightarrow$$
$$(\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(x))(\forall \ell \prec a)(\exists z \,\dot{\in}\, \mathsf{h}_n y\ell)(\forall b)[b \,\dot{\in}\, z \leftrightarrow n\mathsf{I}^\ell_y(b)].$$

The following lemma makes crucial use of our general induction hypothesis, i.e. the claim $(\star)$ of our Main Lemma 15 for n. It directly generalizes Lemma 11 of the previous section.

Lemma 18 *We have that* S_{n+1} *proves*

$$\Re(x) \wedge \mathsf{Hier}_n(\mathsf{u}_{n+1}(x), a) \;\rightarrow\; (\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(x))(\forall \ell \prec a)\mathsf{Prog}(\lambda b.n\mathsf{I}^\ell_y(\varphi(n{+}1)0b)).$$

Proof. Presupposing that x is a name and $\mathsf{Hier}_n(\mathsf{u}_{n+1}(x), a)$, we want to establish that $(\lambda b.n\mathsf{I}^\ell_y(\varphi(n{+}1)0b))$ is progressive for arbitrary $y \,\dot{\in}\, \mathsf{u}_{n+1}(x)$ and limit notations $\ell \prec a$. In literally the same way as in the proof of Lemma 11, this claim is immediate from

$$(1) \qquad (\forall c)[n\mathsf{I}^\ell_y(c) \;\rightarrow\; n\mathsf{I}^\ell_y(\varphi nc0)].$$

Aiming at a proof of (1) we assume $n\mathsf{I}_y^\ell(c)$ and fix a $d \prec \ell$. We have to derive $(\forall z \mathrel{\dot\in} \mathsf{h}_n yd)\mathsf{TI}(z, \varphi nc0)$. Since ℓ is limit we have $d{+}1 \prec \ell$ and, hence, our assumption forces $(\forall z \mathrel{\dot\in} \mathsf{h}_n y(d{+}1))\mathsf{TI}(z, c)$. In addition, since we have $\mathsf{h}_n y(d{+}1) = \mathsf{u}_n(\mathsf{h}_n yd)$ and $\mathsf{h}_n yd = \mathsf{u}_n(w)$ for some name w in $\mathsf{u}_{n+1}(x)$, we are now able to apply our general assumption $(\star)$ for n and obtain

$$(\forall z \mathrel{\dot\in} \mathsf{h}_n yd)\mathsf{TI}(z, \varphi nc0).$$

As $d \prec \ell$ was chosen arbitrarily, we have indeed shown $n\mathsf{I}_y^\ell(\varphi nc0)$. This concludes our verification of (1) and, hence, the proof of this lemma. □

We are now in a position to turn to the natural generalization (and relativization) of the formulas $\mathsf{Main}_\alpha(a)$. Accordingly, we define the formulas $n\mathsf{Main}_a^x(b)$ in the following manner:

$$n\mathsf{Main}_a^x(b) \;:=\; (\forall y \mathrel{\dot\in} x)(\forall c, d)[d \preceq a \,\wedge\, \omega^{1+b} \uparrow d \,\wedge\, n\mathsf{I}_y^d(c) \;\rightarrow\; n\mathsf{I}_y^d(\varphi(n{+}1)bc)]$$

Given a name x and assuming $\mathsf{Hier}_n(\mathsf{u}_{n+1}(x), a)$, the following lemma says that the formula $n\mathsf{Main}_a^{\mathsf{u}_{n+1}(x)}(b)$ defines a type in the universe $\mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x))$. The proof of the lemma is straightforward and very similar in spirit to the proof of Lemma 17 and Lemma 10.

Lemma 19 *We have that* S_{n+1} *proves*

$$\Re(x) \wedge \mathsf{Hier}_n(\mathsf{u}_{n+1}(x), a) \;\rightarrow$$
$$(\exists y \mathrel{\dot\in} \mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x)))(\forall b)[b \mathrel{\dot\in} y \leftrightarrow n\mathsf{Main}_a^{\mathsf{u}_{n+1}(x)}(b)].$$

Proof. Reason in S_{n+1} and assume that x is a name so that $\mathsf{Hier}_n(\mathsf{u}_{n+1}(x), a)$ holds. In particular, we have for each $z \mathrel{\dot\in} \mathsf{u}_{n+1}(x)$,

$$(1) \qquad (\forall c \prec a)\mathsf{h}_n zc \mathrel{\dot\in} \mathsf{u}_{n+1}(x).$$

Applying join twice to (1) allows us to conclude that (a name of) the type

$$(2) \qquad \{(c, u, v) : c \prec a \,\wedge\, u \mathrel{\dot\in} \mathsf{h}_n zc \wedge v \mathrel{\dot\in} u\}$$

belongs to the universe $\mathsf{u}_{n+1}(x)$ (and hence also to $\mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x))$). Since the name of the type (2) is *uniformly* given in each $z \mathrel{\dot\in} \mathsf{u}_{n+1}(x)$ we can apply join in the universe $\mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x))$ in order to obtain a name of the type

$$(3) \qquad \{(z, c, u, v) : z \mathrel{\dot\in} \mathsf{u}_{n+1}(x) \,\wedge\, c \prec a \,\wedge\, u \mathrel{\dot\in} \mathsf{h}_n zc \wedge v \mathrel{\dot\in} u\}$$

in the universe $\mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x))$. But now, clearly, $\{b : n\mathsf{Main}_a^{\mathsf{u}_{n+1}(x)}(b)\}$ is given elementarily in the type (3) and, hence, the claim of our lemma is proved. $\Box$

The crucial result concerning the formulas $n\mathsf{Main}_a^x(b)$ is the following progressivity lemma, which is the analogue of Lemma 12. The proof is literally the same as the one given in all detail for Lemma 12, with the only difference that it uses Lemma 18 and Lemma 17 instead of Lemma 11 and Lemma 10. Hence, we can state the following assertion without proof.

Lemma 20 *We have that* S_{n+1} *proves*

$$\Re(x) \wedge \mathsf{Hier}_n(\mathsf{u}_{n+1}(x), a) \;\rightarrow\; \mathsf{Prog}(\lambda b.n\mathsf{Main}_a^{\mathsf{u}_{n+1}(x)}(b)).$$

This concludes our preparatory work towards a proof of $(\star\star)$ in S_{n+1}, which is now immediate.

Proof of $(\star\star)$ concluded. Let x be a name and suppose

$$(\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(\mathsf{u}_{n+1}(x)))\mathsf{TI}(y, a). \tag{1}$$

Given this assumption, it is our aim to derive

$$(\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(x))\mathsf{TI}(y, \varphi(n{+}1)a0). \tag{2}$$

We can assume without loss of generality that a is an ε number, since universes are closed under arithmetical comprehension. Thus, it is enough to establish

$$(\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(x))\mathsf{TI}(y, \varphi(n{+}1)b0) \tag{3}$$

for each $b \prec a$. We fix such a b and observe that we also have $\omega^{1+b} \cdot \omega \prec a$. Further, by our assumption (1), Lemma 16 and Lemma 20 we have

$$\mathsf{Prog}(\lambda e.n\mathsf{Main}_a^{\mathsf{u}_{n+1}(x)}(e)). \tag{4}$$

But (4) together with (1), Lemma 16 and Lemma 19 immediately show that we have $n\mathsf{Main}_a^{\mathsf{u}_{n+1}(x)}(b)$, i.e. spelled out

$$(\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(x))(\forall c, d)[d \preceq a \,\wedge\, \omega^{1+b} \uparrow d \,\wedge\, n\mathsf{I}_y^d(c) \;\rightarrow\; n\mathsf{I}_y^d(\varphi(n{+}1)bc)]. \tag{5}$$

By choosing $c = 0$ and $d = \omega^{1+b} \cdot \omega$ in (5) we get

$$(\forall y \,\dot{\in}\, \mathsf{u}_{n+1}(x))n\mathsf{I}_y^{\omega^{1+b}\cdot\omega}(\varphi(n{+}1)b0). \tag{6}$$

But now one immediately realizes that (6) entails (3). Since $b \prec a$ was arbitrary, we have thus shown (2). This is as desired and ends our proof of $(\star\star)$, given that the assumption $(\star)$ of our main lemma holds for n. All together this concludes our proof of Main Lemma 15. $\Box$

In exactly the same manner as in the proof of Theorem 8 we can now successively apply our Main Lemma 15 in order to get the desired lower bound for the theories S_n.

Theorem 21 *We have for all natural numbers n and all ordinals α less than $\varphi(n{+}1)00$ that S_n proves $\mathsf{I}(\alpha)$. Thus, $\varphi(n{+}1)00 \leq |\mathsf{S}_n|$.*

Due to Lemma 14 we have now established the desired lower bound for EMA.

Corollary 22 $\varphi\omega 00 \leq |\mathsf{EMA}|$.

We conclude this section by making a few remarks concerning the lower bound of EMA augmented by the full schema $(\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$ of complete induction on the natural numbers. The lower bound computations given so far can be extended in a rather straightforward manner in order to yield $\varphi\varepsilon_0 00$ as a proof-theoretic lower bound of the systems $\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$. The principal benefit of the induction schema compared to the induction axiom is that one has available α-hyperuniverses for α less than ε_0 instead of n-hyperuniverses for n less than ω.

Theorem 23 $\varphi\varepsilon_0 00 \leq |\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})|$.

We will establish in the following two chapters that the lower bounds proved for the systems ETJ, EIN, and EMA, possibly augmented by the schema $(\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$ of formula induction, are indeed best possible.

Chapter 3

From explicit mathematics to theories with ordinals

In this chapter we will introduce certain ordinal theories over Peano arithmetic PA, and we will provide formalized model constructions for ETJ, EIN, and EMA in these theories, cf. Jäger and Strahm [66]. A detailed proof-theoretic analysis of theories with ordinals in the next chapter will yield sharp upper bounds for the various systems of explicit mathematics.

The language and spirit of our ordinal theories is related to Jäger's framework of Peano arithmetic with ordinals PA_Ω, cf. Jäger [61], which has previously been used in the proof-theoretic treatment of various applicative theories and systems of explicit mathematics, cf. Feferman and Jäger [35, 36], Jäger and Strahm [67, 68, 69], Marzetta and Strahm [84], and Strahm [123]. Further, numerous impredicative ordinal theories have been extensively studied by Arai in connection with his program of finitary analysis of proof figures, cf. e.g. Arai [3].

Below we will define the three ordinal theories OAD, OIN, and EMA, whose universe of discourse is *admissible*, *recursively inaccessible*, and *Mahlo*, respectively. All three theories are first order frameworks tailored for dealing with certain *non-monotone* inductive definitions. Crucial for their strength are the various *reflection principles* and the fact that *induction on the ordinals is omitted* completely. Thus, the slogan for our ordinals (or: pseudo-ordinals) is: *strong reflection, but no foundation.*

3.1 Introducing theories with ordinals

All our ordinal theories are based on the language $\mathcal{L}_1$ of first order arithmetic. $\mathcal{L}_1$ has *number variables* $a, b, c, d, e, f, u, v, w, x, y, z, \dots$ (possibly with subscripts), symbols for all primitive recursive functions and relations, as well as a unary relation symbol Q. Q plays the role of an anonymous relation variable with no specific meaning (cf. Definition 4 above). There is also a symbol $\sim$ for forming negative literals.[1]

The *number terms* $(r, s, t, r_1, s_1, t_1, \dots)$ of $\mathcal{L}_1$ are defined as usual. Positive literals of $\mathcal{L}_1$ are all expressions $R(s_1, \dots, s_n)$ for R a symbol for an n-ary primitive recursive relation as well as expressions of the form $\mathsf{Q}(s)$. The negative literals of $\mathcal{L}_1$ have the form $\sim E$ so that E is a positive literal. The *formulas* of $\mathcal{L}_1$ are now generated from the positive and negative literals of $\mathcal{L}_1$ by closing against disjunction, conjunction, as well as existential and universal number quantification. The negation $\neg A$ of an $\mathcal{L}_1$ formula A is defined by making use of De Morgan's laws and the law of double negation. Moreover, the remaining logical connectives are abbreviated in the standard manner.

In the following we make use of the usual primitive recursive coding machinery in $\mathcal{L}_1$: $\langle \dots \rangle$ is a standard primitive recursive function for forming n-tuples $\langle t_1, \dots, t_n \rangle$; *Seq* is the primitive recursive set of sequence numbers; $lh(t)$ denotes the length of (the sequence number coded by) t; $(t)_i$ is the ith component of (the sequence coded by) t if $i < lh(t)$, i.e. $t = \langle (t)_0, \dots, (t)_{lh(t) \dot{-} 1} \rangle$ if t is a sequence number.

Now let P be a fresh n-ary relation symbol and write $\mathcal{L}_1(\mathsf{P})$ for the extension of $\mathcal{L}_1$ by P. An $\mathcal{L}_1(\mathsf{P})$ formula which contains at most $a_1, \dots, a_n$ free is called an n-ary *operator form*, and we let $\mathcal{A}(\mathsf{P}, a_1, \dots, a_n)$ range over such forms.

All our ordinal theories are formulated in the language $\mathcal{L}_{\mathbb{O}}$ which extends $\mathcal{L}_1$ by adding a new sort of ordinal variables $\sigma, \tau, \eta, \xi, \dots$ (possibly with subscripts), new binary relation symbols $<$ and $=$ for the less and equality relation on the ordinals, respectively, and a unary relation symbol Ad to express that an ordinal is admissible. Moreover, $\mathcal{L}_{\mathbb{O}}$ includes an $(n{+}1)$-ary relation symbol $\mathsf{P}_{\mathcal{A}}$ for each operator form $\mathcal{A}(\mathsf{P}, a_1, \dots, a_n)$.

[1] This formulation of the language is chosen for the Tait-style reformulation of our systems in the next section.

The number terms of $\mathcal{L}_{\mathbb{O}}$ are the number terms of $\mathcal{L}_1$, and the ordinal terms of $\mathcal{L}_{\mathbb{O}}$ are the ordinal variables. The positive literals of $\mathcal{L}_{\mathbb{O}}$ are the positive literals of $\mathcal{L}_1$ plus all expressions $(\sigma < \tau)$, $(\sigma = \tau)$ and $\mathsf{P}_{\mathcal{A}}(\sigma, \vec{s})$ for each n-ary operator form $\mathcal{A}(\mathsf{P}, \vec{a})$. We write $\mathsf{P}^{\sigma}_{\mathcal{A}}(\vec{s})$ for $\mathsf{P}_{\mathcal{A}}(\sigma, \vec{s})$. The negative literals of $\mathcal{L}_{\mathbb{O}}$ are the expressions $\sim\!E$ with E a positive literal of $\mathcal{L}_{\mathbb{O}}$.

The *formulas* $(A, B, C, A_1, B_1, C_1, \ldots)$ of $\mathcal{L}_{\mathbb{O}}$ are generated from the positive and negative literals by closing under conjunction and disjunction, quantification over natural numbers, and the bounded ordinal quantifiers $(\exists \xi < \sigma)$ and $(\forall \xi < \sigma)$ as well as the unbounded ordinal quantifiers $(\exists \xi)$ and $(\forall \xi)$.

An $\mathcal{L}_{\mathbb{O}}$ formula is called $\Sigma^{\mathbb{O}}$ if it does not contain ordinal quantifiers of the form $(\forall \xi)$; it is called $\Pi^{\mathbb{O}}$ if it does not have ordinal quantifiers of the form $(\exists \xi)$. Finally, the $\Delta^{\mathbb{O}}_0$ formulas of $\mathcal{L}_{\mathbb{O}}$ are those formulas which are both $\Sigma^{\mathbb{O}}$ and $\Pi^{\mathbb{O}}$; the $\Sigma^{\mathbb{O}}_1$ formulas of $\mathcal{L}_{\mathbb{O}}$ are the $\Delta^{\mathbb{O}}_0$ formulas plus all formulas of the form $(\exists \xi) A(\xi)$ with A a $\Delta^{\mathbb{O}}_0$ formula and accordingly for $\Pi^{\mathbb{O}}_1$ formulas. Further, we write A^{σ} to denote the $\mathcal{L}_{\mathbb{O}}$ formula which is obtained from A by replacing all unbounded ordinal quantifiers $(Q\xi)$ in A by bounded ordinal quantifiers $(Q\xi < \sigma)$. Additional abbreviations are

$$\mathsf{P}^{<\sigma}_{\mathcal{A}}(\vec{s}) \;:=\; (\exists \xi < \sigma)\mathsf{P}^{\xi}_{\mathcal{A}}(\vec{s}) \qquad \text{and} \qquad \mathsf{P}_{\mathcal{A}}(\vec{s}) \;:=\; (\exists \xi)\mathsf{P}^{\xi}_{\mathcal{A}}(\vec{s}).$$

The stage is now set in order to introduce our base theory OAD, whose universe of ordinals is admissible. For reasons of organizational simplicity we have also included in OAD the basic axioms about Ad, although OAD does *not* prove the existence of an admissible ordinal.

OAD is formulated in classical two-sorted predicate logic with equality in both sorts, containing the axioms of Peano arithmetic PA, linearity axioms for ordinals, operator axioms, $\Sigma^{\mathbb{O}}$ reflection axioms, Ad axioms, and $\Delta^{\mathbb{O}}_0$ induction on the natural numbers.

I. Number-theoretic axioms. The axioms of Peano arithmetic PA with the exception of complete induction on the natural numbers.

II. Linearity axioms.

$$\sigma \not< \sigma \;\wedge\; (\sigma < \tau \wedge \tau < \eta \rightarrow \sigma < \eta) \;\wedge\; (\sigma < \tau \vee \sigma = \tau \vee \tau < \sigma).$$

III. Operator axioms. For all operator forms $\mathcal{A}(\mathsf{P}, \vec{a})$:

$$\mathsf{P}^{\sigma}_{\mathcal{A}}(\vec{s}) \leftrightarrow \mathcal{A}(\mathsf{P}^{<\sigma}_{\mathcal{A}}, \vec{s}).$$

IV. $\Sigma^{\mathbb{O}}$ reflection axioms. For all $\Sigma^{\mathbb{O}}$ formulas A:

$$A \to (\exists \xi) A^{\xi}.$$

V. Axioms for Ad. For all $\Sigma^{\mathbb{O}}$ formulas $A(\vec{\tau})$ whose free ordinal variables are from the list $\vec{\tau}$:

$$\mathsf{Ad}(\sigma) \wedge \vec{\tau} < \sigma \wedge A^{\sigma}(\vec{\tau}) \;\to\; (\exists \xi < \sigma) A^{\xi}(\vec{\tau}).$$

VI. $\Delta_0^{\mathbb{O}}$ induction on the natural numbers. For all $\Delta_0^{\mathbb{O}}$ formulas $A(a)$:

$$(\Delta_0^{\mathbb{O}}\text{-}\mathsf{I_N}) \qquad A(0) \wedge (\forall x)(A(x) \to A(x')) \;\to\; (\forall x) A(x).$$

We observe that there are no induction principles for ordinals and that induction on the natural numbers is restricted to $\Delta_0^{\mathbb{O}}$ formulas.

This concludes the description of our base theory OAD. The two systems OIN and OMA are now obtained from OAD by adding the limit axiom and the Mahlo axioms, respectively. The limit axiom for ordinals claims that for each ordinal there is a greater *admissible* ordinal; this is formalized by the following simple axiom (L-Ad):

$$(\mathsf{L\text{-}Ad}) \qquad (\forall \sigma)(\exists \tau)(\sigma < \tau \;\wedge\; \mathsf{Ad}(\tau))$$

The $\mathcal{L}_{\mathbb{O}}$ theory OIN is defined to be OAD plus (L-Ad). Since its ordinal universe of discourse is a limit of admissible ordinals and, in addition, satisfies $\Sigma^{\mathbb{O}}$ reflection on the ordinals, we have that OIN describes a *recursively inaccessible* universe of ordinals.

The principle corresponding to Mahloness in our ordinal-theoretic framework is so-called $\Pi_2^{\mathbb{O}}$ *reflection on admissible ordinals.* The crucial schema ($\Pi_2^{\mathbb{O}}$-Ref-Ad) includes for all $\Delta_0^{\mathbb{O}}$ formulas $A(\xi, \eta, \vec{\tau})$ whose free ordinal variables are from the list $\xi, \eta, \vec{\tau}$:

$$(\Pi_2^{\mathbb{O}}\text{-}\mathsf{Ref\text{-}Ad}) \qquad \begin{array}{l} (\forall \xi)(\exists \eta) A(\xi, \eta, \vec{\tau}) \;\to \\ \qquad (\exists \sigma)[\mathsf{Ad}(\sigma) \wedge \vec{\tau} < \sigma \wedge (\forall \xi < \sigma)(\exists \eta < \sigma) A(\xi, \eta, \vec{\tau})]. \end{array}$$

Thus, ($\Pi_2^{\mathbb{O}}$-Ref-Ad) expresses that each true $\Pi_2^{\mathbb{O}}$ formula is reflected by an admissible ordinal which is greater than all the parameters of the formula. We let OMA be the $\mathcal{L}_{\mathbb{O}}$ theory which extends OAD by the schema ($\Pi_2^{\mathbb{O}}$-Ref-Ad).

In the following we are also interested in extending our three ordinal theories by the schema ($\mathcal{L}_{\mathbb{O}}$-$\mathsf{I_N}$) of complete induction on the natural numbers for *arbitrary* formulas in the language $\mathcal{L}_{\mathbb{O}}$. Accordingly, let us write in the sequel $\mathsf{OAD} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$, $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$, and $\mathsf{OMA} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ for the corresponding extensions of OAD, OIN, and OMA, respectively.

This concludes the description of the ordinal theories which will be relevant in the sequel. In the following section we will show how to model the systems of explicit mathematics ETJ, EIN, and EMA in the ordinal theories OAD, OIN, and OMA.

3.2 Embedding ETJ into OAD

In this section we show how to build a formalized model construction for ETJ in the ordinal theory OAD. This construction will be extended in the next two sections in order to yield an interpretation of EIN and EMA in OIN and OMA, respectively.

The crucial idea for interpreting ETJ in OAD is to choose a suitable operator form $\mathcal{A}(\mathsf{P}, a, b, c)$ so that the relation symbol $\mathsf{P}_{\mathcal{A}}$ can then be used to single out the numbers which name types, and to define elementhood in the names of types. Before doing this, we have to translate term application and the individual constants of the language $\mathbb{L}$ into $\mathcal{L}_1$.

We interpret application $\cdot$ of $\mathbb{L}$ in the sense of ordinary recursion theory so that $(a{\cdot}b)$ in $\mathbb{L}$ is translated into $\{a\}(b)$ in $\mathcal{L}_1$, where $\{n\}$ for $n = 0, 1, 2, 3, \ldots$ is a standard enumeration of the partial recursive functions. Then it is possible to assign pairwise different numerals to the constants $\mathsf{k}, \mathsf{s}, \mathsf{p}, \mathsf{p_0}, \mathsf{p_1}, \mathsf{s_N}, \mathsf{p_N}$ and $\mathsf{d_N}$ so that the applicative axioms (1)–(8) of ETJ are satisfied. We also require that the constant 0 of $\mathbb{L}$ is interpreted as the 0 of $\mathcal{L}_1$ and the term $\mathsf{s_N}a$ of $\mathbb{L}$ as $a{+}1$ in $\mathcal{L}_1$. In addition, we let pairing and projections of $\mathbb{L}$ go over into the primitive recursive pairing and unpairing machinery introduced above.

Further, for each $\mathbb{L}$ term t there exists an $\mathcal{L}_1$ formula $\mathsf{Val}_t(a)$ expressing that a is the value of t under the interpretation described above. Accordingly, the atomic formulas $t{\downarrow}$, $(s = t)$ and $\mathsf{N}(t)$ are given their obvious interpretations in $\mathcal{L}_1$ with the translation of N ranging over all natural numbers.

For dealing with the generators we choose, again by ordinary recursion theory, numerals $\underline{\mathsf{nat}}, \underline{\mathsf{q}}, \underline{\mathsf{id}}, \underline{\mathsf{co}}, \underline{\mathsf{int}}, \underline{\mathsf{dom}}, \underline{\mathsf{inv}}, \underline{\mathsf{j}}, \underline{\mathsf{u}}$, and $\underline{\mathsf{m}}$ so that we have the following properties:

$$\underline{\mathsf{nat}} = \langle 0, 0\rangle,\ \underline{\mathsf{q}} = \langle 1, 0\rangle,\ \underline{\mathsf{id}} = \langle 2, 0\rangle,\ \{\underline{\mathsf{co}}\}(a) = \langle 3, a\rangle,$$

$$\{\underline{\mathsf{int}}\}(\langle a, b\rangle) = \langle 4, a, b\rangle,\ \{\underline{\mathsf{dom}}\}(a) = \langle 5, a\rangle,\ \{\underline{\mathsf{inv}}\}(\langle a, b\rangle) = \langle 6, a, b\rangle,$$

$$\{\underline{\mathsf{j}}\}(\langle a, b\rangle) = \langle 7, a, b\rangle,\ \{\underline{\mathsf{u}}\}(a) = \langle 8, a\rangle,\ \{\underline{\mathsf{m}}\}(\langle a, f\rangle) = \langle 9, a, f\rangle,$$

$$\{e_0\}(a) \neq e_1$$

for all natural numbers a, b and all e_0 and e_1 from the set ranging over $\underline{\mathsf{nat}}, \underline{\mathsf{q}}, \underline{\mathsf{id}}, \underline{\mathsf{co}}, \underline{\mathsf{int}}, \underline{\mathsf{dom}}, \underline{\mathsf{inv}}, \underline{\mathsf{j}}, \underline{\mathsf{u}}$, and $\underline{\mathsf{m}}$. Of course, $\underline{\mathsf{u}}$ and $\underline{\mathsf{m}}$ will not yet play a role in this section and only be used later.

As mentioned above, it is our strategy to define a specific operator form $\mathcal{A}(\mathsf{P}, a, b, c)$ and use the corresponding relation symbol $\mathsf{P}_{\mathcal{A}}$ for dealing with codes for types and elements of types. Later our interpretation will be so that

$$\Re(a) \text{ translates into } (\exists\xi)\mathsf{P}^{\xi}_{\mathcal{A}}(a, 0, 0) \text{ and}$$
$$b \mathrel{\dot\in} a \text{ translates into } (\exists\xi)\mathsf{P}^{\xi}_{\mathcal{A}}(a, b, 1).$$

Before turning to our final operator form $\mathcal{A}(\mathsf{P}, a, b, c)$ we introduce the auxiliary ternary operator form $\mathcal{A}_0(\mathsf{P}, a, b, c)$ which is the disjunction of the following formulas (1)–(16):

(1) $a = \langle 0, 0\rangle \ \wedge\ b = 0 \ \wedge\ c = 0,$

(2) $a = \langle 0, 0\rangle \ \wedge\ c = 1,$

(3) $a = \langle 1, 0\rangle \ \wedge\ b = 0 \ \wedge\ c = 0,$

(4) $a = \langle 1, 0\rangle \ \wedge\ \mathsf{Q}(b) \ \wedge\ c = 1,$

(5) $a = \langle 2, 0\rangle \ \wedge\ b = 0 \ \wedge\ c = 0,$

(6) $a = \langle 2, 0\rangle \ \wedge\ (\exists x)(b = \langle x, x\rangle) \ \wedge\ c = 1,$

(7) $(\exists u)[a = \langle 3, u\rangle \ \wedge\ \mathsf{P}(u, 0, 0)] \ \wedge\ b = 0 \ \wedge\ c = 0,$

(8) $(\exists u)[a = \langle 3, u\rangle \ \wedge\ \mathsf{P}(u, 0, 0) \ \wedge\ \neg\mathsf{P}(u, b, 1)] \ \wedge\ c = 1,$

(9) $(\exists u, v)[a = \langle 4, u, v\rangle \ \wedge\ \mathsf{P}(u, 0, 0) \ \wedge\ \mathsf{P}(v, 0, 0)] \ \wedge\ b = 0 \ \wedge\ c = 0,$

(10) $(\exists u, v)[a = \langle 4, u, v\rangle \wedge \mathsf{P}(u, 0, 0) \wedge \mathsf{P}(v, 0, 0) \wedge \mathsf{P}(u, b, 1) \wedge \mathsf{P}(v, b, 1)]$
$\wedge\ c = 1$,

(11) $(\exists u)[a = \langle 5, u\rangle \wedge \mathsf{P}(u, 0, 0)] \wedge b = 0 \wedge c = 0$,

(12) $(\exists u, x)[a = \langle 5, u\rangle \wedge \mathsf{P}(u, 0, 0) \wedge \mathsf{P}(u, \langle b, x\rangle, 1)] \wedge c = 1$,

(13) $(\exists u, f)[a = \langle 6, u, f\rangle \wedge \mathsf{P}(u, 0, 0)] \wedge b = 0 \wedge c = 0$,

(14) $(\exists u, f)[a = \langle 6, u, f\rangle \wedge \mathsf{P}(u, 0, 0) \wedge \mathsf{P}(u, \{f\}(b), 1)] \wedge c = 1$,

(15) $(\exists u, f)[a = \langle 7, u, f\rangle \wedge \mathsf{P}(u, 0, 0) \wedge (\forall x)(\mathsf{P}(u, x, 1) \rightarrow \mathsf{P}(\{f\}(x), 0, 0))]$
$\wedge\ b = 0 \wedge c = 0$,

(16) $(\exists u, f)[a = \langle 7, u, f\rangle \wedge \mathsf{P}(u, 0, 0) \wedge (\forall x)(\mathsf{P}(u, x, 1) \rightarrow \mathsf{P}(\{f\}(x), 0, 0))$
$\wedge\ (\exists y, z)(b = \langle y, z\rangle \wedge \mathsf{P}(u, y, 1) \wedge \mathsf{P}(\{f\}(y), z, 1))] \wedge c = 1$.

If we had foundation on the ordinals, this operator form $\mathcal{A}_0(\mathsf{P}, a, b, c)$ would be sufficient for our model construction. By induction on the ordinals we could show for example that $(\exists \xi)\mathsf{P}^{\xi}_{\mathcal{A}_0}(a, 0, 0)$ implies that there is a least such ξ. In our context, however, induction on the ordinals is not available. Thus, in order to have a "unique time stamp" for triples (a, b, c) to get into stages generated, we work with the following operator form $\mathcal{A}(\mathsf{P}, a, b, c)$:

$$\mathcal{A}(\mathsf{P}, a, b, c) \;:=\; \mathcal{A}_0(\mathsf{P}, a, b, c) \wedge \neg\mathsf{P}(a, 0, 0).$$

Given this careful definition of the operator form $\mathcal{A}(\mathsf{P}, a, b, c)$, the following lemma concerning the stages of $\mathcal{A}(\mathsf{P}, a, b, c)$ is trivially provable in OAD:

Lemma 24 *The following assertions are provable in* OAD*:*

1. $\mathsf{P}^{\sigma}_{\mathcal{A}}(a, 0, 0) \wedge \mathsf{P}^{\tau}_{\mathcal{A}}(a, 0, 0) \rightarrow \sigma = \tau$,
2. $\mathsf{P}^{\sigma}_{\mathcal{A}}(a, b, 1) \rightarrow \mathsf{P}^{\sigma}_{\mathcal{A}}(a, 0, 0)$,
3. $\mathsf{P}^{\sigma}_{\mathcal{A}}(a, 0, 0) \rightarrow (\forall b)[\mathsf{P}_{\mathcal{A}}(a, b, 1) \leftrightarrow \mathsf{P}^{\sigma}_{\mathcal{A}}(a, b, 1)]$.

Before turning to the interpretation of the types, the $\in$ relation and the naming relation we introduce the following definitions:

$$\mathsf{Rep}(a) \;:=\; (\exists \xi)\mathsf{P}^{\xi}_{\mathcal{A}}(a, 0, 0), \quad \mathsf{E}(b, a) \;:=\; (\exists \xi)\mathsf{P}^{\xi}_{\mathcal{A}}(a, b, 1).$$

In our embedding of ETJ into OAD we first assume that the number and types variables of $\mathbb{L}$ are mapped into the number variables of $\mathcal{L}_{\mathbb{O}}$ so that no conflicts arise; to simplify the notation we often identify the type variables with their translations in $\mathcal{L}_{\mathbb{O}}$. Then we let the type variables of ETJ range over Rep and the translation of the atomic formulas of $\mathbb{L}$ involving types is as follows:

$$\begin{array}{rcl} \Re(t,U)^* & := & (\exists x)[\mathsf{Val}_t(x) \wedge \mathsf{Rep}(x) \wedge \mathsf{Rep}(U) \wedge (\forall y)(\mathsf{E}(y,x) \leftrightarrow \mathsf{E}(y,U))], \\ (t \in U)^* & := & (\exists x)[\mathsf{Val}_t(x) \wedge \mathsf{E}(x,U)], \\ (U = V)^* & := & (\forall x)(\mathsf{E}(x,U) \leftrightarrow \mathsf{E}(x,V)). \end{array}$$

On the basis of these basic cases the translation of arbitrary $\mathbb{L}$ formulas A into $\mathcal{L}_{\mathbb{O}}$ formulas A^* should be obvious. The embedding of ETJ into OAD is given by the following theorem.

Theorem 25 *We have for all $\mathbb{L}$ formulas $A(\vec{U},\vec{a})$ with all its free variables indicated that*

$$\mathsf{ETJ} \vdash A(\vec{U},\vec{a}) \quad \Longrightarrow \quad \mathsf{OAD} \vdash \mathsf{Rep}(\vec{U}) \;\rightarrow\; A^*(\vec{U},\vec{a}).$$

Moreover, this embedding carries over to the presence of full formula induction ($\mathbb{L}$-$\mathsf{I_N}$) *and* ($\mathcal{L}_{\mathbb{O}}$-$\mathsf{I_N}$)*, respectively.*

Proof. The proof proceeds by induction on the length of the derivation of the formula A. If A is an applicative axiom or an axiom concerning the uniqueness of generators, then its translation is provable in OAD by our assumptions about the coding of the first order part of ETJ. The translations of the axioms about explicit representation and extensionality are easily verified. In the case of the basic type existence axioms we confine ourselves to showing the translation of the axioms about *Intersection* and *Join*.

Let us first turn to *Intersection*. Assume we are given two natural numbers a and b so that $\mathsf{Rep}(a)$ and $\mathsf{Rep}(b)$. Hence, there exist ordinals σ and τ with $\mathsf{P}^{\sigma}_{\mathcal{A}}(a,0,0)$ and $\mathsf{P}^{\tau}_{\mathcal{A}}(b,0,0)$. Using $\Sigma^{\mathbb{O}}$ reflection, choose an ordinal η greater than σ and τ, and carry through the following distinction by cases.

Case 1: $\neg\mathsf{P}^{<\eta}_{\mathcal{A}}(\langle 4,a,b\rangle,0,0)$. Then our operator form $\mathcal{A}(\mathsf{P},a,b,c)$ yields $\mathsf{P}^{\eta}_{\mathcal{A}}(\langle 4,a,b\rangle,0,0)$. Moreover, we also have

$$(\forall x)[\mathsf{P}^{\eta}_{\mathcal{A}}(\langle 4,a,b\rangle,x,1) \;\leftrightarrow\; \mathsf{P}^{<\eta}_{\mathcal{A}}(a,x,1) \wedge \mathsf{P}^{<\eta}_{\mathcal{A}}(b,x,1)].$$

In view of Lemma 24 we thus have $\mathsf{Rep}(\langle 4,a,b\rangle)$ and for all natural numbers x that $\mathsf{E}(x,\langle 4,a,b\rangle)$ if and only if $\mathsf{E}(x,a)$ and $\mathsf{E}(x,b)$.

Case 2: $\mathsf{P}^{<\eta}_{\mathcal{A}}(\langle 4, a, b\rangle, 0, 0)$. Because of Lemma 24 there exists a unique ξ less than η so that $\mathsf{P}^{\xi}_{\mathcal{A}}(\langle 4, a, b\rangle, 0, 0)$. Hence, the operator form $\mathcal{A}(\mathsf{P}, a, b, c)$ forces $\mathsf{P}^{<\xi}_{\mathcal{A}}(a, 0, 0)$, $\mathsf{P}^{<\xi}_{\mathcal{A}}(b, 0, 0)$ and $\neg\mathsf{P}^{<\xi}_{\mathcal{A}}(\langle 4, a, b\rangle, 0, 0)$. Now we proceed as in the previous case.

In a next step we discuss *Join*. For that purpose assume that we are given natural numbers a and f so that $\mathsf{Rep}(a)$ and

$$(1) \qquad (\forall x)(\mathsf{E}(x, a) \;\rightarrow\; \mathsf{Rep}(\{f\}(x))).$$

Hence, there is an ordinal τ with $\mathsf{P}^{\tau}_{\mathcal{A}}(a, 0, 0)$. Moreover, thanks to Lemma 24, (1) is equivalent to the assertion

$$(2) \qquad (\forall x)(\mathsf{P}^{\tau}_{\mathcal{A}}(a, x, 1) \;\rightarrow\; (\exists \xi)\mathsf{P}^{\xi}_{\mathcal{A}}(\{f\}(x), 0, 0).$$

We are now in a position to apply $\Sigma^{\mathbb{O}}$ reflection to (2) in order to find an ordinal η greater than τ so that

$$(3) \qquad (\forall x)(\mathsf{P}^{\tau}_{\mathcal{A}}(a, x, 1) \;\rightarrow\; \mathsf{P}^{<\eta}_{\mathcal{A}}(\{f\}(x), 0, 0).$$

Using (3) we can now proceed by case distinction on $\neg\mathsf{P}^{\eta}_{\mathcal{A}}(\langle 7, a, f\rangle, 0, 0)$, respectively $\mathsf{P}^{\eta}_{\mathcal{A}}(\langle 7, a, f\rangle, 0, 0)$ as before in order to obtain the desired conclusion concerning *Join*.

Finally, let us verify the translation of type induction on the natural numbers $(\mathsf{T}\text{-}\mathsf{I}_{\mathsf{N}})$. Given an a so that $\mathsf{Rep}(a)$ as well as

$$\mathsf{E}(0, a) \;\wedge\; (\forall x)(\mathsf{E}(x, a) \rightarrow \mathsf{E}(x', a)),$$

the desired conclusion is immediate by $\Delta^{\mathbb{O}}_0$ induction on the natural numbers $(\Delta^{\mathbb{O}}_0\text{-}\mathsf{I}_{\mathsf{N}})$ and using the fact that there is an ordinal number σ so that for all natural numbers x, $\mathsf{E}(x, a)$ is equivalent to $\mathsf{P}^{\sigma}_{\mathcal{A}}(a, x, 1)$. $\square$

3.3 Embedding EIN into OIN

In a further step we now aim at an extension of the $*$ translation in order to yield an embedding of EIN into OIN and of $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$ into $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})$.

The crucial new aspect in the interpretation of EIN is to take care of the limit axiom (L). First, for the recursion-theoretic interpretation of the generator

u, recall that we have chosen the natural number $\underline{\mathsf{u}}$ so that we have for all natural numbers a, $\{\underline{\mathsf{u}}\}(a) = \langle 8, a \rangle$.

In order to deal with the generator u we have to make sure that $\mathsf{u}(a)$ is only made a name provided that the codes generated so far constitute a universe containing a. We can now very elegantly use our operator $\mathcal{A}_0$ from above in order to express that the names given by P form a universe,

$$\mathsf{Univ}(\mathsf{P}) \;:=\; (\forall a, b, c)[\mathcal{A}_0(\mathsf{P}, a, b, c) \;\rightarrow\; \mathsf{P}(a, b, c)].$$

In a next step we now define the operator $\mathcal{B}_0(\mathsf{P}, a, b, c)$ to be the disjunction of $\mathcal{A}_0(\mathsf{P}, a, b, c)$ and the following formulas (17) and (18):

(17) $(\exists x)[a = \langle 8, x \rangle \;\wedge\; \mathsf{P}(x, 0, 0)] \;\wedge\; \mathsf{Univ}(\mathsf{P}) \;\wedge\; b = 0 \;\wedge\; c = 0,$

(18) $(\exists x)[a = \langle 8, x \rangle \;\wedge\; \mathsf{P}(x, 0, 0)] \;\wedge\; \mathsf{Univ}(\mathsf{P}) \;\wedge\; \mathsf{P}(b, 0, 0) \;\wedge\; c = 1.$

As before we can define the final operator form $\mathcal{B}(\mathsf{P}, a, b, c)$ for our interpretation of EIN in the following manner:

$$\mathcal{B}(\mathsf{P}, a, b, c) \;:=\; \mathcal{B}_0(\mathsf{P}, a, b, c) \;\wedge\; \neg\mathsf{P}(a, 0, 0).$$

The crucial lemma concerning "time stamps" (Lemma 24) now holds for $\mathsf{P}_{\mathcal{B}}$ instead of $\mathsf{P}_{\mathcal{A}}$. Moreover, the translation $*$ of $\mathbb{L}$ into $\mathcal{L}_{\mathbb{O}}$ is defined as above, but always using $\mathsf{P}_{\mathcal{B}}$ instead of $\mathsf{P}_{\mathcal{A}}$. We continue using the shorthand expressions $\mathsf{Rep}(a)$ and $\mathsf{E}(b, a)$ and always assume that the appropriate operator form is given by the context.

The embedding theorem of EIN $[+(\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})]$ into OIN $[+(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})]$ can now be stated in the expected manner.

Theorem 26 *We have for all $\mathbb{L}$ formulas $A(\vec{U}, \vec{a})$ with all its free variables indicated that*

$$\mathsf{EIN} \vdash A(\vec{U}, \vec{a}) \quad\Longrightarrow\quad \mathsf{OIN} \vdash \mathsf{Rep}(\vec{U}) \;\rightarrow\; A^*(\vec{U}, \vec{a}).$$

Moreover, this embedding carries over to the presence of full formula induction $(\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$ and $(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})$, respectively.

Proof. Large parts of the proof are identical to the proof of Theorem 25. In the following we confine ourselves to the main new points only.

First, we observe that that linearity ($\mathcal{U}_{no}$-Lin) and connectivity ($\mathcal{U}_{no}$-Con) of normal universes are easily satisfied by construction and the crucial fact that our ordinals are linearly ordered.

Before turning to the verification of (L) let us make one crucial remark concerning universes. Our treatment of the axioms of ETJ in OAD in the proof of Theorem 25 in fact revealed that our basic type existence axioms can already be validated at *admissible* stages of our construction, i.e.,

$$\mathsf{Ad}(\sigma) \rightarrow \mathsf{Univ}(\mathsf{P}^{<\sigma}_{\mathcal{B}}). \tag{1}$$

This observation is important for dealing with the limit axiom (L). For that purpose, assume that we are given a natural number a so that $\mathsf{Rep}(a)$ holds. Hence, there exists an ordinal number τ so that $\mathsf{P}^{\tau}_{\mathcal{B}}(a, 0, 0)$. Using ($\mathsf{L}$-$\mathsf{Ad}$), we can find an *admissible* ordinal σ so that $\tau < \sigma$. The admissibility of σ ensures $\mathsf{Univ}(\mathsf{P}^{<\sigma}_{\mathcal{B}})$ by (1). We can now proceed by case distinction:

Case 1: $\neg\mathsf{P}^{<\sigma}_{\mathcal{B}}(\langle 8, a\rangle, 0, 0)$. Then our operator $\mathcal{B}(\mathsf{P}, a, b, c)$ gives $\mathsf{P}^{\sigma}_{\mathcal{B}}(\langle 8, a\rangle, 0, 0)$ and, therefore, $\mathsf{Rep}(\langle 8, a\rangle)$. Our operator form also forces for all x,

$$\mathsf{E}(x, \langle 8, a\rangle) \leftrightarrow \mathsf{P}^{<\sigma}_{\mathcal{B}}(x, 0, 0). \tag{2}$$

This shows in particular that $\langle 8, a\rangle$ names a type which contains a. In view of $\mathsf{Univ}(\mathsf{P}^{<\sigma}_{\mathcal{B}})$ we indeed have by (2) that (the translation of) $\mathcal{U}(\langle 8, a\rangle)$ is true in our model.

Case 2: $\mathsf{P}^{<\sigma}_{\mathcal{B}}(\langle 8, a\rangle, 0, 0)$. Because of (the analogue of) Lemma 24 there exists a unique ξ less than σ so that $\mathsf{P}^{\xi}_{\mathcal{B}}(\langle 8, a, f\rangle)$. Hence, we have that the operator form $\mathcal{B}(\mathsf{P}, a, b, c)$ yields $\mathsf{P}^{<\xi}_{\mathcal{B}}(a, 0, 0)$ as well as $\mathsf{Univ}(\mathsf{P}^{<\xi}_{\mathcal{B}})$. The rest is as in the previous case.

Therefore, our limit axiom (L) is shown to be valid in our model, and this completes the proof of the embedding of EIN into OIN. □

3.4 Embedding EMA into OMA

The final section of this chapter is devoted to showing how to embed EMA into OMA. The central point will be to demonstrate how $\Pi^{\mathbb{O}}_2$ reflection on admissible ordinals ($\Pi^{\mathbb{O}}_2$-Ref-Ad) enables us to model the Mahlo axioms ($\mathsf{M}.1$) and ($\mathsf{M}.2$).

We first remind the reader that the generator m has been interpreted by a natural number $\underline{\mathsf{m}}$ so that for all a and f, $\{\mathsf{m}\}(\langle a, f\rangle) = \langle 9, a, f\rangle$. Our aim is now to modify the previous operator form $\mathcal{B}_0(\mathsf{P}, a, b, c)$ by replacing the two clauses (17) and (18) for the limit axiom (L) by two new clauses (19) and (20) taking care of the Mahlo axioms $(\mathsf{M}.1)$ and $(\mathsf{M}.2)$. Accordingly, we define the new operator form $\mathcal{C}_0(\mathsf{P}, a, b, c)$ to be the disjunction of $\mathcal{A}_0(\mathsf{P}, a, b, c)$ and the following clauses (19) and (20):

(19) $(\exists x, f)[a = \langle 9, x, f\rangle \land \mathsf{P}(x, 0, 0) \land (\forall y)(\mathsf{P}(y, 0, 0) \to \mathsf{P}(\{f\}(y), 0, 0))]$
$\land \mathsf{Univ}(\mathsf{P}) \land b = 0 \land c = 0$,

(20) $(\exists x, f)[a = \langle 9, x, f\rangle \land \mathsf{P}(x, 0, 0) \land (\forall y)(\mathsf{P}(y, 0, 0) \to \mathsf{P}(\{f\}(y), 0, 0))]$
$\land \mathsf{Univ}(\mathsf{P}) \land \mathsf{P}(b, 0, 0) \land c = 1$.

The by now standard modification of $\mathcal{C}_0(\mathsf{P}, a, b, c)$ to the final operator form $\mathcal{C}(\mathsf{P}, a, b, c)$ for interpreting EMA is as follows:

$$\mathcal{C}(\mathsf{P}, a, b, c) := \mathcal{C}_0(\mathsf{P}, a, b, c) \land \neg\mathsf{P}(a, 0, 0).$$

Using $\mathsf{P}_{\mathcal{C}}$ we are now in a position to define the expected translation $*$ from $\mathbb{L}$ into $\mathcal{L}_{\mathbb{O}}$. Moreover, we have the following crucial embedding theorem from EMA into OMA.

Theorem 27 *We have for all $\mathbb{L}$ formulas $A(\vec{U}, \vec{a})$ with all its free variables indicated that*

$$\mathsf{EMA} \vdash A(\vec{U}, \vec{a}) \quad \Longrightarrow \quad \mathsf{OMA} \vdash \mathsf{Rep}(\vec{U}) \to A^*(\vec{U}, \vec{a}).$$

Moreover, this embedding carries over to the presence of full formula induction $(\mathbb{L}\text{-}\mathsf{I_N})$ and $(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$, respectively.

Proof. The only new axioms to be discussed in this proof are the Mahlo axioms $(\mathsf{M}.1)$ and $(\mathsf{M}.2)$. To this end assume that we have a and f so that $\mathsf{Rep}(a)$ and $(\forall x)(\mathsf{Rep}(x) \to \mathsf{Rep}(\{f\}(x))$. Hence, there exists a τ so that

(1) $$\mathsf{P}^{\tau}_{\mathcal{C}}(a, 0, 0).$$

A simple transformation of our second assumption yields, in addition, that

(2) $$(\forall \xi)(\forall x)(\exists \eta)[\mathsf{P}^{<\xi}_{\mathcal{C}}(x, 0, 0) \to \mathsf{P}^{<\eta}_{\mathcal{C}}(\{f\}(x), 0, 0)],$$

and, therefore, $\Sigma^{\mathbb{O}}$ reflection applied to (2) gives

$$(3) \qquad (\forall \xi)(\exists \eta)(\forall x)[\mathsf{P}_{\mathcal{C}}^{<\xi}(x,0,0) \rightarrow \mathsf{P}_{\mathcal{C}}^{<\eta}(\{f\}(x),0,0)].$$

Hence, $\Pi_2^{\mathbb{O}}$ reflection on Ad, ($\Pi_2^{\mathbb{O}}$-Ref-Ad), provides an admissible σ so that

$$(4) \qquad \tau < \sigma \wedge (\forall \xi < \sigma)(\exists \eta < \sigma)(\forall x)[\mathsf{P}_{\mathcal{C}}^{<\xi}(x,0,0) \rightarrow \mathsf{P}_{\mathcal{C}}^{<\eta}(\{f\}(x),0,0)].$$

In view of (1) and by a simple transformation we derive

$$(5) \qquad \mathsf{P}_{\mathcal{C}}^{<\sigma}(a,0,0) \wedge (\forall x)[\mathsf{P}_{\mathcal{C}}^{<\sigma}(x,0,0) \rightarrow \mathsf{P}_{\mathcal{C}}^{<\sigma}(\{f\}(x),0,0)].$$

As we have remarked above, the admissibility of σ forces $\mathsf{Univ}(\mathsf{P}_{\mathcal{C}}^{<\sigma})$. Using the last assertion (5) we are now in a position to proceed by definition by cases on $\neg\mathsf{P}_{\mathcal{C}}^{<\sigma}(\langle 9,a,f\rangle,0,0)$, respectively $\mathsf{P}_{\mathcal{C}}^{<\sigma}(\langle 9,a,f\rangle,0,0)$, in exactly the same manner as at the end of the proof of Theorem 26.

All together this concludes the embedding of EMA into OMA. $\square$

Chapter 4

Proof-theoretic analysis of theories with ordinals

In this chapter we will provide proof-theoretic upper bounds for the ordinal theories OIN and OMA, possibly augmented by the full induction schema $(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})$, cf. Jäger and Strahm [66]. Together with the embedding theorems of the previous chapter, we will finally obtain the desired upper proof-theoretic bounds for the four systems of explicit mathematics EIN, $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$, EMA, and $\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$. We will assume in the following that the reader is familiar with the tools and techniques of predicative proof theory, e.g. predicative cut elimination theorems and the method of asymmetric interpretation.

We will see, in particular, that the strength of OIN and $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})$ is already exhausted by claiming the existence of n, respectively α many admissible ordinals for n less than ω and α less than ε_0. Accordingly, OMA and $\mathsf{OMA} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})$ are reducible to axioms asserting the existence of n-inaccessible, respectively α-inaccessible ordinals for n less than ω and α less than ε_0. Hence, in order to conclude our proof-theoretic analysis, a detailed treatment of α-inaccessibility will be in order.

The analysis of theories with ordinals clearly makes use of methods of predicative proof theory only, although the corresponding reduction procedures are more complex than for well-known predicative systems. Thus, the *metapredicativity* of our ordinal theories and, hence, systems of explicit mathematics is witnessed.

4.1 Upper bounds for OAD

In the following let us very quickly sketch the well-known proof theory of the ordinal theories OAD and $\mathsf{OAD}+(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$. Since this is a very well-understood territory, we confine ourselves to mentioning the main techniques and giving pointers to the relevant literature.

Let us first turn to the theory OAD and recall that induction on the natural numbers is restricted in OAD to $\Delta_0^{\mathbb{O}}$ formulas. OAD is very similar to the theory $\mathsf{PA}^{\mathsf{r}}_{\Omega}$ of Jäger [61]. It is shown loc. cit. that $\mathsf{PA}^{\mathsf{r}}_{\Omega}$ is a conservative extension of Peano arithmetic PA. Indeed, one can literally follow the analysis of $\mathsf{PA}^{\mathsf{r}}_{\Omega}$ in [61] in order to establish the conservativity over PA of our ordinal theory OAD.

We briefly outline the analysis of OAD. In a *first step*, one reformulates OAD in Tait-style manner and observes that the main formulas of non-logical axioms and rules are $\Sigma^{\mathbb{O}}$. Hence, one obtains by *finite* partial cut elimination that all but $\Sigma^{\mathbb{O}}$ and $\Pi^{\mathbb{O}}$ cuts can be eliminated. In a *second step*, the $\Sigma^{\mathbb{O}}$-$\Pi^{\mathbb{O}}$ fragment of our Tait calculus is reduced to PA via an *asymmetric interpretation*: ordinal variables are replaced by finite ordinals so that a formula $\mathsf{P}_{\mathcal{A}}(m,\vec{s})$ with $m \in \mathbb{N}$ translates into an $\mathcal{L}_1$ formula that describes the build up in stages of the corresponding inductive definition, and if m is a bound for universal ordinal quantifiers, then $m{+}2^n$ provides a bound for existential ordinal quantifiers, where n is the length of a given quasi cut-free derivation. The so-obtained asymmetric interpretation validates $\Sigma^{\mathbb{O}}$ and $\Pi^{\mathbb{O}}$ cuts as well as $\Sigma^{\mathbb{O}}$ reflection. In addition, $\Delta_0^{\mathbb{O}}$ induction on the natural numbers, $(\Delta_0^{\mathbb{O}}\text{-}\mathsf{I_N})$, translates into complete induction for arbitrary $\mathcal{L}_1$ formulas. This concludes our sketch of the conservativity of OAD over PA. Thus, we have the following theorem.

Theorem 28 $|\mathsf{OAD}| \leq \varepsilon_0$.

The principal difference in the proof-theoretic treatment of $\mathsf{OAD} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ consists in the fact that the presence of complete induction on the natural numbers for arbitrary $\mathcal{L}_{\mathbb{O}}$ formulas does no longer enable us to establish a *finite* partial cut elimination theorem. To overcome this difficulty one follows the standard procedure and replaces $(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ by the ω rule at the prize of infinite derivation lengths. Accordingly, we can now obtain quasi cut-free derivations in (a Tait-style version of) OAD plus ω rule of length bounded by ε_0. Then we can proceed in a similar manner as above by using an asymmetric interpretation into a ramified system of ordinals describing the stages of inductive definitions, where only ordinals below ε_0 are needed. Finally, the *second cut elimination theorem* of predicative proof theory (cf. [110, 92]) holds for this ramified system so that ordinal levels and derivation lengths bounded by ε_0 give rise to *cut-free* derivations of length bounded by $\varphi\varepsilon_0 0$. We refer the reader to [67, 69, 119, 123] for similar arguments. In conclusion, we have the following theorem concerning the upper bound of our ordinal theory $\mathsf{OAD} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})$.

Theorem 29 $|\mathsf{OAD} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})| \leq \varphi\varepsilon_0 0$.

4.2 Upper bounds for OIN

It is the aim of this section to establish the upper proof-theoretic bounds Γ_0 and $\varphi 1 \varepsilon_0 0$ for the two systems OIN and $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})$, respectively. Since the treatment of OIN is rather standard and well-known from the analysis of similar formal systems, we will put some emphasis in the sequel on describing the analysis of $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I}_{\mathsf{N}})$.

Let us start with introducing the basic semiformal system H, which will be used in various ways in the rest of this chapter. Essentially, H is OAD without $\Sigma^{\mathbb{O}}$ reflection and with complete induction on the natural numbers replaced by the ω rule. Moreover, in H we no longer have *unbounded* ordinal quantifiers, and since H is semiformal with respect to the natural numbers, free number variables are not present.

The language $\mathcal{L}$ of H is obtained from $\mathcal{L}_{\mathbb{O}}$ by omitting free number variables and *unbounded* quantifiers over ordinals. In addition, we assume that $\mathcal{L}$ includes a new constant 0 for the least ordinal. Therefore, the ordinal terms of $\mathcal{L}$ are the constant 0 and the ordinal variables. We call two literals of $\mathcal{L}$ *numerically equivalent*, if they are syntactically identical modulo number subterms which have the same value.

H is formulated in a Tait-style manner for finite sets $\Gamma, \Lambda, \ldots$ (possibly with subscripts) of $\mathcal{L}$ formulas. If A is an $\mathcal{L}$ formula, then Γ, A is a shorthand for $\Gamma \cup \{A\}$, and similar for expressions of the form Γ, A, B. The axioms and rules of inference of H are now given as follows.

I. Axioms, group 1. For all finite sets Γ of $\mathcal{L}$ formulas, all numerically equivalent $\mathcal{L}$ literals A and B, and all true $\mathcal{L}_1$ literals C:

$$\Gamma,\, \neg A,\, B \quad \text{and} \quad \Gamma,\, C.$$

II. Axioms, group 2. For all finite sets Γ of $\mathcal{L}$ formulas, all literals $A(\sigma)$ of $\mathcal{L}$, all ordinal terms μ, ν of $\mathcal{L}$ and all (instances of) axioms B of OAD from the groups II, III and V:

$$\Gamma,\, 0 = \mu,\, 0 < \mu \quad \text{and} \quad \Gamma,\, \mu \neq \nu,\, \neg A(\mu),\, A(\nu) \quad \text{and} \quad \Gamma,\, B.$$

III. Propositional rules. These are the usual Tait-style rules for disjunction and conjunction.

IV. Number quantifier rules. For all finite sets Γ of $\mathcal{L}$ formulas and all $\mathcal{L}$ formulas $A(s)$:

$$\frac{\Gamma,\, A(s)}{\Gamma,\, (\exists x)A(x)}, \qquad \frac{\Gamma,\, A(t) \text{ for all closed number terms } t}{\Gamma,\, (\forall x)A(x)} \quad (\omega).$$

V. Ordinal quantifier rules. For all finite sets Γ of $\mathcal{L}$ formulas, all $\mathcal{L}$ formulas A, all ordinal terms μ, ν of $\mathcal{L}$ and all ordinal variables σ so that the usual variable conditions are satisfied:

$$\frac{\Gamma,\, \mu < \nu \wedge A(\mu)}{\Gamma,\, (\exists \xi < \nu)A(\xi)}, \qquad \frac{\Gamma,\, \sigma < \nu \rightarrow A(\sigma)}{\Gamma,\, (\forall \xi < \nu)A(\xi)}.$$

VI. Cut rules. For all finite sets Γ of $\mathcal{L}$ formulas and all $\mathcal{L}$ formulas A:

$$\frac{\Gamma,\, A \qquad \Gamma,\, \neg A}{\Gamma}.$$

The derivability relation $\mathsf{H} \stackrel{\alpha}{\vdash} \Gamma$ is used to express that the finite set Γ of $\mathcal{L}$ formulas has an H proof of depth less than or equal to α. Furthermore, we write $\mathsf{H} \stackrel{\alpha}{\vdash}_0 \Gamma$ if Γ has a cut-free proof in H of depth less than or equal to α. Moreover, we write $\mathsf{H} \stackrel{<\alpha}{\vdash} \Gamma$ and $\mathsf{H} \stackrel{<\alpha}{\vdash}_0 \Gamma$ if there exists a $\beta < \alpha$ such that $\mathsf{H} \stackrel{\beta}{\vdash} \Gamma$ and $\mathsf{H} \stackrel{\beta}{\vdash}_0 \Gamma$, respectively.

In the following we will be interested in extensions of H by axioms claiming the existence of α many admissible ordinals which are ordered in an increasing chain. The corresponding systems will be called $\mathsf{H}[\alpha]$. Later we will study the generalizations $\mathsf{H}[S, n, \alpha]$ of $\mathsf{H}[\alpha]$.

The language $\mathcal{L}[\alpha]$ of $\mathsf{H}[\alpha]$ extends the language $\mathcal{L}$ by *new* constants $\mathsf{c}[\beta]$ for each $\beta < \alpha$. The semiformal system $\mathsf{H}[\alpha]$ includes the axioms and rules of inference of H (extended to the language $\mathcal{L}[\alpha]$) plus the following axioms

$$\Gamma,\, \mathsf{Ad}(\mathsf{c}[\beta]) \quad \text{and} \quad \Gamma,\, \mathsf{c}[\gamma] < \mathsf{c}[\beta]$$

for all finite sets Γ of $\mathcal{L}[\alpha]$ formulas and all ordinals $\gamma < \beta < \alpha$. Finally, the deducibility relation $\mathsf{H}[\alpha] \overset{\beta}{\vdash} \Gamma$ is understood as above.

Next we turn to a standard preliminary reduction of OIN and $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$. In particular, the $\Sigma^{\mathbb{O}}$ fragment of OIN and $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ can be reduced to the systems $\mathsf{H}[n]$ for $n < \omega$ and $\mathsf{H}[\alpha]$ for $\alpha < \varepsilon_0$, respectively. This procedure is a rather standard combination of partial cut elimination and asymmetric interpretation, cf. e.g. the reduction of the set theory KPi^0 to finitely many admissibles in Jäger [57]. After eliminating $(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ by means of the ω rule and paying the well-known price with respect to derivation lengths, the reduction for $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ is analogous. Hence, we can state the following lemma without proof.

Lemma 30 *We have the following reductions:*

1. *Assume that the $\Sigma^{\mathbb{O}}$ sentence A is provable in* OIN. *Then there exists a natural number n so that $\mathsf{H}[n{+}1] \overset{<\varepsilon_0}{\vdash} A^{\mathsf{c}[n]}$.*

2. *Assume that the $\Sigma^{\mathbb{O}}$ sentence A is provable in* $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$. *Then there exists an ordinal α less than ε_0 so that $\mathsf{H}[\alpha{+}1] \overset{<\varepsilon_0}{\vdash} A^{\mathsf{c}[\alpha]}$.*

Moreover, the treatment of the systems $\mathsf{H}[n]$ for $n < \omega$ is very well understood, cf. e.g. [57] for a similar scenario. Indeed, one iterates the procedure for treating $\mathsf{OAD} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ (cf. the previous section) finitely often in order to eliminate finitely many admissible ordinals. The elimination of each admissible forces a further application of the binary φ function and, hence, the whole procedure yields ordinal bounds below the Feferman-Schütte ordinal Γ_0. Thus, we are able to state the following upper bound for OIN.

Theorem 31 $|\mathsf{OIN}| \leq \Gamma_0$.

Let us now turn to the analysis of the systems $\mathsf{H}[\alpha]$ for limit ordinals α, finally leading to the desired upper bound $\varphi 1 \varepsilon_0 0$ for the ordinal theory $\mathsf{OIN}{+}(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$.

Before stating the crucial lemma, let us introduce some notation and terminology. For a formula A we use the notation $A[\tau_1, \ldots, \tau_n]$ to express that all its free ordinal variables belong to the list $\tau_1, \ldots, \tau_n$; the analogous convention is employed for finite sets of formulas. Further, we call a finite set $\Gamma[\tau_1, \ldots, \tau_n]$ of $\mathcal{L}[\alpha]$ formulas *quasi closed* if there exist $\beta_1, \ldots, \beta_n < \alpha$ so that Γ is of the form

$$\tau_1 \not< \mathsf{c}[\beta_1],\ \ldots,\ \tau_n \not< \mathsf{c}[\beta_n],\ \Lambda[\tau_1, \ldots, \tau_n].$$

Hence, in a quasi closed set of $\mathcal{L}[\alpha]$ formulas all occurring free ordinal variables are bound by some ordinal constant $\mathsf{c}[\beta]$ with $\beta < \alpha$.

We are now in a position to formulate the following main lemma about the reduction of $\mathsf{H}[\beta+\omega^{1+\rho}]$. Its status is similar to the one of the second elimination theorem of predicative proof theory, cf. e.g. [92, 110].

Lemma 32 (Reduction of $\mathsf{H}[\beta+\omega^{1+\rho}]$) *Assume that Γ is a quasi closed set of $\mathcal{L}[\beta+\omega^{1+\rho}]$ formulas with the property that*

$$\mathsf{H}[\beta+\omega^{1+\rho}] \mid\!\frac{\alpha}{}\, \Gamma.$$

Then we have for all ordinals γ less than $\omega^{1+\rho}$ which are big enough for Γ being a quasi closed set of $\mathcal{L}[\beta+\gamma]$ formulas that

$$\mathsf{H}[\beta+\gamma] \mid\!\frac{\varphi 1\rho\alpha}{}\, \Gamma.$$

Proof. We prove the claim by main induction on ρ and side induction on α. We distinguish cases whether $\rho = 0$, ρ is a successor, or ρ is a limit ordinal.

$\rho = 0$. Let us assume that Γ is a finite and quasi closed set of $\mathcal{L}[\beta+k]$ formulas for some natural number k so that $\mathsf{H}[\beta+\omega] \mid\!\frac{\alpha}{}\, \Gamma$. If Γ is an axiom of $\mathsf{H}[\beta+\omega]$, then the claim is trivial. Furthermore, if Γ is the conclusion of a rule different from the cut rule, the claim is immediate from the induction hypothesis. Hence, the only critical case comes up if Γ is the conclusion of a cut rule. Then there exist a natural number $l \geq k$, $\alpha_0, \alpha_1 < \alpha$ and an $\mathcal{L}[\beta+l]$ formula A so that

(1) $$\mathsf{H}[\beta+\omega] \mid\!\frac{\alpha_0}{}\, \Gamma, A \quad \text{and} \quad \mathsf{H}[\beta+\omega] \mid\!\frac{\alpha_1}{}\, \Gamma, \neg A.$$

Let A_Γ be the formula which results from A by replacing all free ordinal variables of A which do not occur in Γ by the ordinal constant 0. Then we also have

(2) $$\mathsf{H}[\beta+\omega] \mid\!\frac{\alpha_0}{}\, \Gamma, A_\Gamma \quad \text{and} \quad \mathsf{H}[\beta+\omega] \mid\!\frac{\alpha_1}{}\, \Gamma, \neg A_\Gamma.$$

By the induction hypothesis we can conclude that

(3) $$\mathsf{H}[\beta+l] \mid\!\frac{\varphi 10\alpha_0}{}\, \Gamma, A_\Gamma \quad \text{and} \quad \mathsf{H}[\beta+l] \mid\!\frac{\varphi 10\alpha_1}{}\, \Gamma, \neg A_\Gamma.$$

Hence, by applying cut we yield $\mathsf{H}[\beta+l] \mid\!\frac{\delta}{}\, \Gamma$, where δ denotes the ordinal $\max(\varphi 10\alpha_0, \varphi 10\alpha_1)+1$. Finally, we obtain by the standard elimination procedure of finitely many admissibles (cf. our discussion above) that

(4) $$\mathsf{H}[\beta+k] \mid\!\frac{\varphi 10\alpha}{}\, \Gamma.$$

This concludes our proof in the case $\rho = 0$.

$\rho = \rho_0 + 1$. Let $\gamma < \omega^{1+\rho_0+1}$ and Γ be a finite and quasi closed set of $\mathcal{L}[\beta{+}\gamma]$ formulas so that

$$\mathsf{H}[\beta{+}\omega^{1+\rho_0+1}] \overset{\alpha}{\vdash} \Gamma. \tag{5}$$

Note that $\gamma = \omega^{1+\rho_0} \cdot k + \gamma'$ for some natural number k and some γ' less than $\omega^{1+\rho_0}$. Again the only crucial case occurs if Γ is the conclusion of a cut. Then there exist a natural number $l > k$, $\alpha_0, \alpha_1 < \alpha$ and an $\mathcal{L}[\beta{+}\omega^{1+\rho_0} \cdot l]$ formula A so that

$$\mathsf{H}[\beta{+}\omega^{1+\rho_0+1}] \overset{\alpha_0}{\vdash} \Gamma, A_\Gamma \quad \text{and} \quad \mathsf{H}[\beta{+}\omega^{1+\rho_0+1}] \overset{\alpha_1}{\vdash} \Gamma, \neg A_\Gamma, \tag{6}$$

where A_Γ is defined as before. By applying the side induction hypothesis to the two assertions in (6) we derive

$$\mathsf{H}[\beta{+}\omega^{1+\rho_0} \cdot l] \overset{\varphi 1 \rho \alpha_0}{\vdash} \Gamma, A_\Gamma \quad \text{and} \quad \mathsf{H}[\beta{+}\omega^{1+\rho_0} \cdot l] \overset{\varphi 1 \rho \alpha_1}{\vdash} \Gamma, \neg A_\Gamma, \tag{7}$$

and, hence, we also have $\mathsf{H}[\beta{+}\omega^{1+\rho_0} \cdot l] \overset{\delta}{\vdash} \Gamma$, for δ being the ordinal number $\max(\varphi 1 \rho \alpha_0, \varphi 1 \rho \alpha_1) + 1$. If we inductively define a sequence of ordinals δ_i by $\delta_0 := \delta$ and $\delta_{i+1} := \varphi 1 \rho_0 \delta_i$, then by applying the main induction hypothesis $l - k$ times one readily obtains:

$$\begin{aligned} &\mathsf{H}[\beta{+}\omega^{1+\rho_0} \cdot (l{-}1)] \overset{\delta_1}{\vdash} \Gamma, \\ &\qquad\vdots \\ &\mathsf{H}[\beta{+}\omega^{1+\rho_0} \cdot (k{+}1)] \overset{\delta_{l-k-1}}{\vdash} \Gamma, \\ &\mathsf{H}[\beta{+}\omega^{1+\rho_0} \cdot k{+}\gamma'] \overset{\delta_{l-k}}{\vdash} \Gamma. \end{aligned} \tag{8}$$

Here we have successively replaced β by

$$\beta + \omega^{1+\rho_0} \cdot (l - 1), \quad \ldots, \quad \beta + \omega^{1+\rho_0} \cdot (k + 1), \quad \beta + \omega^{1+\rho_0} \cdot k$$

in the main induction hypothesis. Since $\delta_{l-k} < \varphi 1 \rho \alpha$, we have indeed established the desired assertion $\mathsf{H}[\beta{+}\gamma] \overset{\varphi 1 \rho \alpha}{\vdash} \Gamma$.

ρ *limit*. Assume that $\gamma < \omega^{1+\rho}$ and Γ is a finite and quasi closed set of $\mathcal{L}[\beta{+}\gamma]$ formulas so that

$$\mathsf{H}[\beta{+}\omega^{\rho}] \overset{\alpha}{\vdash} \Gamma. \tag{9}$$

Again assume that Γ is the conclusion of cut. Then there exists $\rho_0 < \rho$ with $\gamma \leq \omega^{1+\rho_0}$, $\alpha_0, \alpha_1 < \alpha$ and an $\mathcal{L}[\beta+\omega^{1+\rho_0}]$ formula A so that

$$\text{(10)} \qquad \mathsf{H}[\beta+\omega^{\rho}] \mid\!\frac{\alpha_0}{\quad} \Gamma, A_\Gamma \quad \text{and} \quad \mathsf{H}[\beta+\omega^{\rho}] \mid\!\frac{\alpha_1}{\quad} \Gamma, \neg A_\Gamma.$$

The side induction hypothesis applied to (10) produces

$$\text{(11)} \qquad \mathsf{H}[\beta+\omega^{1+\rho_0}] \mid\!\frac{\varphi 1\rho\alpha_0}{\quad} \Gamma, A_\Gamma \quad \text{and} \quad \mathsf{H}[\beta+\omega^{1+\rho_0}] \mid\!\frac{\varphi 1\rho\alpha_1}{\quad} \Gamma, \neg A_\Gamma,$$

and, hence, we also have $\mathsf{H}[\beta+\omega^{1+\rho_0}] \mid\!\frac{\delta}{\quad} \Gamma$, for δ being the ordinal number $\max(\varphi 1\rho\alpha_0, \varphi 1\rho\alpha_1)+1$. From this, we conclude by the main induction hypothesis $\mathsf{H}[\beta+\gamma] \mid\!\frac{\varphi 1\rho_0\delta}{\quad} \Gamma$. Since $\varphi 1\rho_0\delta < \varphi 1\rho\alpha$, this is our claim. This finishes the discussion of the limit case and also the verification of the lemma. □

We are now in a position to combine our main lemma with Lemma 30 in order to obtain the desired upper bound for the system $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$.

Theorem 33 $|\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})| \leq \varphi 1\varepsilon_0 0$.

Proof. Let A be an arbitrary sentence of $\mathcal{L}_1$ and assume that A is derivable in $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$. By Lemma 30 there exists an ordinal α less than ε_0 so that $\mathsf{H}[\omega^{1+\alpha}] \mid\!\frac{\alpha}{\quad} A$. Now we can apply our previous lemma with $\beta = \gamma = 0$ and $\rho = \alpha$ and obtain $\mathsf{H}[0] \mid\!\frac{\varphi 1\alpha\alpha}{\quad} A$. Observe that the system $\mathsf{H}[0]$ does not contain constants for admissible ordinals and, hence, is identical to H. Thus, we finally obtain by predicative cut elimination for H that $\mathsf{H} \mid\!\frac{<\varphi 1\varepsilon_0 0}{0} A$.

The desired upper bound for the proof-theoretic ordinal of $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ now follows by choosing the formula $\mathsf{TI}(\sqsubset, \mathsf{Q})$ for A and using the well-known boundedness argument for cut-free derivations of $\mathsf{TI}(\sqsubset, \mathsf{Q})$, cf. [7, 92, 110]. □

4.3 Upper bounds for OMA

We finally turn to the proof-theoretic treatment of the ordinal theories OMA as well as $\mathsf{OMA} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$. The plan is to establish $|\mathsf{OMA}| \leq \varphi\omega 00$ in detail and only indicate how the relevant arguments can be generalized to obtain a treatment of $\mathsf{OMA} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$.

The analysis of OMA proceeds in two steps. First, the schema of $\Pi_2^{\mathbb{O}}$ reflection on admissible ordinals, $(\Pi_2^{\mathbb{O}}\text{-}\mathsf{Ref}\text{-}\mathsf{Ad})$, is eliminated in favor of n-inaccessible ordinals for sufficiently large n less than ω. In a second step, we will treat auxiliary systems claiming the existence of α many n-inaccessible ordinals. The latter calculi are the natural generalizations of the systems $\mathsf{H}[\alpha]$ studied in the previous section.

In order to be able to carry through the first reduction step, we need to reformulate OMA in Tait-style manner. The Tait calculus OMA^T is defined in the obvious way and contains the following axioms and rules of inference.

I. Axioms. For all finite sets Γ of $\mathcal{L}_\mathbb{O}$ formulas, all $\Delta_0^\mathbb{O}$ formulas A and all $\Delta_0^\mathbb{O}$ formulas B which are axioms of OMA:

$$\Gamma,\, \neg A,\, A \quad \text{and} \quad \Gamma,\, B.$$

II. Propositional and quantifier rules. These include the usual Tait-style inference rules for the propositional connectives and all sorts of quantifiers.

III. $\Sigma^\mathbb{O}$ reflection rules. For all finite sets Γ of $\mathcal{L}_\mathbb{O}$ formulas and for all $\Sigma^\mathbb{O}$ formulas A:

$$\frac{\Gamma,\, A}{\Gamma,\, (\exists \xi) A^\xi}.$$

IV. $\Pi_2^\mathbb{O}$ refection on Ad rules. For all finite sets Γ of $\mathcal{L}_\mathbb{O}$ formulas and for all $\Delta_0^\mathbb{O}$ formulas $A(\xi,\eta,\vec{\tau})$ whose free ordinal variables are from the list $\xi,\eta,\vec{\tau}$:

$$\frac{\Gamma,\, (\forall \xi)(\exists \eta) A(\xi,\eta,\vec{\tau})}{\Gamma,\, (\exists \sigma)[\mathsf{Ad}(\sigma) \,\wedge\, \vec{\tau} < \sigma \,\wedge\, (\forall \xi < \sigma)(\exists \eta < \sigma) A(\xi,\eta,\vec{\tau})]}.$$

V. Cut rules. For all finite sets Γ of $\mathcal{L}_\mathbb{O}$ formulas and all $\mathcal{L}_\mathbb{O}$ formulas A:

$$\frac{\Gamma,\, A \qquad \Gamma,\, \neg A}{\Gamma}.$$

The notion $\mathsf{OMA}^\mathsf{T} \mathrel{|\!\overset{n}{=}} \Gamma$ is used to express that the set Γ is provable in OMA^T by a proof of depth less than or equal to n; we write $\mathsf{OMA}^\mathsf{T} \mathrel{|\!\overset{n}{\underset{\star}{=}}} \Gamma$ if Γ is provable in OMA^T by a proof of depth less than or equal to n so that all its cut formulas are $\Sigma_1^\mathbb{O}$ or $\Pi_1^\mathbb{O}$ formulas. In addition, $\mathsf{OMA}^\mathsf{T} \vdash \Gamma$ or $\mathsf{OMA}^\mathsf{T} \vdash_\star \Gamma$ means that there exists a natural number n so that $\mathsf{OMA}^\mathsf{T} \mathrel{|\!\overset{n}{=}} \Gamma$ or $\mathsf{OMA}^\mathsf{T} \mathrel{|\!\overset{n}{\underset{\star}{=}}} \Gamma$, respectively.

One readily notes that the main formulas of all axioms and rules of OMA^T are $\Sigma_1^\mathbb{O}$ formulas. As a consequence, we obtain the following weak cut elimination lemma for OMA^T.

Lemma 34 (Weak cut elimination) *We have for all finite sets* Γ *of* $\mathcal{L}_{\mathbb{O}}$ *formulas that*

$$\mathsf{OMA}^{\mathsf{T}} \vdash \Gamma \quad \Longrightarrow \quad \mathsf{OMA}^{\mathsf{T}} \mid\!\!\frac{}{\star}\, \Gamma.$$

Of course, the axioms and rules of $\mathsf{OMA}^{\mathsf{T}}$ are tailored so that OMA can be embedded into $\mathsf{OMA}^{\mathsf{T}}$ in a straightforward manner. Thus we obtain the following corollary.

Corollary 35 *If the* $\mathcal{L}_{\mathbb{O}}$ *formula* A *is provable in* OMA*, then there exists a natural number* n *so that* $\mathsf{OMA}^{\mathsf{T}} \mid\!\!\frac{n}{\star}\, A$*.*

For the reduction of $\mathsf{OMA}^{\mathsf{T}}$ below, the notion of n-inaccessibility is crucial. By recursion on $n < \omega$ we define a formula $\mathsf{Ia}_n(\sigma)$ to express that σ is an n-inaccessible ordinal as follows:

$$\begin{aligned} \mathsf{Ia}_0(\sigma) &:= \mathsf{Ad}(\sigma), \\ \mathsf{Ia}_{n+1}(\sigma) &:= \mathsf{Ad}(\sigma) \wedge (\forall \xi < \sigma)(\exists \eta < \sigma)[\xi < \eta \wedge \mathsf{Ia}_n(\eta)]. \end{aligned}$$

We observe that each formula $\mathsf{Ia}_n(\sigma)$ is a $\Delta_0^{\mathbb{O}}$ formula without free number variables and therefore also an $\mathcal{L}$ formula.

Some further terminology is needed before we turn to the reduction of $\mathsf{OMA}^{\mathsf{T}}$ to n-inaccessibility. If $\vec{\tau}$ is the sequence of ordinal variables $\tau_1, \ldots, \tau_m$, then $(\vec{\tau} \not< \sigma)$ stands for the set of literals

$$\{\tau_1 \not< \sigma,\, \ldots,\, \tau_m \not< \sigma\}.$$

A finite set of $\mathcal{L}$ formulas Λ is called an *instance* of the finite set of $\mathcal{L}_{\mathbb{O}}$ formulas Γ if it results from Γ by replacing all free number variables of formulas in Γ by closed number terms of $\mathcal{L}_1$. Finally, we write Γ^σ for the finite set of formulas which is obtained from Γ by replacing each formula A in Γ by its restriction A^σ.

Lemma 36 (Reduction of $\mathsf{OMA}^{\mathsf{T}}$) *Assume that* $\Gamma[\vec{\tau}]$ *is a finite set of* $\Sigma^{\mathbb{O}}$ *formulas of* $\mathcal{L}_{\mathbb{O}}$*. Then we have for all instances* $\Lambda[\vec{\tau}]$ *of* $\Gamma[\vec{\tau}]$ *and all natural numbers* n *that*

$$\mathsf{OMA}^{\mathsf{T}} \mid\!\!\frac{n}{\star}\, \Gamma[\vec{\tau}] \quad \Longrightarrow \quad \mathsf{H} \mid\!\!\frac{\omega(n+2)}{}\, \neg \mathsf{Ia}_n(\sigma),\, (\vec{\tau} \not< \sigma),\, \Lambda^\sigma[\vec{\tau}].$$

Proof. This lemma is proved by induction on n. In the following we exemplarily treat the cases of cut and $\Pi_2^{\mathbb{O}}$ reflection on admissible ordinals.

We note that complete induction on the natural numbers is dealt with as usual by making use of the ω rule. In all other cases the claim is immediate from the induction hypothesis.

Let us first look at the case where $\Gamma[\vec{\tau}]$ is the conclusion of a cut. Then there are natural numbers $n_0, n_1 < n$ and a $\Delta_0^{\mathbb{O}}$ formula $A[\xi, \vec{\eta}]$ so that

$$(1) \qquad \mathsf{OMA}^{\mathsf{T}} \mathrel{\overset{n_0}{\underset{\star}{\vdash}}} \Gamma[\vec{\tau}],\ (\exists \xi) A[\xi, \vec{\eta}] \quad \text{and} \quad \mathsf{OMA}^{\mathsf{T}} \mathrel{\overset{n_1}{\underset{\star}{\vdash}}} \Gamma[\vec{\tau}],\ (\forall \xi) \neg A[\xi, \vec{\eta}].$$

Suppose that $\Lambda[\vec{\tau}],\ (\exists \xi) B[\xi, \vec{\eta}]$ is an instance of $\Gamma[\vec{\tau}],\ (\exists \xi) A[\xi, \vec{\eta}]$. Then inversion applied to the second premise and the induction hypothesis yield

$$(2) \qquad \mathsf{H} \mathrel{\overset{\omega(n_0+2)}{\vdash}} \neg \mathsf{Ia}_{n_0}(\sigma),\ (\vec{\tau}, \vec{\eta} \not< \sigma),\ \Lambda^{\sigma}[\vec{\tau}],\ (\exists \xi < \sigma) B[\xi, \vec{\eta}],$$

$$(3) \qquad \mathsf{H} \mathrel{\overset{\omega(n_1+2)}{\vdash}} \neg \mathsf{Ia}_{n_1}(\sigma),\ (\vec{\tau}, \vec{\eta}, \eta_0 \not< \sigma),\ \Lambda^{\sigma}[\vec{\tau}],\ \neg B[\eta_0, \vec{\eta}],$$

where η_0 is a fresh ordinal variable. From (2) and (3) we obtain

$$(4) \qquad \mathsf{H} \mathrel{\overset{<\omega(n+2)}{\vdash}} \neg \mathsf{Ia}_{n}(\sigma),\ (\vec{\tau}, \vec{\eta} \not< \sigma),\ \Lambda^{\sigma}[\vec{\tau}],\ (\exists \xi < \sigma) B[\xi, \vec{\eta}],$$

$$(5) \qquad \mathsf{H} \mathrel{\overset{<\omega(n+2)}{\vdash}} \neg \mathsf{Ia}_{n}(\sigma),\ (\vec{\tau}, \vec{\eta} \not< \sigma),\ \Lambda^{\sigma}[\vec{\tau}],\ (\forall \xi < \sigma) \neg B[\xi, \vec{\eta}].$$

Here we have used the obvious fact that $\mathsf{H} \mathrel{\overset{<\omega}{\vdash}} \neg \mathsf{Ia}_n(\sigma),\ \mathsf{Ia}_k(\sigma)$ for each natural number k less than n. A cut applied to (4) and (5) reveals

$$(6) \qquad \mathsf{H} \mathrel{\overset{\omega(n+2)}{\vdash}} \neg \mathsf{Ia}_{n}(\sigma),\ (\vec{\tau} \not< \sigma),\ \Lambda^{\sigma}[\vec{\tau}]$$

since superfluous ordinal variables can be easily eliminated. This is as desired and completes the treatment of the cut rule.

Let us now turn to the heart of the reduction, namely the interpretation of $\Pi_2^{\mathbb{O}}$ reflection on admissible ordinals. Assume that $\Gamma[\vec{\tau}]$ is the conclusion of the corresponding rule of $\mathsf{OMA}^{\mathsf{T}}$. Hence, there exist an $n_0 < n$ and a $\Delta_0^{\mathbb{O}}$ formula $A[\xi, \eta, \vec{\tau}]$ so that

$$(7) \qquad \mathsf{OMA}^{\mathsf{T}} \mathrel{\overset{n_0}{\underset{\star}{\vdash}}} \Gamma[\vec{\tau}],\ (\forall \xi)(\exists \eta) A[\xi, \eta, \vec{\tau}].$$

An application of inversion to (7) with a fresh ordinal variable τ_0 forces

$$(8) \qquad \mathsf{OMA}^{\mathsf{T}} \mathrel{\overset{n_0}{\underset{\star}{\vdash}}} \Gamma[\vec{\tau}],\ (\exists \eta) A[\tau_0, \eta, \vec{\tau}].$$

Next we choose an instance $\Lambda[\vec{\tau}],\ (\exists \eta) B[\tau_0, \eta, \vec{\tau}]$ of $\Gamma[\vec{\tau}],\ (\exists \eta) A[\tau_0, \eta, \vec{\tau}]$ and apply the induction hypothesis to (8) in order to obtain

$$(9) \qquad \mathsf{H} \mathrel{\overset{\omega(n_0+2)}{\vdash}} \neg \mathsf{Ia}_{n_0}(\sigma_0),\ (\tau_0, \vec{\tau} \not< \sigma_0),\ \Lambda^{\sigma_0}[\vec{\tau}],\ (\exists \eta < \sigma_0) B[\tau_0, \eta, \vec{\tau}].$$

But from (9) we can immediately derive by bounded universal ordinal quantification that

$$\text{(10)}\quad \mathsf{H} \overset{<\omega(n+2)}{\vdash} \neg\mathsf{Ia}_{n_0}(\sigma_0),\, (\vec{\tau} \not< \sigma_0),\, \Lambda^{\sigma_0}[\vec{\tau}],\, (\forall\xi < \sigma_0)(\exists\eta < \sigma_0)B[\xi,\eta,\vec{\tau}].$$

Moreover, it is an easy task to check that we also have

$$\text{(11)}\quad \mathsf{H} \overset{<\omega}{\vdash} \neg\mathsf{Ia}_n(\sigma),\, (\vec{\tau} \not< \sigma),\, (\exists\sigma_0 < \sigma)[\mathsf{Ia}_{n_0}(\sigma_0) \wedge \vec{\tau} < \sigma_0].$$

By combining (10) and (11) and applying persistency, we can finally derive

$$\text{(12)}\quad \mathsf{H} \overset{\omega(n+2)}{\vdash} \neg\mathsf{Ia}_n(\sigma),\, (\vec{\tau} \not< \sigma),\, \Lambda^{\sigma}[\vec{\tau}],\, C[\sigma,\vec{\tau}]$$

for $C[\sigma,\vec{\tau}]$ denoting the formula

$$(\exists\sigma_0 < \sigma)[\mathsf{Ad}(\sigma_0) \wedge \vec{\tau} < \sigma_0 \wedge (\forall\xi < \sigma_0)(\exists\eta < \sigma_0)B[\xi,\eta,\vec{\tau}]].$$

Since $C[\sigma,\vec{\tau}]$ is in fact an element of $\Lambda^{\sigma}[\vec{\tau}]$ we have indeed established that $\mathsf{H} \overset{\omega(n+2)}{\vdash} \neg\mathsf{Ia}_n(\sigma),\, (\vec{\tau} \not< \sigma),\, \Lambda^{\sigma}[\vec{\tau}]$ as desired. Observe that we have made crucial use of the fact that $\Lambda[\vec{\tau}]$ contains $\Sigma^{\mathbb{O}}$ formulas only in order to be able to apply persistency to $\Lambda^{\sigma_0}[\vec{\tau}]$. All together this completes the reduction of $\mathsf{OMA}^{\mathsf{T}}$ to n-inaccessibility. □

Let $\mathcal{L}'$ be some extension of $\mathcal{L}$ by additional constants for ordinals and let S be a (finite or infinite) set of $\mathcal{L}'$ formulas. The final part of this section is devoted to the proof-theoretic analysis of semiformal systems $\mathsf{H}[S,n,\alpha]$ for each $n < \omega$ and each ordinal α. The crucial axioms of $\mathsf{H}[S,n,\alpha]$ claim: (i) all formulas of S; (ii) the existence of α many n-inaccessible ordinals which are ordered in an increasing chain and are greater than all the ordinal constants occurring in S. Thus, the calculi $\mathsf{H}[S,n,\alpha]$ are generalizations of the systems $\mathsf{H}[\alpha]$ studied in the previous section. In particular, $\mathsf{H}[\alpha]$ corresponds to $\mathsf{H}[S,0,\alpha]$ for S being the empty set.

The language $\mathcal{L}[S,n,\alpha]$ of $\mathsf{H}[S,n,\alpha]$ is the extension of the language $\mathcal{L}$ generated by the constants occurring in S plus additional *new* constants $\mathsf{c}[S,n,\beta]$ for each $\beta < \alpha$. The semiformal system $\mathsf{H}[S,n,\alpha]$ includes the axioms and rules of inference of H (extended to the language $\mathcal{L}[S,n,\alpha]$) plus the following axioms

(i) $\Gamma,\, A$ and $\Gamma,\, \mathsf{d} < \mathsf{c}[S,n,0]$,

(ii) $\Gamma,\, \mathsf{Ia}_n(\mathsf{c}[S,n,\beta])$ and $\Gamma,\, \mathsf{c}[S,n,\gamma] < \mathsf{c}[S,n,\beta]$, for all finite sets Γ of $\mathcal{L}[S,n,\alpha]$ formulas, all elements A of S, all ordinal constants d from S, and all ordinals $\gamma < \beta < \alpha$. The deducibility relation $\mathsf{H}[S,n,\alpha] \mathrel{|\!\!\underline{\beta}} \Gamma$ is understood in the obvious manner. Finally, the notion of a quasi-closed set of $\mathcal{L}[S,n,\alpha]$ formulas is defined analogously as for $\mathcal{L}[\alpha]$.

The following lemma is the natural generalization of Lemma 32 to the context of n-inaccessible ordinals.

Lemma 37 (Reduction of $\mathsf{H}[S,n,\beta{+}\omega^{1+\rho}]$) *Let Γ a quasi closed set of $\mathcal{L}[S,n,\beta{+}\omega^{1+\rho}]$ formulas with the property that*

$$\mathsf{H}[S,n,\beta{+}\omega^{1+\rho}] \mathrel{|\!\!\underline{\alpha}} \Gamma.$$

Then we have for all ordinals γ less than $\omega^{1+\rho}$ which are big enough for Γ being a quasi closed set of $\mathcal{L}[S,n,\beta{+}\gamma]$ formulas that

$$\mathsf{H}[S,n,\beta{+}\gamma] \mathrel{|\!\!\underline{\varphi(n+1)\rho\alpha}} \Gamma.$$

Proof. This lemma is proved by induction on $n < \omega$. The claim in case of $n = 0$ is (apart from the presence of S) exactly our previous Lemma 32. For the induction step we assume that our lemma holds for some natural number n. Then we show our claim for $n{+}1$ by main induction on ρ and side induction on α. The main steps of the argument are the same as those in the proof of Lemma 32 in case ρ is a successor or limit ordinal. Therefore, in the following we confine ourselves to discussing the case $\rho = 0$ only.

Hence, assume that Γ is a finite and quasi closed set of $\mathcal{L}[S,n{+}1,\beta{+}k]$ formulas for some natural number k so that $\mathsf{H}[S,n{+}1,\beta{+}\omega] \mathrel{|\!\!\underline{\alpha}} \Gamma$. The only critical case is if Γ is the conclusion of a cut rule. Then there exist a natural number $l \geq k$, $\alpha_0, \alpha_1 < \alpha$ and an $\mathcal{L}[S,n{+}1,\beta{+}l]$ formula A so that

(1) $\quad \mathsf{H}[S,n{+}1,\beta{+}\omega] \mathrel{|\!\!\underline{\alpha_0}} \Gamma, A \quad$ and $\quad \mathsf{H}[S,n{+}1,\beta{+}\omega] \mathrel{|\!\!\underline{\alpha_1}} \Gamma, \neg A.$

Let A_Γ be the formula which results from A by replacing all free ordinal variables of A which do not occur in Γ by the ordinal constant 0. Then we also have

(2) $\quad \mathsf{H}[S,n{+}1,\beta{+}\omega] \mathrel{|\!\!\underline{\alpha_0}} \Gamma, A_\Gamma \quad$ and $\quad \mathsf{H}[S,n{+}1,\beta{+}\omega] \mathrel{|\!\!\underline{\alpha_1}} \Gamma, \neg A_\Gamma.$

By the induction hypothesis we infer

(3) $$\mathsf{H}[S,n{+}1,\beta{+}l] \mathrel{|\!\!\underline{\varphi(n+2)0\alpha_0}} \Gamma, A_\Gamma,$$

(4) $$\mathsf{H}[S,n{+}1,\beta{+}l] \mathrel{|\!\!\underline{\varphi(n+2)0\alpha_1}} \Gamma, \neg A_\Gamma.$$

Hence, we obtain by a cut

$$(5) \qquad \mathsf{H}[S, n{+}1, \beta{+}l] \vdash^{\delta} \Gamma,$$

where δ denotes the ordinal $\max(\varphi(n{+}2)0\alpha_0, \varphi(n{+}2)0\alpha_1){+}1$. If $l = k$ then we are done; therefore, let us assume that $l = l'{+}1 > k$. In order to get rid of the $(n{+}1)$-inaccessible $\mathsf{c}[S, n{+}1, l']$ one uses standard partial cut elimination and asymmetric interpretation (cf. also Lemma 30) in order to show

$$(6) \qquad \mathsf{H}[S', n, \delta^+] \vdash^{\delta^+} \Gamma,$$

where S' denotes the set of axioms of the system $\mathsf{H}[S, n{+}1, \beta{+}l']$ and δ^+ is the least ε number greater than δ. We know by induction hypothesis that the claim of our theorem is true for n and, hence, we can conclude from (6),

$$(7) \qquad \mathsf{H}[S', n, 0] \vdash^{\varphi(n+1)\delta^+\delta^+} \Gamma$$

since $\Gamma \subset \mathcal{L}[S', n, 0]$ by hypothesis. But in fact the system $\mathsf{H}[S', n, 0]$ is just $\mathsf{H}[S, n{+}1, \beta{+}l']$ and, moreover, we have that $\varphi(n{+}1)\delta^+\delta^+ < \varphi(n{+}2)0\alpha$. Thus (7) immediately reveals

$$(8) \qquad \mathsf{H}[S, n{+}1, \beta{+}l'] \vdash^{<\varphi(n+2)0\alpha} \Gamma.$$

Repeating this whole step $l - k$ times enables us to get rid of finitely many $(n{+}1)$-inaccessibles and we finally obtain

$$(9) \qquad \mathsf{H}[S, n{+}1, \beta{+}k] \vdash^{\varphi(n+2)0\alpha} \Gamma.$$

This concludes our proof in the case $\rho = 0$. $\square$

It remains to combine our previous results in order to get the desired upper bound $\varphi\omega00$ for OMA. In the following we simply write $\mathsf{H}[n, \alpha]$ instead of $\mathsf{H}[S, n, \alpha]$ if S is the empty set. Similarly $\mathsf{c}[n, \alpha]$ stands for $\mathsf{c}[S, n, \alpha]$ with empty S.

Theorem 38 $|\mathsf{OMA}| \leq \varphi\omega00$.

Proof. Assume that the $\mathcal{L}_1$ sentence A is provable in OMA. By Corollary 35 there exists a natural number n so that $\mathsf{OMA}^{\mathsf{T}} \vdash^{n}_{\star} A$. This enables us to invoke the reduction lemma for $\mathsf{OMA}^{\mathsf{T}}$, i.e. Lemma 36, in order to derive

$$(1) \qquad \mathsf{H} \vdash^{<\omega^2} \neg\mathsf{Ia}_n(\sigma),\ A.$$

A substitution of the ordinal constant $\mathsf{c}[n,0]$ for the ordinal variable σ followed by a cut on the formula $\mathsf{Ia}_n(\mathsf{c}[n,0])$ reveals

$$\mathsf{H}[n,\omega] \mathrel{|}\!\!\frac{<\omega^2}{}\; A. \tag{2}$$

Now we are in a position to apply Lemma 37 with S the empty set and $\beta = \gamma = \rho = 0$ and obtain

$$\mathsf{H}[n,0] \mathrel{|}\!\!\frac{<\varphi(n+1)0\omega^2}{}\; A. \tag{3}$$

The theory $\mathsf{H}[n,0]$ does not contain constants for n-inaccessibles and, hence, is identical to H. Finally, by standard predicative cut elimination for H and observing that $\varphi(n{+}1)0\omega^2 < \varphi\omega 00$ we conclude

$$\mathsf{H} \mathrel{|}\!\!\frac{<\varphi\omega 00}{0}\; A. \tag{4}$$

The desired upper bound with respect to the proof-theoretic ordinal of OMA now follows by the standard argument as explained at the end of the proof of Theorem 33. $\Box$

The methods applied in this section indeed also provide sharp bounds for the ordinal theory $\mathsf{OMA}+(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$. The pattern is as follows. First, $\mathsf{OMA}+(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ is embedded into OMA^T plus ω rule, thus getting rid of full complete induction $(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$ in favor of infinite derivation lengths. Weak cut elimination for OMA^T plus ω rule is proved as before, but because of the infinite derivations we now have

$$\mathsf{OMA} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N}) \vdash A \quad \Longrightarrow \quad \mathsf{OMA}^\mathsf{T} + (\omega) \mathrel{|}\!\!\frac{<\varepsilon_0}{\star}\; A$$

for each $\mathcal{L}_{\mathbb{O}}$ sentence A. From now on we can proceed as before, but always working with families $(\mathsf{Ia}_\alpha(\sigma) : \alpha < \varepsilon_0)$ instead of families $(\mathsf{Ia}_n(\sigma) : n < \omega)$. Carrying through everything in detail finally gives the following result about the upper proof-theoretic bound of $\mathsf{OMA} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$.

Theorem 39 $|\mathsf{OMA} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})| \leq \varphi\varepsilon_0 00$.

4.4 Putting the pieces together

We are now in the position to piece together all the results concerning the strength of the systems of explicit mathematics ETJ, EIN, and EMA, as well as the ordinal theories OAD, OIN, and OMA, all systems possibly augmented by the schema of complete induction on the natural numbers $(\mathbb{L}\text{-}\mathsf{I_N})$ and $(\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$, respectively.

The following main theorem is an immediate consequence of Theorem 5, Theorem 6, Theorem 8, Theorem 13, Corollary 22, Theorem 23, Theorem 25, Theorem 26, Theorem 27, Theorem 28, Theorem 29, Theorem 31, Theorem 33, Theorem 38, and Theorem 39.

Theorem 40 *We have the following proof-theoretic ordinals:*

1. $|\mathsf{ETJ}| = |\mathsf{OAD}| = \varepsilon_0$.
2. $|\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})| = |\mathsf{OAD} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})| = \varphi\varepsilon_0 0$.
3. $|\mathsf{EIN}| = |\mathsf{OIN}| = \Gamma_0$.
4. $|\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})| = |\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})| = \varphi 1 \varepsilon_0 0$.
5. $|\mathsf{EMA}| = |\mathsf{OMA}| = \varphi\omega 00$.
6. $|\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I_N})| = |\mathsf{OMA} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})| = \varphi\varepsilon_0 00$.

This concludes our proof-theoretic analysis of the various systems of explicit mathematics as well as theories with ordinals. Let us mention once more that our wellordering proofs and upper bound computations made use of techniques and tools of predicative proof theory only; hence, all our systems clearly belong to the world of *metapredicativity*.

Chapter 5

Related systems

The final chapter of the first part of this habilitation thesis is devoted to an informal discussion of various systems which are closely related to the systems of explicit mathematics and, hence, theories with ordinals which we have been examining in the previous chapters.

We will divide our overview into systems which are related to ETJ, EIN, and EMA, respectively, all possibly augmented by the full induction schema $(\mathbb{L}\text{-}\mathsf{I_N})$. In our discussion we will consider (i) subsystems of second order arithmetic, (ii) admissible set theories without foundation, (iii) fixed point theories, (iv) Martin-Löf type theories, and (v) Frege structures. Let us mention that our list of theories below is neither systematic nor complete.

5.1 Systems of strength ETJ

Let us briefly mention some systems which are proof-theoretically equivalent to ETJ and $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})$. Of course, there is a huge variety of such systems and, therefore, in the sequel we concentrate on theories which are somehow close in spirit to ETJ and $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})$.

The standard system of second order arithmetic $\Sigma^1_1\text{-}\mathsf{AC}_0$ based on the Σ^1_1 *axiom of choice*,

$$(\Sigma^1_1\text{-}\mathsf{AC}) \qquad (\forall x)(\exists Y)A(x,Y) \rightarrow (\exists Z)(\forall x)A(x,Z_x),$$

for Σ^1_1 formulas $A(x,Y)$, and induction on the natural numbers restricted to sets, is well-known to be a conservative extension of Peano arithmetic PA, cf. e.g. [14]. Clearly, (Σ^1_1-AC) is closely related to *Join* in explicit mathematics and, indeed, we have that ETJ and Σ^1_1-AC_0 are proof-theoretically equivalent.

The corresponding system Σ^1_1-AC with induction on the natural numbers admitted for all formulas in the language $\mathcal{L}_2$ of second order arithmetic, is proof-theoretically equivalent to the system Π^1_0-$\mathsf{CA}_{<\varepsilon_0}$ for iterated arithmetic comprehension below ε_0 (cf. [14]) and, hence, according to our discussion in Section 2.2, we have that $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})$ and Σ^1_1-AC are proof-theoretically equivalent with limiting ordinal $\varphi\varepsilon_0 0$.

Moreover, there are very natural subsystems of Kripke-Platek set theory which are closely related to ETJ and $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})$, but also to our ordinal theories OAD and $\mathsf{OAD} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$. The set theory KPu^0 of Jäger [58] formalizes an admissible universe of sets above the natural numbers as urelements; in KPu^0 induction on the natural numbers is restricted to set-theoretic Δ_0 formulas and $\in$ induction is omitted completely. It is shown loc. cit. that KPu^0 is a conservative extension of Peano arithmetic PA. If the schema of induction is added to KPu^0 for all set-theoretic formulas or $\mathcal{L}_s$ formulas for short, then one obtains a system of strength exactly Σ^1_1-AC, cf. Jäger [54, 58]. Thus, ETJ and $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})$ are proof-theoretically equivalent to KPu^0 and $\mathsf{KPu}^0 + (\mathcal{L}_s\text{-}\mathsf{I_N})$, respectively.

Finally, there are well-known systems of Martin Löf's type theory which can be measured against ETJ and $\mathsf{ETJ} + (\mathbb{L}\text{-}\mathsf{I_N})$. The standard version ML of constructive type theory is known to be of the same strength as Heyting arithmetic HA and, hence, ETJ, cf. e.g. [8, 128] for a detailed discussion. Moreover, if ML is extended by one universe which reflects the type generating principles of ML, then the resulting system is proof-theoretically equivalent to Σ^1_1-AC; this was first shown by Aczel [1], see also Feferman [30].

5.2 Systems of strength EIN

We proceed with our discussion and now turn to systems which are related to EIN and $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$. Very close in spirit to EIN is Jäger's admissible set theory KPi^0, which extends KPu^0 by the limit axiom claiming that each set is contained in an admissible set,

$$(\forall x)(\exists y)(x \in y \wedge \mathsf{Ad}(y)).$$

Thus KPi^0 formalizes a recursively inaccessible universe of sets. Recall, however, that we have no foundation in KPi^0 and induction on the natural numbers for Δ_0 formulas only. We assume, in addition, that admissibles are linearly ordered in KPi^0. It is shown in Jäger [57] that $|\mathsf{KPi}^0| = \Gamma_0$.

If we augment KPi^0 with the full schema of complete induction on the natural numbers, $(\mathcal{L}_s\text{-}\mathsf{I_N})$, then we obtain that $|\mathsf{KPi}^0 + (\mathcal{L}_s\text{-}\mathsf{I_N})| = \varphi 1 \varepsilon_0 0$. The lower bound computation runs similar to the wellordering proof for $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$, however, some additional considerations are needed, cf. the crucial lemma due to Jäger reported in [122, 118]. For determining the upper proof-theoretic bound of $\mathsf{KPi}^0 + (\mathcal{L}_s\text{-}\mathsf{I_N})$ one essentially follows the pattern of the treatment of our ordinal theory $\mathsf{OIN} + (\mathcal{L}_{\mathbb{O}}\text{-}\mathsf{I_N})$.

Turning to subsystems of second order arithmetic, we have already mentioned that EIN interprets Friedman's ATR_0 (cf. Marzetta [82, 83]), and in fact such an embedding was first given by Jäger in KPi^0. We recall that the axiom schema (ATR) of arithmetical transfinite recursion claims the existence of the arithmetic jump hierarchy along each given wellordering. Recently, Avigad [4] gave a neat equivalent formulation of (ATR). His principle (FP) asserts for each X positive arithmetic operator $\mathcal{A}(X, Y, x, y)$[1] (possibly depending on a set parameter Y and a number parameter y) the existence of an $\mathcal{A}$ fixed point,

$$(\mathsf{FP}) \qquad (\exists X)(\forall x)[x \in X \leftrightarrow \mathcal{A}(X, Y, x, y)].$$

It is shown in [4] that (ATR) and (FP) are equivalent over ACA_0.

Further systems which are closely related to the axioms (FP) and (ATR) are *iterated fixed point theories for positive arithmetic operators.* These were first studied by Feferman [30] in finitely iterated form and, more recently, in Jäger, Kahle, Setzer, and Strahm [62] for transfinite iterations. The principal axioms of the theory $\widehat{\mathsf{ID}}_\alpha$ claim for each X positive arithmetic operator $\mathcal{A}(X, Y, x, y)$ a hierarchy $(\mathsf{H}^{\mathcal{A}}_a)_{a \prec \alpha}$ of $\mathcal{A}$ fixed points,

$$(\forall a \prec \alpha)(\forall x)[\mathsf{H}^{\mathcal{A}}_a(x) \leftrightarrow \mathcal{A}(\mathsf{H}^{\mathcal{A}}_a, \mathsf{H}^{\mathcal{A}}_{\prec a}, x, a)].$$

Here $\prec$ denotes a standard primitive recursive wellordering of order type at least α and $\mathsf{H}^{\mathcal{A}}_{\prec a}$ is the disjoint union of $\mathsf{H}^{\mathcal{A}}_b$ $(b \prec a)$. Moreover, $\widehat{\mathsf{ID}}_{<\alpha}$ denotes

[1] These operators are not to be confused with the operators for non-monotone inductive definitions used in our ordinal theories above.

the union of the theories $\widehat{\mathsf{ID}}_\beta$ for β less than α. It has been shown in [30] that $|\widehat{\mathsf{ID}}_{<\omega}| = \Gamma_0$. Moreover, the proof-theoretic analysis of $\widehat{\mathsf{ID}}_\alpha$ for $\alpha \geq \omega$ in [62] reveals, in particular, the following important special cases:

$$|\widehat{\mathsf{ID}}_\omega| = \Gamma_{\varepsilon_0}, \qquad |\widehat{\mathsf{ID}}_{<\omega^\omega}| = \varphi 1\omega 00, \qquad |\widehat{\mathsf{ID}}_{<\varepsilon_0}| = \varphi 1 \varepsilon_0 0.$$

Summarizing, we have that EIN and $\mathsf{EIN}+(\mathbb{L}\text{-}\mathsf{I_N})$ are proof-theoretically equivalent to $\widehat{\mathsf{ID}}_{<\omega}$ and $\widehat{\mathsf{ID}}_{<\varepsilon_0}$, respectively. For a direct reduction of $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ to $\widehat{\mathsf{ID}}_{<\varepsilon_0}$ see Strahm [121]. Moreover, autonomous fixed point progressions and a notion of transfinite fixed point recursion are studied in Strahm [118], cf. also Rüede [106]. Finally, various research efforts were made in connection with fixed point theories based on *intuitionistic logic*, cf. the work of Arai [2], Buchholz [13], as well as Rüede and Strahm [107].

Let briefly return to subsystems of second order arithmetic, in particular extensions of ATR_0 which can be measured against transfinitely iterated fixed point theories. If we denote by ATR the system ATR_0 plus the full schema of induction on the natural numbers, then it is known by Friedman (cf. Simpson [114]) and Jäger [55] that $|\mathsf{ATR}| = \Gamma_{\varepsilon_0}$; thus, ATR is equivalent to $\widehat{\mathsf{ID}}_\omega$ by the above mentioned result from [62]. On the other hand, it is known since long that the schema of Σ^1_1 dependent choice, (Σ^1_1-DC), consisting of the assertions

$$(\Sigma^1_1\text{-}\mathsf{DC}) \qquad (\forall X)(\exists Y)A(X,Y) \rightarrow (\forall X)(\exists Z)[Z_0 = X \wedge (\forall u)A(Z_u, Z_{u+1})]$$

for each Σ^1_1 formula A of $\mathcal{L}_2$, is not derivable in ATR, cf. Simpson [115]. Hence, it is natural to ask about the strength of $\mathsf{ATR}_0 + (\Sigma^1_1\text{-}\mathsf{DC})$ and $\mathsf{ATR} + (\Sigma^1_1\text{-}\mathsf{DC})$: it is shown in Jäger and Strahm [71] that these two systems prove the same arithmetic sentences as $\widehat{\mathsf{ID}}_{<\omega^\omega}$ and $\widehat{\mathsf{ID}}_{<\varepsilon_0}$, respectively. In particular, we have that our system of explicit mathematics $\mathsf{EIN} + (\mathbb{L}\text{-}\mathsf{I_N})$ is also equivalent to $\mathsf{ATR} + (\Sigma^1_1\text{-}\mathsf{DC})$.

Coming back to Martin-Löf type theory, it is known from Feferman [30] that constructive type theory with finitely many universes $\mathsf{ML}_{<\omega}$ is proof-theoretically equivalent to $\widehat{\mathsf{ID}}_{<\omega}$ and, hence, to the system EIN. More recently, Rathjen [97, 99] has investigated the so-called superuniverse in type theory and shown that its strength is exactly $\varphi 1 \Gamma_0 0$, the proof-theoretic ordinal of the system $\widehat{\mathsf{ID}}_{<\Gamma_0}$. Finally, we want to mention the recent work of Kahle [73] in the context of Aczel's Frege structures (cf. also Cantini [17]) augmented by a suitable notion of universe, which provides systems that can also be compared with transfinitely iterated fixed point theories.

5.3 Systems of strength EMA

To conclude this chapter, let us discuss some systems of admissible set theory and second order arithmetic which are proof-theoretically equivalent to EMA and $\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I_N})$.

The prime candidate in the framework of admissible set theory is the theory KPm^0 of Jäger and Strahm [66], which is the metapredicative version of Rathjen's KPM, cf. [94, 95]. KPm^0 extends KPu^0 by the schema of Π_2 reflection on admissible sets, which asserts for all set-theoretic Δ_0 formulas $A(a, b, \vec{c})$ whose parameters belong to the list $a, b, \vec{c}$ the statement

$$(\forall x)(\exists y)A(x, y, \vec{c}) \rightarrow (\exists z)[\mathsf{Ad}(z) \wedge \vec{c} \in z \wedge (\forall x \in z)(\exists y \in z)A(x, y, \vec{c})].$$

We have that $|\mathsf{KPm}^0| = \varphi\omega00$ and $|\mathsf{KPm}^0 + (\mathcal{L}_s\text{-}\mathsf{I_N})| = \varphi\varepsilon_0 00$. The lower bound and upper bound computations are very similar in spirit to those presented in this thesis for EMA and OMA, respectively, cf. also Jäger and Strahm [66] and Strahm [118].

An interesting principle in the context of second order arithmetic formally similar to set-theoretic Π_2 reflection on admissibles is the schema of Π^1_2 reflection on countably coded ω models of Σ^1_1-DC, cf. Rüede [106, 103, 104]. This axiom schema asserts for each arithmetic $\mathcal{L}_2$ formula $A(X, Y, \vec{Z})$ with all its set parameters indicated,

$$(\forall X)(\exists Y)A(X, Y, \vec{Z}) \rightarrow \\ (\exists M)[M \models_\omega \Sigma^1_1\text{-}\mathsf{DC} \wedge \vec{Z} \,\dot{\in}\, M \wedge (\forall X \,\dot{\in}\, M)(\exists Y \,\dot{\in}\, M)A(X, Y, \vec{Z})].$$

Here $Z \,\dot{\in}\, M$ abbreviates $(\exists u)(Z = M_u)$ with M_u denoting as usual the uth "slice" of the countably coded model M. Rüede [106, 103, 104] analyzes the above principle on the basis of ACA_0 and ACA_0 plus full induction on the natural numbers and establishes the limiting ordinals $\varphi\omega00$ and $\varphi\varepsilon_0 00$.

A further natural axiom analyzed by Rüede is the so-called schema of Σ^1_1 transfinite dependent choice, (Σ^1_1-TDC), which is the expected transfinite generalization of (Σ^1_1-DC). It includes for each Σ^1_1 formula $A(X, Y)$ of $\mathcal{L}_2$,

$$(\Sigma^1_1\text{-}\mathsf{TDC}) \quad (\forall X)(\exists Y)A(X, Y) \wedge \mathsf{WO}(U) \rightarrow (\exists Z)(\forall a \in \mathsf{field}(U))A(Z_{Ua}, Z_a).$$

In this formula, $\mathsf{WO}(U)$ expresses that U codes a wellordering and $\mathsf{field}(U)$ signifies the field of U. Moreover, Z_{Ua} is the disjoint union of all projections Z_b with $\langle b, a\rangle$ in the wellordering U.

Using the technique of so-called pseudo hierarchies (cf. [115]), Rüede [106, 103] was able to show that the schema of Π^1_2 reflection on ω models of Σ^1_1-DC and the schema of transfinite dependent choice (Σ^1_1-TDC) are *equivalent* on the basis of ACA_0. Thus, in view of our discussion above, we have that the systems $\mathsf{ACA}_0 + (\Sigma^1_1\text{-}\mathsf{TDC})$ and $\mathsf{ACA}_0 + (\Sigma^1_1\text{-}\mathsf{TDC}) + (\mathcal{L}_2\text{-}\mathsf{I_N})$ are of the same proof-theoretic strength as EMA and $\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I_N})$, respectively.

This concludes our discussion on systems related to EMA and $\mathsf{EMA} + (\mathbb{L}\text{-}\mathsf{I_N})$.

Conclusion of Part I

In this first part of our habilitation thesis we have studied various formalisms of explicit mathematics based on elementary comprehension and join and augmented by certain principles for generating *universes*. In particular, we have examined the *limit axiom* (L) and the *Mahlo axiom* (M) leading to the two theories EIN and EMA of explicit inaccessibility and Mahloness, respectively. We have given a detailed proof-theoretic analysis of these axiomatic frameworks and classified their strength using a ternary Veblen or φ function. Characteristic for the underlying proof-theoretic analyses is their *metapredicativity*: the techniques and tools used in this part of our thesis entirely belong to the world of predicative proof theory, although the systems under consideration go well beyond the Feferman Schütte ordinal Γ_0 with respect to their proof-theoretic strength.

The results obtained in the first part of this thesis significantly extend the realm of metapredicativity. Next immediate steps concern the analysis of reflection principles going beyond Mahloness, i.e. Π_2 reflection on admissibles. The obvious candidates are Π_3 reflection or even reflection for arbitrary set-theoretic statements. It turns out that the former corresponds to φ functions of arbitrary finite arity whereas the latter takes us up to the Bachmann Howard ordinal. An interesting question in this connection is whether there is a sensible formal notion of the *limit of metapredicativity* and, if the answer is positive, to determine this limit.

Let us return to explicit mathematics and briefly address the question of how to formalize higher reflection principles in the language of types and names. In their seminal paper [100], Richter and Aczel introduced the notion of a 2-admissible ordinal, which can be generalized in order to define the concept of an n-admissible ordinal. In particular, they showed loc. cit. that the 2-admissible ordinals are exactly the Π_3 reflecting ordinals. Moreover, the cardinal analogue of 2-admissible ordinals are the 2-regular cardinals, which turn out to be exactly the Π^1_1 indescribable cardinals. Indeed, the definition of a 2-admissible ordinal gives immediate rise to an explicit analogue of Π_3 reflection, and the corresponding metapredicative system of explicit mathematics has the strength mentioned above. It seems that even the notion of an n-admissible ordinal for $n > 2$ can be used to obtain a suitable analogue in the language of explicit mathematics of Π_n reflection for each $n > 3$.

Summarizing, the considerations in the first part of this habilitation thesis give rise to further interesting research work in connection with metapredicativity as well as explicit notions of higher reflection. The latter are of interest not only in their metapredicative but also in impredicative form.

Part II

Applicative theories and complexity

Plan of Part II

Let us give a quick guided tour through this part of our habilitation thesis.

We start in Chapter 6 with a short review of known recursion-theoretic characterizations of various function complexity classes on the binary words $\mathbb{W}$ by means of bounded recursion on notation as well as bounded unary recursion. The so-obtained machine-independent characterizations will be crucial for lower as well as upper bound arguments used in the sequel of the thesis.

In Chapter 7 we set up the central applicative framework. We start with introducing the basic theory B of operations and words and recall some of its crucial properties. Then we present various forms of bounded induction and define the four central systems PT, PS, PTLS, and LS, corresponding to the functions computable in polynomial time, polynomial space, simultaneously polynomial time and linear space, as well as linear space, respectively.

In Chapter 8 we provide lower bound arguments for our applicative systems, i.e., we show that the functions from the respective function complexity classes are provably total in the four applicative theories mentioned above. In particular, we will see that forms of bounded recursion are very naturally derived by means of the fixed point theorem and exploiting our various principles of bounded induction.

Higher type issues are at the heart of Chapter 9. There we will recapitulate an intensional and an extensional version of the Cook-Urquhart system PV^ω and show that both systems are naturally contained in our applicative system PT for the polynomial time computable functions. Indeed, the embeddings presented in this chapter also give rise immediately to higher type systems corresponding to PS, PTLS, and LS.

Upper bounds for the four systems PT, PS, PTLS, and LS are established in Chapter 10. The upper bound arguments proceed in two steps. First, standard *partial cut elimination* is employed in a sequent-style version of our systems in order to show that derivations of sequents of positive formulas can be restricted to positive cuts. The second crucial step consists in establishing very uniform *realizability* theorems for our four theories, where a notion of realizability for positive formulas in the standard open term model $\mathcal{M}(\lambda\eta)$ is used.

In Chapter 11 we present further natural applicative systems for various classes of computable functions. In particular, we will study a system PH which is closely related to the *polynomial time hierarchy*; the crucial axiom of PH is a very uniform type two functional π for bounded quantification. Further investigations in this chapter concern applicative theories whose provably total functions are exactly the primitive recursive functions.

Chapter 6

Some recursion-theoretic characterizations of complexity classes

In this chapter we review know recursion-theoretic characterizations of various classes of computational complexity. We will work over the set of binary words $\mathbb{W} = \{0,1\}^*$. Our main interest in the sequel are the functions on $\mathbb{W}$ which are computable on a Turing machine in *polynomial time*, *simultaneously polynomial time and linear space*, *polynomial space*, and *linear space*. For an extensive discussion of recursion-theoretic or function algebra characterizations of complexity classes the reader is referred to the survey article Clote [22].

Historically, the first function algebra characterizations of time and space complexity classes are due to Ritchie and Cobham in 1963 and 1964, respectively. Cobham [23] gave a characterization of the polynomial time computable functions by means of bounded recursion on notation. Ritchie [101] showed that Grzegorczyk's class $\mathcal{E}_2$ coincides with the linear space computable functions. Thus, both the polynomial time and linear space computable functions can be considered in a suitable manner as a natural restriction of the class of primitive recursive functions.

6.1 Time and space complexity classes

In the sequel we denote by $\mathbb{W}$ the set of *finite* binary words $\{\epsilon, 0, 1, 00, 01, \ldots\}$, more compactly, $\mathbb{W} = \{0, 1\}^*$. Here as usual ϵ signifies the empty word. In order to study computability of word functions F from $\mathbb{W}^n$ to $\mathbb{W}$ we make use of the usual notion of a *multitape Turing machine*, cf. e.g. [5, 12, 51, 86, 87, 89] for this and related concepts. Our main concern is the characterization of complexity classes according to the use of resources, particularly *time* and *space*. More precisely, we are interested in the following four classes FPtime, FPtimeLinspace, FPspace, and FLinspace of functions on $\mathbb{W}$, cf. also the above cited references for more information on these standard classes.

Definition 41 *We introduce the following four classes of word functions.*

1. *Define* FPtime *to denote the class of functions on* $\mathbb{W}$ *which are computable on a multitape Turing machine in* time *bounded by a* polynomial *in the length of the input.*

2. *Define* FPtimeLinspace *to denote the class of functions on* $\mathbb{W}$ *which are computable on a multitape Turing machine in* time *bounded by a* polynomial *in the length of the input and* simultaneously *in* space *bounded by a* linear *function in the length of the input.*

3. *Define* FPspace *to denote the class of functions on* $\mathbb{W}$ *which are computable on a multitape Turing machine in* space *bounded by a* polynomial *in the length of the input.*

4. *Define* FLinspace *to denote the class of functions on* $\mathbb{W}$ *which are computable on a multitape Turing machine in* space *bounded by a* linear *function in the length of the input.*

We are interested in various kinds of successor operations on the binary words $\mathbb{W}$. As usual, s_0 and s_1 denote the binary successor functions which concatenate 0 and 1 to the end of a given binary word, respectively. We are also given a unary lexicographic successor s_ℓ on $\mathbb{W}$, which satisfies for all x in $\mathbb{W}$ the following recursion equations,

$$\mathsf{s}_\ell(\epsilon) = 0, \qquad \mathsf{s}_\ell(\mathsf{s}_0 x) = \mathsf{s}_1 x, \qquad \mathsf{s}_\ell(\mathsf{s}_1 x) = \mathsf{s}_0(\mathsf{s}_\ell x).$$

Observe that s_ℓ is the successor operation in the natural wellordering $<_\ell$ of $\mathbb{W}$ according to which words are ordered by length and words of the same length are ordered lexicographically. Thinking of binary words as binary representations of natural numbers, s_ℓ essentially corresponds to the usual successor operation on the natural numbers. Clearly, we have that s_0, s_1, and s_ℓ belong to FPtimeLinspace.

Finally, we let $*$ and $\times$ stand for the binary operations of *word concatenation* and *word multiplication*, respectively, where $x{\times}y$ denotes the word x, length of y times concatenated with itself. We have that $*$ and $\times$ satisfy for all x, y in $\mathbb{W}$ the following recursion equations,

$$\begin{aligned} x{*}\epsilon &= x, & x{\times}\epsilon &= \epsilon, \\ x{*}(\mathsf{s}_0 y) &= \mathsf{s}_0(x{*}y), & x{\times}(\mathsf{s}_0 y) &= (x{\times}y){*}x, \\ x{*}(\mathsf{s}_1 y) &= \mathsf{s}_1(x{*}y), & x{\times}(\mathsf{s}_1 y) &= (x{\times}y){*}x. \end{aligned}$$

Word concatenation $*$ belongs to FPtimeLinspace, whereas word multiplication $\times$ belongs to FPtime. Obviously, $\times$ is not a member of the class FPtimeLinspace.

6.2 Four function algebras

Towards a function algebra characterization of the complexity classes mentioned above, we now want to introduce two schemas of *bounded recursion*. For that purpose, let G, H_0, H_1 and K be given functions on binary words of appropriate arities. We say the function F is defined by *bounded recursion on notation* (BRN) from G, H_0, H_1 and K, if

$$\begin{array}{rcll} F(\vec{x}, \epsilon) & = & G(\vec{x}), & \\ F(\vec{x}, \mathsf{s}_i y) & = & H_i(\vec{x}, y, F(x, \vec{y})), & \qquad (i = 0, 1) \\ F(\vec{x}, y) & \leq & K(\vec{x}, y) & \end{array}$$

for all $\vec{x}, y$ in $\mathbb{W}$. Here $x \leq y$ signifies that the length of the word x is less than or equal to the length of the word y. On the other hand, a function F is defined by *bounded lexicographic recursion* (BRL) from G, H and K, if

$$\begin{array}{rcl} F(\vec{x}, \epsilon) & = & G(\vec{x}), \\ F(\vec{x}, \mathsf{s}_\ell y) & = & H(\vec{x}, y, F(x, \vec{y})), \\ F(\vec{x}, y) & \leq & K(\vec{x}, y) \end{array}$$

for all $\vec{x}, y$ in $\mathbb{W}$. Hence, the crucial difference between bounded recursion on notation (BRN) and bounded lexicographic or unary recursion (BRL) is that the former recursion scheme acts along the *branches of the binary tree*, whereas the latter form of bounded recursion is with respect to the lexicographic ordering of the *full binary tree*.

In the following we use the notation of Clote [22] for a compact representation of function algebras. Accordingly, we call (partial) mappings from functions on $\mathbb{W}$ to functions on $\mathbb{W}$ *operators*. If $\mathcal{X}$ is a set of functions on $\mathbb{W}$ and OP is a collection of operators, then $[\mathcal{X}; \mathsf{OP}]$ is used to denote the smallest set of functions containing $\mathcal{X}$ and closed under the operators in OP. We call $[\mathcal{X}; \mathsf{OP}]$ a *function algebra*.

Our crucial examples of operators in the sequel are (BRN) and (BRL). A further operator is the *composition operator* (COMP), which takes functions $F, G_1, \ldots, G_n$ and maps them to the usual composition H of F with $G_1, \ldots, G_n$, i.e., we have for all $\vec{x}$ in $\mathbb{W}$,

$$H(\vec{x}) = F(G_1(\vec{x}), \ldots, G_n(\vec{x})).$$

Below we also use I for the usual collection of projection functions and we simply write ϵ for the 0-ary function being constant to the empty word ϵ.

We are now ready to state the function algebra characterizations of the four complexity classes which are relevant in this thesis. The characterization of FPtime is due to Cobham [23]. The delineations of FPtimeLinspace and FPspace are due to Thompson [126]. Finally, the fourth assertion of our theorem is due to Ritchie [101]. For a uniform presentation of all these results we urge the reader to consult Clote [22].

Theorem 42 *We have the following function algebra characterizations of the complexity classes defined above:*

1. $[\epsilon, \mathsf{I}, \mathsf{s}_0, \mathsf{s}_1, *, \times; \mathsf{COMP}, \mathsf{BRN}] = \text{FPtime}$.
2. $[\epsilon, \mathsf{I}, \mathsf{s}_0, \mathsf{s}_1, *; \mathsf{COMP}, \mathsf{BRN}] = \text{FPtimeLinspace}$.
3. $[\epsilon, \mathsf{I}, \mathsf{s}_\ell, *, \times; \mathsf{COMP}, \mathsf{BRL}] = \text{FPspace}$.
4. $[\epsilon, \mathsf{I}, \mathsf{s}_\ell, *; \mathsf{COMP}, \mathsf{BRL}] = \text{FLinspace}$.

Let us mention that indeed word concatenation $*$ is redundant in the presence of word multiplication $\times$, and we have included it in the formulation of this theorem for reasons of uniformity only.

The inclusion "$\subseteq$" in the proof of the equations in the above theorem is rather straightforward. A crucial task in establishing the reverse direction is to show that the function $NEXT_M(x, c) = d$ belongs to the corresponding function algebra. Here c, d encode configurations of a suitable Turing machine M on input x and d is the configuration obtained in one step from configuration c.

Chapter 7

The applicative framework

In this chapter we will introduce the applicative systems which will be relevant in the rest of this thesis. We start with a precise description of the basic theory of operations and words B. The crucial axioms of B are those of an untyped *partial combinatory algebra*. We briefly review the central consequences of these axioms, namely *abstraction* and *recursion*. Further we will address B's standard recursion-theoretic model and mention further important classes of models.

Later we will discuss two basic forms of bounded induction on the binary words W, which will be used to set up the central applicative frameworks PT, PTLS, PS, and LS. An important notion is the one of a so-called $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formula, which can be seen as an abstract applicative analogue of Σ^b_1 or NP formulas in the context of first order bounded arithmetic.

7.1 The theory B of operations and words

All applicative systems to be considered below are formulated in the language $\mathcal{L}_{\mathsf{W}}$; it is a language of partial terms with *individual variables* $a, b, c, x, y, z, u, v, f, g, h, \ldots$ (possibly with subscripts). $\mathcal{L}_{\mathsf{W}}$ includes *individual constants* k, s (combinators), $\mathsf{p}, \mathsf{p}_0, \mathsf{p}_1$ (pairing and unpairing), d_{W} (definition by cases on binary words), ϵ (empty word) $\mathsf{s}_0, \mathsf{s}_1$ (binary successors), p_{W} (binary predecessor), $\mathsf{s}_\ell, \mathsf{p}_\ell$ (lexicographic successor and predecessor), $\mathsf{c}_\subseteq$ (initial subword relation) and l_{W} (tally length of binary words). We also assume that the two constants $*$ (word concatenation) and $\times$ (word multiplication) belong to $\mathcal{L}_{\mathsf{W}}$, however, not all our applicative systems will have axioms about $*$ and

$\times$. Finally, $\mathcal{L}_{\mathsf{W}}$ has a binary function symbol $\cdot$ for (partial) term application, unary relation symbols $\downarrow$ (defined) and W (binary words) as well as a binary relation symbol $=$ (equality).

The *terms* $(r, s, t, \ldots)$ of $\mathcal{L}_{\mathsf{W}}$ (possibly with subscripts) are inductively generated from the variables and constants by means of application $\cdot$. We write ts instead of $\cdot(t, s)$ and follow the standard convention of association to the left when omitting brackets in applicative terms. As usual, (s, t) is a shorthand for $\mathsf{p}st$. Moreover, we use the abbreviations 0 and 1 for $\mathsf{s}_0\epsilon$ and $\mathsf{s}_1\epsilon$, respectively. Furthermore, we write $s \subseteq t$ instead of $\mathsf{c}_{\subseteq}st = 0$ and $s \leq t$ for $\mathsf{l}_{\mathsf{W}}s \subseteq \mathsf{l}_{\mathsf{W}}t$; $s \subset t$ and $s < t$ are understood accordingly. Finally, $s{*}t$ stands for $*st$, and $s{\times}t$ for $\times st$.

The *formulas* $(A, B, C, \ldots)$ of $\mathcal{L}_{\mathsf{W}}$ (possibly with subscripts) are built from the atomic formulas $(s = t)$, $s\downarrow$ and $\mathsf{W}(s)$ by closing under negation, disjunction, conjunction, implication, as well as existential and universal quantification over individuals.

Our conventions concerning substitutions are as follows. As usual we write $t[\vec{s}/\vec{x}]$ and $A[\vec{s}/\vec{x}]$ for the substitution of the terms $\vec{s}$ for the variables $\vec{x}$ in the term t and the formula A, respectively. In this connection we often write $A(\vec{x})$ instead of A and $A(\vec{s})$ instead of $A[\vec{s}/\vec{x}]$.

Our applicative theories are based on *partial* term application. Hence, it is not guaranteed that terms have a value, and $t\downarrow$ is read as t *is defined* or t *has a value*. The *partial equality relation* $\simeq$ is introduced by

$$s \simeq t \;:=\; (s\downarrow \vee\, t\downarrow) \rightarrow (s = t).$$

In the following we will use the following natural abbreviations concerning the predicate W ($\vec{s} = s_1, \ldots, s_n$):

$$\begin{array}{rcl}
\vec{s} \in \mathsf{W} & := & \mathsf{W}(s_1) \wedge \cdots \wedge \mathsf{W}(s_n), \\
(\exists x \in \mathsf{W})A & := & (\exists x)(x \in \mathsf{W} \wedge A), \\
(\forall x \in \mathsf{W})A & := & (\forall x)(x \in \mathsf{W} \rightarrow A), \\
(\exists x \leq t)A & := & (\exists x \in \mathsf{W})(x \leq t \wedge A), \\
(\forall x \leq t)A & := & (\forall x \in \mathsf{W})(x \leq t \rightarrow A), \\
(t : \mathsf{W} \rightarrow \mathsf{W}) & := & (\forall x \in \mathsf{W})(tx \in \mathsf{W}), \\
(t : \mathsf{W}^{m+1} \rightarrow \mathsf{W}) & := & (\forall x \in \mathsf{W})(tx : \mathsf{W}^m \rightarrow \mathsf{W}).
\end{array}$$

Before we turn to precise axiomatizations, let us give a short informal interpretation of the syntax of the language $\mathcal{L}_\mathsf{W}$. The individual variables are conceived of as ranging over a universe V of computationally amenable objects, which can freely be applied to each other. Self-application is meaningful, but not necessarily total. V is assumed to be combinatory complete, due to the presence of the well-known combinators k and s, and V is closed under pairing. There is a collection of objects $W \subseteq V$, consisting of finite sequences of 0's and 1's. W is closed under various kinds of successor and predecessor operations as well as definition by cases. In addition, there are operations for the initial subword relation as well as the tally length of a binary word. Possibly, operations for word concatenation and/or word multiplication are explicitly included.

We now introduce the *basic theory of operations and words* B. The underlying logic of B is the *classical* logic of partial terms due to Beeson [8, 9]; it corresponds to E^+ logic with strictness and equality of Troelstra and Van Dalen [127]. According to this logic, quantifiers range over defined objects only, so that the usual axioms for $\exists$ and $\forall$ are modified to

$$A(t) \wedge t{\downarrow} \rightarrow (\exists x)A(x) \quad \text{and} \quad (\forall x)A(x) \wedge t{\downarrow} \rightarrow A(t),$$

and one further assumes that $(\forall x)(x{\downarrow})$. The *strictness axioms* claim that if a compound term is defined, then so also are all its subterms, and if a positive atomic statement holds, then all terms involved in that statement are defined. Note that $t{\downarrow} \leftrightarrow (\exists x)(t = x)$, so definedness need not be taken as basic symbol. The reader is referred to [8, 9, 127] for a detailed exposition of the logic of partial terms.

We are now ready to spell out in detail the *non-logical axioms* of B. To improve readability we divide the axioms into the following six groups.

I. Partial combinatory algebra and pairing

(1) $\mathsf{k}xy = x$,

(2) $\mathsf{s}xy{\downarrow} \wedge \mathsf{s}xyz \simeq xz(yz)$,

(3) $\mathsf{p}_0(x, y) = x \wedge \mathsf{p}_1(x, y) = y$.

II. Definition by cases on W

(4) $a \in \mathsf{W} \wedge b \in \mathsf{W} \wedge a = b \rightarrow \mathsf{d}_\mathsf{W}xyab = x$,

(5) $a \in \mathsf{W} \wedge b \in \mathsf{W} \wedge a \neq b \;\rightarrow\; \mathsf{d_W} xyab = y.$

III. Closure, binary successors and predecessor

(6) $\epsilon \in \mathsf{W} \;\wedge\; (\forall x \in \mathsf{W})(\mathsf{s_0} x \in \mathsf{W} \wedge \mathsf{s_1} x \in \mathsf{W}),$

(7) $\mathsf{s_0} x \neq \mathsf{s_1} y \;\wedge\; \mathsf{s_0} x \neq \epsilon \;\wedge\; \mathsf{s_1} x \neq \epsilon,$

(8) $\mathsf{p_W} : \mathsf{W} \rightarrow \mathsf{W} \;\wedge\; \mathsf{p_W} \epsilon = \epsilon,$

(9) $x \in \mathsf{W} \;\rightarrow\; \mathsf{p_W}(\mathsf{s_0} x) = x \;\wedge\; \mathsf{p_W}(\mathsf{s_1} x) = x,$

(10) $x \in \mathsf{W} \wedge x \neq \epsilon \;\rightarrow\; \mathsf{s_0}(\mathsf{p_W} x) = x \vee \mathsf{s_1}(\mathsf{p_W} x) = x.$

IV. Lexicographic successor and predecessor

(11) $\mathsf{s}_\ell : \mathsf{W} \rightarrow \mathsf{W} \;\wedge\; \mathsf{s}_\ell \epsilon = 0,$

(12) $x \in \mathsf{W} \;\rightarrow\; \mathsf{s}_\ell(\mathsf{s_0} x) = \mathsf{s_1} x \;\wedge\; \mathsf{s}_\ell(\mathsf{s_1} x) = \mathsf{s_0}(\mathsf{s}_\ell x),$

(13) $\mathsf{p}_\ell : \mathsf{W} \rightarrow \mathsf{W} \;\wedge\; \mathsf{p}_\ell \epsilon = \epsilon,$

(14) $x \in \mathsf{W} \;\rightarrow\; \mathsf{p}_\ell(\mathsf{s}_\ell x) = x,$

(15) $x \in \mathsf{W} \wedge x \neq \epsilon \;\rightarrow\; \mathsf{s}_\ell(\mathsf{p}_\ell x) = x.$

V. Initial subword relation.

(16) $x \in \mathsf{W} \wedge y \in \mathsf{W} \;\rightarrow\; \mathsf{c}_\subseteq xy = 0 \vee \mathsf{c}_\subseteq xy = 1,$

(17) $x \in \mathsf{W} \;\rightarrow\; (x \subseteq \epsilon \leftrightarrow x = \epsilon),$

(18) $x \in \mathsf{W} \wedge y \in \mathsf{W} \wedge y \neq \epsilon \;\rightarrow\; (x \subseteq y \leftrightarrow x \subseteq \mathsf{p_W} y \vee x = y),$

(19) $x \in \mathsf{W} \wedge y \in \mathsf{W} \wedge z \in \mathsf{W} \wedge x \subseteq y \wedge y \subseteq z \;\rightarrow\; x \subseteq z.$

VI. Tally length of binary words

(20) $\mathsf{l_W} : \mathsf{W} \rightarrow \mathsf{W} \;\wedge\; \mathsf{l_W} \epsilon = \epsilon,$

(21) $x \in \mathsf{W} \;\rightarrow\; \mathsf{l_W}(\mathsf{s_0} x) = \mathsf{s_1}(\mathsf{l_W} x) \;\wedge\; \mathsf{l_W}(\mathsf{s_1} x) = \mathsf{s_1}(\mathsf{l_W} x),$

(22) $x \in \mathsf{W} \wedge \mathsf{l_W}(x) = x \;\rightarrow\; \mathsf{l_W}(\mathsf{s}_\ell x) = \mathsf{s_1} x,$

(23) $x \in \mathsf{W} \wedge \mathsf{l_W}(x) \neq x \;\rightarrow\; \mathsf{l_W}(\mathsf{s}_\ell x) = \mathsf{l_W}(x),$

(24) $x \in \mathsf{W} \wedge y \in \mathsf{W} \,\rightarrow\, x \leq y \vee y \leq x$.

Let us immediately turn to two crucial consequences of the partial combinatory algebra axioms (1) and (2) of B, namely *abstraction* and *recursion*. These two central results appear in slightly different form than in the setting of a total combinatory algebra, the essential ingredients in the proofs, however, are the same. The relevant arguments are given, for example, in Beeson [8] or Feferman [27].

Lemma 43 (Abstraction) *For each $\mathcal{L}_{\mathsf{W}}$ term t and all variables x there exists an $\mathcal{L}_{\mathsf{W}}$ term $(\lambda x.t)$ whose variables are those of t, excluding x, so that* B *proves*

$$(\lambda x.t)\downarrow \,\wedge\, (\lambda x.t)x \simeq t.$$

As usual, we generalize λ abstraction to several arguments by iterating abstraction for one argument, i.e., $(\lambda x_1 \dots x_n.t)$ abbreviates $(\lambda x_1.(\dots(\lambda x_n.t))$.

Lemma 44 (Recursion) *There exists a closed $\mathcal{L}_{\mathsf{W}}$ term* rec *so that* B *proves*

$$\mathsf{rec} f\downarrow \,\wedge\, \mathsf{rec} f x \simeq f(\mathsf{rec} f)x.$$

Clearly, recursion nicely demonstrates the power of self-application. It will be an essential tool for defining operations in the various applicative systems to be introduced below.

In the meanwhile let us briefly sketch B's standard recursion-theoretic model *PRO* of *partial recursive operations*. The universe of *PRO* consists of the set of finite 0-1 sequences $\mathbb{W} = \{0,1\}^*$, and W is interpreted by $\mathbb{W}$. Application $\cdot$ is interpreted as partial recursive function application, i.e. $x \cdot y$ means $\{x\}(y)$ in *PRO*, where $\{x\}$ is a standard enumeration of the partial recursive functions over $\mathbb{W}$. It is easy to find interpretations of the constants of $\mathcal{L}_{\mathsf{W}}$ so that all the axioms of B are true in *PRO*.

There are many more interesting models of the combinatory axioms, which can easily be extended to models of B. These include further recursion-theoretic models, term models, continuous models, generated models, and set-theoretic models. For detailed descriptions and results the reader is referred to Beeson [8], Feferman [29], and Troelstra and van Dalen [128]. We will make use of the so-called *extensional term model* of B in our upper bound arguments in Chapter 10; there we will define this model in some detail.

We finish this section by spelling out the obvious axioms for word concatenation and word multiplication in our applicative framework. Note, however, that these axioms do not belong to the theory B.

VII. Word concatenation.

(25) $* : \mathsf{W}^2 \to \mathsf{W}$,

(26) $x \in \mathsf{W} \;\to\; x{*}\epsilon = x$,

(27) $x \in \mathsf{W} \wedge y \in \mathsf{W} \;\to\; x{*}(\mathsf{s}_0 y) = \mathsf{s}_0(x{*}y) \;\wedge\; x{*}(\mathsf{s}_1 y) = \mathsf{s}_1(x{*}y)$.

VIII. Word multiplication.

(28) $\times : \mathsf{W}^2 \to \mathsf{W}$,

(29) $x \in \mathsf{W} \;\to\; x{\times}\epsilon = \epsilon$,

(30) $x \in \mathsf{W} \wedge y \in \mathsf{W} \;\to\; x{\times}(\mathsf{s}_0 y) = (x{\times}y){*}x \;\wedge\; x{\times}(\mathsf{s}_1 y) = (x{\times}y){*}x$.

In the following we write $\mathsf{B}(*)$ for the extension of B by the axioms (25)–(27), and $\mathsf{B}(*, \times)$ for B plus the axioms (25)–(30).

7.2 Bounded forms of induction

We have not yet specified induction principles on the binary words W; these are of course crucial for our proof-theoretic characterizations of complexity classes below. We start by defining three central classes of $\mathcal{L}_\mathsf{W}$ formulas.

We call an $\mathcal{L}_\mathsf{W}$ formula *positive* if it is built from the atomic formulas by means of disjunction, conjunction as well as existential and universal quantification over individuals; i.e., the positive formulas are exactly the implication and negation free $\mathcal{L}_\mathsf{W}$ formulas. We let Pos stand for the collection of positive formulas. Further, an $\mathcal{L}_\mathsf{W}$ formula is called W *free*, if the relation symbol W does not occur in it.

Most important in the sequel are the so-called *bounded (with respect to* W*) existential formulas* or $\Sigma^\mathsf{b}_\mathsf{W}$ *formulas* of $\mathcal{L}_\mathsf{W}$. A formula $A(f, x)$ belongs to the class $\Sigma^\mathsf{b}_\mathsf{W}$ if it has the form $(\exists y \leq fx)B(f, x, y)$ for $B(f, x, y)$ a *positive and* W *free* formula. It is important to recall here that bounded quantifiers range over W, i.e., $(\exists y \leq fx)B(f, x, y)$ stands for

$$(\exists y \in \mathsf{W})[y \leq fx \;\wedge\; B(f, x, y)].$$

Further observe that the matrix B of a $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formula can have unrestricted existential and universal individual quantifiers, not ranging over W, however.

Assuming that the bounding operation f in a $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formula has polynomial growth, $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formulas can be seen as a very abstract applicative analogue of Buss' Σ^b_1 formulas (cf. [15]) or Ferreira's NP formulas (cf. [37, 38]). Notice, however, whereas the latter classes of formulas define exactly the NP predicates, $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formulas of $\mathcal{L}_{\mathsf{W}}$ in general define highly undecidable sets in the standard recursion theoretic model *PRO*.

At the heart of our delineation of complexity classes below are forms of bounded (with respect to W) induction. These principles allow induction with respect to formulas in the class $\Sigma^{\mathsf{b}}_{\mathsf{W}}$, under the proviso that the bounding operation f has the right type. We will distinguish usual notation induction on binary words and the corresponding "slow" induction principle with respect to the lexicographic successor s_ℓ.

The scheme ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\mathsf{W}}$) of $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ *notation induction on* W includes for each formula $A(x) \equiv (\exists y \leq fx)B(f,x,y)$ in the formula class $\Sigma^{\mathsf{b}}_{\mathsf{W}}$,

$$(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}}) \qquad \begin{array}{l} f : \mathsf{W} \to \mathsf{W} \wedge A(\epsilon) \wedge (\forall x \in \mathsf{W})(A(x) \to A(\mathsf{s}_0 x) \wedge A(\mathsf{s}_1 x)) \\ \qquad\qquad\qquad \to (\forall x \in \mathsf{W})A(x) \end{array}$$

Accordingly, the induction scheme ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_\ell$) of $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ *lexicographic induction on* W claims for each formula $A(x) \equiv (\exists y \leq fx)B(f,x,y)$ in the class $\Sigma^{\mathsf{b}}_{\mathsf{W}}$,

$$(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell) \qquad \begin{array}{l} f : \mathsf{W} \to \mathsf{W} \wedge A(\epsilon) \wedge (\forall x \in \mathsf{W})(A(x) \to A(\mathsf{s}_\ell x)) \\ \qquad\qquad\qquad \to (\forall x \in \mathsf{W})A(x) \end{array}$$

We will prove in the next chapter (cf. Lemma 48) that indeed ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_\ell$) entails ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\mathsf{W}}$) over our base theory B. Further, let us mention that the principles of set induction and NP induction considered in Strahm [120] (cf. also Cantini [20]) are directly entailed by ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\mathsf{W}}$). Moreover, also the axiom of operation induction of Jäger and Strahm [69] is covered by the above bounded induction schemes. An induction principle related to ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\mathsf{W}}$) has previously been studied by Cantini [16] in the context of polynomially bounded operations (cf. also Cantini [20]).

Depending on whether we include ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\mathsf{W}}$) or ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_\ell$), and whether we assume as given only word concatenation or both word concatenation and word multiplication, we can now distinguish the following four applicative theories PT, PTLS, PS, and LS:

$$\mathsf{PT} := \mathsf{B}(*, \times) + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}}) \qquad \mathsf{PTLS} := \mathsf{B}(*) + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$$
$$\mathsf{PS} := \mathsf{B}(*, \times) + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell}) \qquad \mathsf{LS} := \mathsf{B}(*) + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell})$$

As the naming of these system suggests, it is our aim in the sequel to establish that the provably total operations on words of PT, PTLS, PS, and LS coincide with FPtime, FPtimeLinspace, FPspace, and FLinspace, respectively. On our way we will also be interested in some higher type aspects of our applicative systems.

Chapter 8

Deriving bounded recursions

It is the main purpose of this chapter to show that the provably total word functions of the systems PT, PTLS, PS, and LS include the classes FPtime, FPtimeLinspace, FPspace, and FLinspace, respectively. We set up our lower bound arguments in such a way as to facilitate the discussion on higher type issues in the subsequent chapter.

In proving lower bounds, we will make use of the function algebra characterizations of our complexity classes according to Theorem 42. A key step will be to use the recursion or fixed point lemma (Lemma 44) and combine it with our forms of bounded induction, $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ and $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell})$. As a byproduct, we will also show that $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ lexicographic induction $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell})$ entails $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ notation induction $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ over the base applicative theory B.

8.1 Provably total word functions

Let us first start with a formal definition of the notion of *provably total word function* of a given $\mathcal{L}_{\mathsf{W}}$ theory. First note that for each word $w \in \mathbb{W}$ we have a canonical closed term $\overline{w}$ of $\mathcal{L}_{\mathsf{W}}$ which represents w; $\overline{w}$ is inductively constructed form ϵ by means of the successor operations s_0 and s_1 as follows:

$$\overline{\epsilon} = \epsilon, \qquad \overline{\mathsf{s}_0 w} = \mathsf{s}_0 \overline{w}, \qquad \overline{\mathsf{s}_1 w} = \mathsf{s}_1 \overline{w}.$$

In the sequel we sometimes identify the $\mathcal{L}_{\mathsf{W}}$ term $\overline{w}$ with the binary word w when working in the language $\mathcal{L}_{\mathsf{W}}$.

Definition 45 *A function $F : \mathbb{W}^n \to \mathbb{W}$ is called provably total in an $\mathcal{L}_{\mathsf{W}}$ theory T, if there exists a closed $\mathcal{L}_{\mathsf{W}}$ term t_F such that* *(i)* $\mathsf{T} \vdash t_F : \mathsf{W}^n \to \mathsf{W}$ *and, in addition,*

(ii) $\mathsf{T} \vdash t_F \overline{w}_1 \cdots \overline{w}_n = \overline{F(w_1, \ldots, w_n)}$ *for all* $w_1, \ldots, w_n$ *in* $\mathbb{W}$.

The notion of a provably total word function is divided into two conditions (i) and (ii). The first condition (i) expresses that t_F is a total operation from W^n to W, *provably in the* $\mathcal{L}_\mathsf{W}$ *theory* T. Condition (ii), on the other hand, claims that t_F indeed represents the given function $F : \mathbb{W}^n \to \mathbb{W}$, for each fixed word w in $\mathbb{W}$.

Observe that one gets a too weak notion of provably total function if one drops condition (i). For example, in the theory B it is well-known that one can represent *all recursive functions* in the sense of (ii). The proof of this fact runs completely analogous to the argument in the untyped λ calculus showing that all recursive function are representable there (cf. [6, 48]). The crucial ingredient in the proof is of course the recursion or fixed point lemma (Lemma 44). Hence, for example, it is possible to find a closed $\mathcal{L}_\mathsf{W}$ term exp representing a suitable form of exponentiation on $\mathbb{W}$ in the sense of condition (ii) above, but indeed none of the theories introduced in the previous chapter is able to derive the totality or convergence statement $\mathsf{exp} : \mathsf{W} \to \mathsf{W}$.

8.2 Bounded induction yields bounded recursion

Our general strategy for proving lower bounds in the sequel is to make use of the function algebra characterizations of our complexity classes which we have discussed in Chapter 6. Crucial in the set up of the four function algebras of Theorem 42 are two forms of bounded recursion, namely *bounded recursion on notation* (BRN) and *bounded lexicographic recursion* (BRL). We will now show that (BRN) and (BRL) can be very smoothly and naturally represented in $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\mathsf{W})$ and $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$, respectively. The key in the proof below is the recursion or fixed point lemma (Lemma 44) and of course our carefully chosen forms of bounded induction.

In the sequel we also need the cut-off operator $|$ in order to describe bounded recursion in our systems. Informally speaking, $t \mid s$ is t if $t \leq s$ and s else. More formally, we can make use of definition by cases $\mathsf{d_W}$ and the characteristic function $\mathsf{c}_\subseteq$ in order to define $|$; then $t \mid s$ simply is an abbreviation for the $\mathcal{L}_\mathsf{W}$ term $\mathsf{d_W} ts(\mathsf{c}_\subseteq(\mathsf{l_W} t)(\mathsf{l_W} s))0$.

Let us now first turn to bounded recursion on notation (BRN) in the system $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$. In favor of a more compact and uniform presentation we state this form of recursion in our applicative setting by making use of one step function only and using the predecessor operation p_{W} instead. Moreover, in order to simplify notation, we have only displayed one parameter; the general case with an arbitrary list of parameters is completely analogous.

Lemma 46 *There exists a closed $\mathcal{L}_{\mathsf{W}}$ term r_{W} so that $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ proves*

$$f : \mathsf{W} \to \mathsf{W} \wedge g : \mathsf{W}^3 \to \mathsf{W} \;\wedge\; b : \mathsf{W}^2 \to \mathsf{W} \quad \to$$

$$\begin{cases} \mathsf{r}_{\mathsf{W}} fgb : \mathsf{W}^2 \to \mathsf{W} \wedge \\ x \in \mathsf{W} \wedge y \in \mathsf{W} \wedge y \neq \epsilon \wedge h = \mathsf{r}_{\mathsf{W}} fgb \;\to \\ \qquad hx\epsilon = fx \wedge hxy = gxy(hx(\mathsf{p}_{\mathsf{W}} y)) \mid bxy \end{cases}$$

Proof. The crucial strategy of this proof consists in applying the recursion or fixed point lemma (Lemma 44) in order to define the term r_{W} and make subsequent use of $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ in order to establish the required totality or convergence assertion about r_{W}.

We first define t to be the following $\mathcal{L}_{\mathsf{W}}$ term depending on f, g, and b,

$$t \;:=\; \lambda hxy.\mathsf{d}_{\mathsf{W}} f(\lambda z.gzy(hz(\mathsf{p}_{\mathsf{W}} y)) \mid bzy)\epsilon yx,$$

and then set $\mathsf{r}_{\mathsf{W}} \;:=\; \lambda fgb.\mathsf{rec}\, t$. We now have for $h \simeq \mathsf{r}_{\mathsf{W}} fgb$,

$$hxy \simeq \mathsf{rec}\, txy \simeq t(\mathsf{rec}\, t)xy \simeq thxy \simeq \mathsf{d}_{\mathsf{W}} f(\lambda z.gzy(hz(\mathsf{p}_{\mathsf{W}} y)) \mid bzy)\epsilon yx.$$

In particular, we obtain for all x and y in W with $y \neq \epsilon$,

$$hx\epsilon \simeq fx \;\wedge\; hxy \simeq gxy(hx(\mathsf{p}_{\mathsf{W}} y)) \mid bxy. \tag{1}$$

In the following we reason in $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ and assume in addition that

$$f : \mathsf{W} \to \mathsf{W} \;\wedge\; g : \mathsf{W}^3 \to \mathsf{W} \;\wedge\; b : \mathsf{W}^2 \to \mathsf{W}. \tag{2}$$

Our crucial task is to show that indeed $h : \mathsf{W}^2 \to \mathsf{W}$, and this is of course where bounded induction enters the scene. First, let c be an operation so that cxy is simply fx if $y = \epsilon$ and bxy, otherwise. Obviously, we have that $c : \mathsf{W}^2 \to \mathsf{W}$. Now we define $A(y)$ to be the $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formula

$$A(y) \;:=\; (\exists z \leq cxy)(hxy = z).$$

Recall at this point that bounded quantifiers range over W. Fixing the parameter $x \in \mathsf{W}$, it is now a matter of routine to derive from (1) and (2),

$$(3) \qquad A(\epsilon) \ \wedge \ (\forall y \in \mathsf{W})(A(y) \ \rightarrow \ A(\mathsf{s_0}y) \wedge A(\mathsf{s_1}y)).$$

Further, (3) brings us in the position to apply notation induction for $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formulas, $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I_W})$, and we can thus conclude

$$(4) \qquad (\forall y \in \mathsf{W})(\exists z \in \mathsf{W})(z \leq cxy \wedge hxy = z),$$

for an arbitrarily chosen x in W. But (4) shows indeed that we have established h to be an operation from W^2 to W, i.e., $h : \mathsf{W}^2 \rightarrow \mathsf{W}$. This is as claimed and ends our proof. □

We want to emphasize that indeed we have established the existence of a type two functional for bounded recursion on notation in $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I_W})$; this will be the key for interpreting the Cook-Urquhart system PV^ω into PT in the next chapter. At any rate, the previous lemma shows that the functions in FPtime and FPtimeLinspace are provably total in PT and PTLS, respectively. Moreover, observe that in fact we have only used very special instances of $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I_W})$, namely $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I_W})$ has been applied for statements of the form $(\exists z \leq fy)(gy = z)$.

If we replace notation induction on W, $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I_W})$, by lexicographic induction on W, $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$, then of course one expects that we can derive bounded lexicographic recursion (BRL) instead of bounded recursion on notation (BRN). The proof of this fact runs completely analogous to the proof of the previous lemma and is hence omitted. Clearly, the following lemma shows that FPspace and FLinspace are contained in the provably total functions of PS and LS, respectively.

Lemma 47 *There exists a closed $\mathcal{L}_\mathsf{W}$ term r_ℓ so that $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$ proves*

$$f : \mathsf{W} \rightarrow \mathsf{W} \wedge g : \mathsf{W}^3 \rightarrow \mathsf{W} \ \wedge \ b : \mathsf{W}^2 \rightarrow \mathsf{W} \ \ \rightarrow$$

$$\begin{cases} \mathsf{r}_\ell fgb : \mathsf{W}^2 \rightarrow \mathsf{W} \ \wedge \\ x \in \mathsf{W} \wedge y \in \mathsf{W} \wedge y \neq \epsilon \wedge h = \mathsf{r}_\ell fgb \ \rightarrow \\ \qquad hx\epsilon = fx \wedge hxy = gxy(hx(\mathsf{p}_\ell y)) \mid bxy \end{cases}$$

A natural question to ask is whether bounded recursion on notation in the above functional form using the recursor $\mathsf{r_W}$ is directly available in $\mathsf{B}+(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$, too. The answer is indeed positive due to the fact that over the base theory B, lexicographic induction ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_\ell$) entails notation induction ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\mathsf{W}}$). The proof of this fact is completely analogous to the argument showing that Buss' theory T^1_2 contains his system S^1_2, cf. Buss [15]. Nevertheless, since our setting is different, and in some sense simpler, we spell out the relevant arguments in some detail.

Lemma 48 *We have that* ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_\ell$) *entails* ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\mathsf{W}}$) *over our base theory* B.

Proof. Let us work informally in the theory $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$. By the previous lemma, bounded lexicographic recursion is at our disposal. Hence, we can define the well-known "most significant part" function $\mathsf{msp} : \mathsf{W}^2 \to \mathsf{W}$,

$$\mathsf{msp}a\epsilon = a, \quad \mathsf{msp}ab = \mathsf{p}_{\mathsf{W}}(\mathsf{msp}a(\mathsf{p}_\ell b)), \quad \mathsf{msp}ab \leq a,$$

for all a in W and b in W with $b \neq \epsilon$. This function cuts off b bits to the right of a, where b is understood in the sense of the lexicographic ordering $<_\ell$ on W. Further, we have the cut-off operation $\dot{-} : \mathsf{W}^2 \to \mathsf{W}$,

$$a \dot{-} \epsilon = a, \quad a \dot{-} b = \mathsf{p}_\ell(a \dot{-} \mathsf{p}_\ell b), \quad a \dot{-} b \leq a,$$

for $a, b \in \mathsf{W}$ and $b \neq \epsilon$. Finally, the length function $|\cdot| : \mathsf{W} \to \mathsf{W}$, which measures the length of a word by means of the $<_\ell$ ordering can be defined by bounded lexicographic recursion and by making us of the "tally" length function which is available in B,

$$|\epsilon| = \epsilon, \quad |a| = \mathsf{if}\ \mathsf{p}_\ell a < a\ \mathsf{then}\ \mathsf{s}_\ell|\mathsf{p}_\ell a|\ \mathsf{else}\ |\mathsf{p}_\ell a|, \quad |a| \leq a,$$

for all a in W with $a \neq \epsilon$. It is not difficult to see that the usual properties of msp, $\dot{-}$, and $|\cdot|$ are derivable in $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$.

Consider now an arbitrary $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formula $A(x) \equiv (\exists y \leq fx)B(f, x, y)$ and assume the premise of ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\mathsf{W}}$), i.e.,

$$(1) \qquad f : \mathsf{W} \to \mathsf{W} \wedge A(\epsilon) \wedge (\forall x \in \mathsf{W})(A(x) \to A(\mathsf{s}_0 x) \wedge A(\mathsf{s}_1 x)).$$

We fix an $a \in \mathsf{W}$ and aim at showing $A(a)$. For that purpose we let $C(a, x)$ be the formula $A(\mathsf{msp}a(|a| \dot{-} x))$. Observe that since $f : \mathsf{W} \to \mathsf{W}$, we also have $g : \mathsf{W} \to \mathsf{W}$, for g being the operation $\lambda x.f(\mathsf{msp}a(|a| \dot{-} x))$. We can now readily derive from (1),

$$(2) \qquad g : \mathsf{W} \to \mathsf{W} \wedge C(a, \epsilon) \wedge (\forall x \in \mathsf{W})(C(a, x) \to C(a, \mathsf{s}_\ell x)).$$

This brings us in the position to apply $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$ in order to derive the statement $(\forall x \in \mathsf{W})C(a,x)$ and, in particular, $C(a,|a|)$. Clearly, $A(a)$ is entailed by $C(a,|a|)$. We have established in $\mathsf{B}+(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$ the schema of notation induction on W for $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ formulas, $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$. □

Corollary 49 *The assertion of Lemma 46 is derivable in* $\mathsf{B} + (\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$.

Corollary 50 *We have that* PT *and* PTLS *are directly contained in* PS *and* LS*, respectively.*

In this section we have established lower bounds in terms of provably total functions of the four central systems, PT, PTLS, PS, and LS. We collect the corresponding results in the following theorem.

Theorem 51 *We have the following lower bound results:*

1. *The provably total functions of* PT *include* FPtime.
2. *The provably total functions of* PTLS *include* FPtimeLinspace.
3. *The provably total functions of* PS *include* FPspace.
4. *The provably total functions of* LS *include* FLinspace.

Let us finish this chapter by mentioning that the results of this section show that the applicative theories PTO and PTO^+ introduced and analyzed in Strahm [120] are directly contained in our system PT. In particular, the induction principles presented in [120] directly follow from the more general induction principle $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$, and the axioms about bounded recursion on notation in [120] are derivable in PT thanks to Lemma 46.

Chapter 9

Higher types in PT and the system PV^ω

In the last decade intense research efforts have been made in the area of so-called higher type complexity theory and, in particular, feasible functionals of higher types. This research is still ongoing and it is not yet clear what the right higher type analogue of the polynomial time computable functions is. Most prominent in the previous research is the class of so-called *basic feasible functionals* BFF, which has proved to be a very robust class with various kinds of interesting characterizations.

The basic feasible functionals of type 2, BFF_2, were first studied in Melhorn [85]. More than ten years later in 1989, Cook and Urquhart [26] introduced the basic feasible functionals at all finite types in order to provide functional interpretations of feasibly constructive arithmetic; in particular, they defined a typed formal system PV^ω and used it to establish functional and realizability interpretations of an intuitionistic version of Buss' theory S^1_2. The basic feasible functionals BFF are exactly those functionals which can be defined by PV^ω terms. Subsequently, much work has been devoted to BFF, cf. e.g. Cook and Kapron [25, 75], Irwin, Kapron and Royer [52], Pezzoli [91], Royer [102], and Seth [111].

In this chapter we introduce an intensional and an extensional version of the Cook-Urquhart system PV^ω and show that both systems are naturally contained in our applicative system PT. Hence, in a sense, PT *provides a direct justification of* PV^ω *in a type-free applicative setting.* In addition, the embeddings established in the sequel also show that the well-known systems of bounded arithmetic PTCA and PTCA^+ of Ferreira [37, 38] or, equivalently, Cook's system PV [24] and Buss' S^1_2 [15] are directly contained in PT.

9.1 The systems PV^ω and EPV^ω

We start off with defining the collection $\mathcal{T}$ *of finite type symbols* $(\alpha, \beta, \gamma, \ldots)$. $\mathcal{T}$ is inductively generated by the usual clauses, (i) $0 \in \mathcal{T}$, (ii) if $\alpha, \beta \in \mathcal{T}$, then $(\alpha \times \beta) \in \mathcal{T}$, and (iii) if $\alpha, \beta \in \mathcal{T}$, then $(\alpha \to \beta) \in \mathcal{T}$. Hence, we have product and function types as usual. Observe, however, that in our setting the ground type 0 stands for the set of binary words and not for the set of natural numbers. We use the usual convention and write $\alpha_1 \to \alpha_2 \to \cdots \to \alpha_k$ instead of $(\alpha_1 \to (\alpha_2 \to \cdots \to (\alpha_{k-1} \to \alpha_k) \cdots))$.

In the following we sketch a version of PV^ω which is similar in spirit to the presentation of Heyting's arithmetic in all finite types HA^ω in Troelstra and Van Dalen [128]; however, the logic of PV^ω is classical logic. PV^ω is based on combinators and noncommittal as to the exact nature of equality between objects of higher types. Later we will also discuss an extensional version EPV^ω of PV^ω.[1]

The language of PV^ω includes for each type symbol $\alpha \in \mathcal{T}$ a countable collection $x^\alpha, y^\alpha, z^\alpha, u^\alpha, v^\alpha, w^\alpha, \ldots$ of variables of type α. Further, for each $\alpha \in \mathcal{T}$ we have a binary relation symbol $=^\alpha$ for equality at type α, and for all $\alpha, \beta \in \mathcal{T}$ there is an application operator $\cdot^{\alpha,\beta}$. The *constants* of PV^ω first of all include the "arithmetical" constants of $\mathcal{L}_\mathsf{W}$, namely $\epsilon, \mathsf{s}_0, \mathsf{s}_1, \mathsf{p}_\mathsf{W}, \mathsf{s}_\ell, \mathsf{p}_\ell, \mathsf{c}_\subseteq, \mathsf{l}_\mathsf{W}, *$, and $\times$; these constants now receive their obvious types in the typed language of PV^ω. In addition, we have typed versions of $\mathsf{k}, \mathsf{s}, \mathsf{p}, \mathsf{p}_0, \mathsf{p}_1$ as well as d_W, and most importantly, a recursion operator r. More precisely, we have for all types $\alpha, \beta, \gamma \in \mathcal{T}$ the following constants with their associated types:

$$\begin{aligned} \mathsf{p}^{\alpha,\beta} &: \alpha \to \beta \to (\alpha \times \beta), \\ \mathsf{p}_0^{\alpha,\beta} &: (\alpha \times \beta) \to \alpha, \\ \mathsf{p}_1^{\alpha,\beta} &: (\alpha \times \beta) \to \beta, \end{aligned}$$

[1] Actually, the system EPV^ω introduced below corresponds to the Cook-Urquhart system IPV^ω in [26] *with classical logic* instead of intuitionistic logic. What we call PV^ω in this thesis is just an intensional version of EPV^ω. We follow Troelstra and Van Dalen [128] in using this terminology.

$$\begin{aligned}
\mathsf{k}^{\alpha,\beta} &: \alpha \to \beta \to \alpha,\\
\mathsf{s}^{\alpha,\beta,\gamma} &: (\alpha \to \beta \to \gamma) \to (\alpha \to \beta) \to \alpha \to \gamma,\\
\mathsf{d}^{\alpha} &: \alpha \to \alpha \to 0 \to 0 \to \alpha,\\
\mathsf{r} &: 0 \to (0 \to 0 \to 0) \to (0 \to 0) \to 0 \to 0.
\end{aligned}$$

In the sequel we often omit the type superscripts of variables and constants if these are clear from the context or unimportant.

The *terms of* PV^ω are now generated from the variables and constants by the expected clause for application, namely: if t is a term of type $(\alpha \to \beta)$ and s a term of type α, then $(t \cdot^{\alpha,\beta} s)$ is a term of type β. As usual we write (ts) instead of $(t \cdot^{\alpha,\beta} s)$; moreover, outer parenthesis are often dropped, and we make free use of the convention of association to the left when writing applicative terms. The *formulas of* PV^ω are built from the prime formulas $(t =^\alpha s)$ for t, s of type α, by means of $\neg$, $\wedge$, $\vee$, $\to$, $(\forall x^\alpha)$, and $(\exists x^\alpha)$. As in the applicative setting above, we call a formula *positive*, if it is implication and negation free.

The logic of PV^ω is many-sorted *classical* predicate calculus with equality. The non-logical axioms of PV^ω include the defining axioms for the constants of PV^ω: these consist of (i) the defining axioms for the "arithmetical" constants of PV^ω, which are just the obvious rewriting to the typed setting of the corresponding axioms of B, and (ii) the following axioms for the combinators $\mathsf{k}, \mathsf{s}, \mathsf{p}, \mathsf{p}_0, \mathsf{p}_1$ and r:

$$\begin{array}{lll}
\mathsf{k}xy = x, & \mathsf{s}xyz = xz(yz), & \\
\mathsf{p}_0(\mathsf{p}xy) = x, & \mathsf{p}_1(\mathsf{p}xy) = y, & \mathsf{p}(\mathsf{p}_0 z)(\mathsf{p}_1 z) = z,\\
\mathsf{d}xyuu = x, & u \neq v \to \mathsf{d}xyuv = y, & \\
\mathsf{r}xyz\epsilon = x, & u \neq \epsilon \to \mathsf{r}xyzu = yu(\mathsf{r}xyz(\mathsf{p_W}u)) \mid zu. &
\end{array}$$

In the defining equations for r, the cut-off operator $\mid$ is understood in the same way as in the untyped applicative setting via the definition by cases operator d and the characteristic function $\mathsf{c}_\subseteq$. We have that r provides a type two functional for bounded recursion on notation in the natural expected manner.

Last but not least, the system PV^ω includes induction on notation,

$$A(\epsilon) \wedge (\forall x^0)(A(x) \to A(\mathsf{s}_0 x) \wedge A(\mathsf{s}_1 x)) \;\to\; (\forall x^0)A(x),$$

for all formulas $A(x)$ in the language of the system PV^ω which have the shape $(\exists y \leq tx)B(x,y)$, with B being a positive and quantifier free formula and t a term of type $(0 \to 0)$.

As usual, the availability of the typed combinators k and s allows for the definition of simply typed λ terms $(\lambda x^\alpha.t)$, for each type symbol $\alpha \in \mathcal{T}$. The definition follows the usual pattern, cf. e.g. [128].

In a further step we now turn to an extensional version EPV^ω of PV^ω. The *extensionality axioms* $(\mathsf{Ext}_{\alpha,\beta})$ for all $\alpha, \beta \in \mathcal{T}$ are given in the expected manner by

$$(\mathsf{Ext}_{\alpha,\beta}) \qquad (\forall y, z)[(\forall x)(yx = zx) \to y = z],$$

for y, z of type $(\alpha \to \beta)$ and x of type α. Now EPV^ω is defined in the same way as PV^ω, except that (i) it includes $(\mathsf{Ext}_{\alpha,\beta})$ for all $\alpha, \beta \in \mathcal{T}$, and (ii) the induction formulas $(\exists y \leq tx)B(x,y)$ of PV^ω are restricted in EPV^ω to positive quantifier free formulas B *not containing equalities of higher type.*

To conclude this section, let us mention that the system PTCA^+ of Ferreira [37, 38] is directly contained in PV^ω. It corresponds to Buss' [15] famous bounded arithmetic theory S^1_2.

9.2 Embedding PV^ω and EPV^ω into PT

Recall that in PV^ω we do not claim that equality $=^\alpha$ for α a higher type is extensional equality. Accordingly, we now sketch an embedding of PV^ω into PT by means of the abstract *intensional type structure* $\langle(\mathsf{IT}_\alpha, =)\rangle_{\alpha\in\mathcal{T}}$. This embedding is analogous to the embedding of HA^ω into the theory of operations and numbers APP in [128]. We work in the applicative language $\mathcal{L}_\mathsf{W}$ and define IT_α inductively as follows:

$$\begin{aligned}
x \in \mathsf{IT}_0 \quad &:= \quad x \in \mathsf{W}, \\
x \in \mathsf{IT}_{\alpha\times\beta} \quad &:= \quad \mathsf{p_0}x \in \mathsf{IT}_\alpha \wedge \mathsf{p_1}x \in \mathsf{IT}_\beta \wedge \mathsf{p}(\mathsf{p_0}x)(\mathsf{p_1}x) = x, \\
x \in \mathsf{IT}_{\alpha\to\beta} \quad &:= \quad (\forall y \in \mathsf{IT}_\alpha)(xy \in \mathsf{IT}_\beta).
\end{aligned}$$

Equality in IT_α is simply the restriction of equality in PT. We now get an embedding $(\cdot)^\mathsf{IT}$ of PV^ω into PT by letting the variables of type α range over IT_α. Further, application $\cdot^{\alpha,\beta}$ in PV^ω carries over to application $\cdot$ in PT, restricted to $\mathsf{IT}_{\alpha\to\beta} \times \mathsf{IT}_\alpha$. Moreover, the constants of PV^ω different from r are interpreted by the corresponding constants in $\mathcal{L}_\mathsf{W}$. The recursor r of PV^ω can be interpreted, for example, by the closed $\mathcal{L}_\mathsf{W}$ term $\lambda xyzu.\mathsf{r}_\mathsf{W}(\mathsf{k}x)(\mathsf{k}y)(\mathsf{k}z)\epsilon u$, where r_W denotes the closed term stated in the assertion of Lemma 46. We now have the following embedding theorem.

Theorem 52 *We have for all sentences A that $\mathsf{PV}^\omega \vdash A$ entails $\mathsf{PT} \vdash A^{\mathsf{IT}}$.*

Proof. The proof of the theorem is immediate except for the case of recursion and induction in PV^ω. But the defining axioms for r and, more importantly, the fact the r has the right type, are readily derivable in PT by the results of Lemma 46. Moreover, the translation of notation induction in PV^ω directly carries over to ($\Sigma^\mathsf{b}_\mathsf{W}$-$\mathsf{I}_\mathsf{W}$) in PT; for, a formula of the form $(\exists y \leq tx)B(x,y)$ with t of type $(0 \rightarrow 0)$ and B positive and quantifier free directly translates into a $\Sigma^\mathsf{b}_\mathsf{W}$ formula in the untyped applicative setting of PT. □

Next we also give an embedding of EPV^ω into our type-free applicative setting PT, which is analogous to the embedding of an extensional version EHA^ω of HA^ω into APP in [128]. In this embedding we now make use of an abstract *extensional type structure* $\langle(\mathsf{ET}_\alpha, =_\alpha)\rangle_{\alpha\in\mathcal{T}}$ in $\mathcal{L}_\mathsf{W}$, cf. [128]. ET_α and $=_\alpha$ are inductively given in the following manner:

$$
\begin{aligned}
x \in \mathsf{ET}_0 &\quad := \quad x \in \mathsf{W},\\
x =_0 y &\quad := \quad x \in \mathsf{W} \wedge y \in \mathsf{W} \wedge x = y,\\
x \in \mathsf{ET}_{\alpha\times\beta} &\quad := \quad \mathsf{p}_0 x \in \mathsf{ET}_\alpha \wedge \mathsf{p}_1 x \in \mathsf{ET}_\beta \wedge \mathsf{p}(\mathsf{p}_0 x)(\mathsf{p}_1 x) = x,\\
x =_{\alpha\times\beta} y &\quad := \quad (\mathsf{p}_0 x =_\alpha \mathsf{p}_0 y) \wedge (\mathsf{p}_1 x =_\alpha \mathsf{p}_1 y),\\
x \in \mathsf{ET}_{\alpha\rightarrow\beta} &\quad := \quad (\forall y,z)(y =_\alpha z \rightarrow xy =_\beta xz),\\
x =_{\alpha\rightarrow\beta} y &\quad := \quad x \in \mathsf{ET}_{\alpha\rightarrow\beta} \wedge y \in \mathsf{ET}_{\alpha\rightarrow\beta} \wedge (\forall z \in \mathsf{ET}_\alpha)(xz =_\beta yz).
\end{aligned}
$$

EPV^ω can now be interpreted into PT via an embedding $(\cdot)^{\mathsf{ET}}$ in the same way as we have embedded PV^ω into PT via $(\cdot)^{\mathsf{IT}}$ above. Therefore, we omit the proof of the following theorem.

Theorem 53 *We have for all sentences A that $\mathsf{EPV}^\omega \vdash A$ entails $\mathsf{PT} \vdash A^{\mathsf{ET}}$.*

Let us observe that if we interpret PT in its standard recursion-theoretic model of partial recursive operations *PRO*, then IT and ET correspond to the so-called *hereditarily recursive operations HRO* and *hereditarily effective operations HEO*, respectively, cf. [128]. *HRO* forms the standard recursion-theoretic model of PV^ω and *HEO* is the corresponding interpretation of EPV^ω.

We finish this chapter with the observation that the results of the previous chapter give rise immediately to higher type systems for FPtimeLinspace, FPspace, and FLinspace, which are naturally contained in the corresponding type-free settings PTLS, PS and LS, respectively. For example, the type system for FPspace has a type two recursor for bounded lexicographic recursion, which is available in PS by Lemma 47. It might be of interest to study these type systems from the recursion-theoretic and abstract machine point of view.

Chapter 10

Realizing positive derivations

It is the aim of this chapter to establish proof-theoretic upper bounds of PT, PTLS, PS, and LS. Namely, we will show that the lower bounds with respect to provably total functions derived in Theorem 51 are indeed sharp.

For our upper bound arguments we will proceed in two steps. First, a *partial cut elimination* argument in a sequent-style reformulation of our four systems is employed in order to show that as far as the computational content of our systems is concerned, we can restrict ourselves to positive derivations, i.e., sequent style proofs using positive formulas only.

In a second crucial step we use a notion of *realizability for positive formulas* in the standard open term model of our systems: quasi cut-free positive sequent derivations of PT, PTLS, PS, and LS are suitably realized by word functions in FPtime, FPtimeLinspace, FPspace, and FLinspace, respectively, thus yielding the desired computational information concerning the provably total functions of these systems.

10.1 Adding totality and extensionality

Actually, in the following we will establish our upper bounds for slight strengthenings of PT, PTLS, PS, and LS. Namely, we augment our applicative frameworks by the axioms (Tot) for *totality of application* and (Ext) for *extensionality of operations*,

(Tot) $\quad (\forall x, y)(xy\downarrow) \qquad$ (Ext) $\quad (\forall f, g)[(\forall x)(fx = gx) \rightarrow f = g]$

We observe that $\mathsf{B} + (\mathsf{Tot})$ proves $t\downarrow$ for each term t, so that in the presence of (Tot) the logic of partial terms reduces to ordinary classical predicate calculus. Accordingly, if T denotes one of the systems PT, PTLS, PS, or LS, then we write T^+ for the system T based on ordinary classical logic with equality and augmented with the axiom of extensionality (Ext). Observe that in the setting of T^+ we no longer have the relation symbol $\downarrow$, so that instead of axiom (2) of B we simply have the usual total version of the s combinator, given by the axiom $\mathsf{s}xyz = xz(yz)$.

The simplest model of T^+ is just the standard open term model $\mathcal{M}(\lambda\eta)$, which is based on a straightforward extension of usual $\lambda\eta$ reduction. We will discuss this model is some more detail below, where it will be used in our realizability interpretation of (the positive fragment of) T^+.

The fact that the presence of (Tot) and (Ext) does not raise the strength of a given partial applicative system is not too surprising as is witnessed by the previous work on applicative theories. For sample references cf. Cantini [17, 20] and Jäger and Strahm [68].

10.2 Preparatory partial cut elimination

In this section we turn to a preparatory partial cut elimination argument for T^+, where again T denotes any of the systems PT, PTLS, PS, or LS. For that purpose, we will make use of a reformulation of T^+ in terms of Gentzen's classical sequent calculus LK; in the sequel we assume that the reader is familiar with LK as it is presented, for example, in Girard [42].

In the following we let $\Gamma, \Delta, \Lambda, \ldots$ range over finite *sequences* of formulas in the language $\mathcal{L}_{\mathsf{W}}$; a *sequent* is a formal expression of the form $\Gamma \Rightarrow \Delta$. As usual, the natural interpretation of the sequent $A_1, \ldots, A_n \Rightarrow B_1, \ldots, B_m$ is $(A_1 \wedge \cdots \wedge A_n) \rightarrow (B_1 \vee \cdots \vee B_m)$.

We are now aiming at a suitable sequent-style reformulation of T^+. As mentioned above, our crucial aim is to prove a partial cut elimination theorem so that the only cuts occurring in partially cut free derivations have *positive cut formulas*. Hence, in order to solve this task, we must find a Gentzen-style reformulation of T^+ so that all the main formulas of non-logical axioms and rules (including equality) are positive. In the following we sketch such a reformulation of T^+; we are confining ourselves to the essential points without spelling out each single axiom and rule in detail.

The axioms of our basic theory of operations and words, B, are easily reformulated in positive form. Just to give an example, axioms (4) and (5) about definition by cases $\mathsf{d_W}$ on W translate into the pair of sequents

$$\mathsf{W}(r),\, \mathsf{W}(s),\, r = s \;\Rightarrow\; \mathsf{d_W} t_1 t_2 r s = t_1,$$
$$\mathsf{W}(r),\, \mathsf{W}(s) \;\Rightarrow\; r = s,\, \mathsf{d_W} t_1 t_2 r s = t_2,$$

for all terms r, s, t_1, t_2 of $\mathcal{L}_\mathsf{W}$. Observe that as usual in sequent formulations, we take all substitution instances of the axioms of B. It is a matter of routine to spell out in positive sequent form the other axioms of B. In some cases, an axiom has to be split into several sequents, e.g., axiom (18) about the initial subword relation is now given by the two sequents

$$\mathsf{W}(s),\, \mathsf{W}(t),\, s \subseteq t \;\Rightarrow\; t = \epsilon,\, s \subseteq \mathsf{p_W} t,\, s = t,$$
$$\mathsf{W}(s),\, \mathsf{W}(t),\, s \subseteq \mathsf{p_W} t \vee s = t \;\Rightarrow\; t = \epsilon,\, s \subseteq t.$$

We leave it to the reader to provide suitable positive sequents of the other axioms of B, and also of the axioms (25)–(30) concerning word concatenation and word multiplication. Moreover, the extensionality axiom (Ext) of T^+ now simply takes the positive sequent form

$$(\forall x)(sx = tx) \;\Rightarrow\; s = t,$$

for s and t being arbitrary terms in our applicative language $\mathcal{L}_\mathsf{W}$, not containing the variable x. Of course, T^+ also includes the usual equality axioms; clearly, these can be stated in positive sequent form as follows:

$$\Rightarrow\; t = t \qquad s = t \;\Rightarrow\; t = s \qquad s = t,\, t = r \;\Rightarrow\; s = r,$$
$$s_1 = t_1,\, s_2 = t_2 \;\Rightarrow\; s_1 s_2 = t_1 t_2 \qquad s = t,\, \mathsf{W}(s) \;\Rightarrow\; \mathsf{W}(t).$$

Let us now turn to the reformulation of the schemas $(\Sigma^\mathsf{b}_\mathsf{W}\text{-}\mathsf{I_W})$ and $(\Sigma^\mathsf{b}_\mathsf{W}\text{-}\mathsf{I}_\ell)$ of $\Sigma^\mathsf{b}_\mathsf{W}$ notation induction on W and lexicographic induction on W, respectively. These will be replaced by suitable rules of inference in the Gentzen-style formulation of T^+. Let $A(u)$ be of the form $(\exists y \leq tu)B(u, y)$ for B being a positive and W free formula. Then an instance of the $(\Sigma^\mathsf{b}_\mathsf{W}\text{-}\mathsf{I_W})$ notation induction rule is given as follows:

$$\frac{\Gamma,\, \mathsf{W}(u) \;\Rightarrow\; \mathsf{W}(tu),\, \Delta \quad \Gamma \;\Rightarrow\; A(\epsilon),\, \Delta \quad \Gamma,\, \mathsf{W}(u),\, A(u) \;\Rightarrow\; A(\mathsf{s}_i u),\, \Delta}{\Gamma,\, \mathsf{W}(s) \;\Rightarrow\; A(s),\, \Delta}$$

Here u denotes a fresh variable not occurring in Γ, Δ and i ranges over $0, 1$, i.e., the rule has four premises. Clearly, the main formulas of this rule are positive. We do not need to spell out the corresponding rule for $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$ lexicographic induction, since it is formulated in the very same manner except that it uses the successor s_ℓ instead of s_0 and s_1, thus only having three premises.

This ends the Gentzen-style reformulation of the non-logical axioms and rules of T^+. The logical axioms and rules of T^+ are just the usual ones for Gentzen's LK, cf. e.g. [42]. I.e., we have identity axioms, the well-known logical rules for introducing $\wedge, \vee, \neg, \rightarrow, \forall$ and $\exists$ on the right-hand side and on the left-hand side, the structural rules for weakening, exchange, and contraction, as well as the cut rule. In contrast to [42], however, we are using the so-called context-sharing or additive versions of these rules: this means that rules of inference with several premises are using the same context; we have already used this convention in the formulation of the induction rules above. To give a further example, the cut rule in its context-sharing version takes the form

$$\frac{\Gamma, A \Rightarrow \Delta \qquad \Gamma \Rightarrow A, \Delta}{\Gamma \Rightarrow \Delta}$$

As usual, we call the formula A the *cut formula* of this cut. We do not spell out all the rules of LK at the moment and refer the reader to the proofs of the realizability theorems in the next section, where some of these rules will be treated in all detail.

It should be clear that we have provided an adequate sequent-style reformulation of T^+; in particular, the axioms schemas $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ and $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$ as given in Section 7.2 of this thesis are readily derivable by means of the corresponding rules of inference stated above, where as usual the presence of side formulas is crucial. In the following we often identify T^+ with its Gentzen-style version and write $\mathsf{T}^+ \vdash \Gamma \Rightarrow \Delta$ in order to express that the sequent $\Gamma \Rightarrow \Delta$ is derivable in T^+. Moreover, we will use the notation $\mathsf{T}^+ \vdash_{\star} \Gamma \Rightarrow \Delta$ if the sequent $\Gamma \Rightarrow \Delta$ has a proof in T^+ so that all cut formulas appearing in this proof are *positive*.

Due to the fact that all the main formulas of non-logical axioms and rules of T^+ are positive, we now obtain the desired partial cut elimination theorem for T^+. Its proof is immediate from the well-known proof of the cut elimination theorem for LK and is therefore omitted.

Theorem 54 (Partial cut elimination for T^+) *We have for all sequents* $\Gamma \Rightarrow \Delta$ *that* $\mathsf{T}^+ \vdash \Gamma \Rightarrow \Delta$ *entails* $\mathsf{T}^+ \vdash_{\star} \Gamma \Rightarrow \Delta$.

The following corollary directly follows from the above theorem and a quick inspection of the axioms and rules of T^+. It will be crucial for our realizability arguments below.

Corollary 55 *Assume that* $\Gamma \Rightarrow \Delta$ *is a sequent of positive formulas so that* $\mathsf{T}^+ \vdash \Gamma \Rightarrow \Delta$. *Then* $\Gamma \Rightarrow \Delta$ *has a* T^+ *derivation containing positive formulas only.*

10.3 The realizability theorems

In this section we use a realizability interpretation in the term model $\mathcal{M}(\lambda\eta)$ in order to determine the computational content of sequent-style derivations in the positive fragment of PT, PTLS, PS, and LS, respectively. We will show that the crucial realizing functions for our four systems belong to the corresponding function complexity classes on binary words, FPTIME, FPTIMELINSPACE, FPSPACE, and FLINSPACE. As immediate corollaries of the four realizability theorems below we obtain the desired upper bounds for the provably total functions of PT, PTLS, PS, and LS.

The notion of realizability as well as the style and spirit of our realizability theorems are related to the work of Leivant [77], Schlüter [108], and Cantini [19, 21], all three in the context of FPTIME. However, in contrast to these papers, we work in a bounded unramified setting. Moreover, and this is similar to [21, 108], we are able to realize directly quasi cut-free positive derivations in the *classical* sequent calculus. Finally, in order to find our realizing functions, we can make direct use of the function algebra characterizations of FPTIME, FPTIMELINSPACE, FPSPACE, and FLINSPACE given in Theorem 42; hence, direct reference to a machine model is not needed.

In our definition of realizability below we will make use of the open term model $\mathcal{M}(\lambda\eta)$ of T^+. This model is based on the usual $\lambda\eta$ reduction of the untyped lambda calculus (cf. [6, 48]) and exploits the well-known equivalence between combinatory logic with extensionality and $\lambda\eta$. In order to deal with the constants different from k and s, one extends $\lambda\eta$ reduction by the obvious reduction clauses for these new constants and checks that the so-obtained new reduction relation enjoys the Church Rosser property.[1]

The universe of the model $\mathcal{M}(\lambda\eta)$ now consists of the set of all $\mathcal{L}_{\mathsf{W}}$ terms. Equality $=$ means reduction to a common reduct and W is interpreted as the set of all $\mathcal{L}_{\mathsf{W}}$ terms t so that t reduces to a "standard" word $\overline{w}$ for some $w \in \mathbb{W}$. Finally, the constants are interpreted as indicated above and application of t to s is simply the term ts. As usual, we write $\mathcal{M}(\lambda\eta) \models A$ in order to express that the formula A is true in $\mathcal{M}(\lambda\eta)$.

We are now ready to turn to realizability. Our realizers $\rho, \sigma, \tau, \dots$ are simply elements of the set $\mathbb{W}$ of binary words. We presuppose a low-level pairing operation $\langle \cdot, \cdot \rangle$ on $\mathbb{W}$ with associated projections $(\cdot)_0$ and $(\cdot)_1$; for definiteness, we assume that $\langle \cdot, \cdot \rangle$, $(\cdot)_0$, and $(\cdot)_1$ are in FPtimeLinspace. Further, for each natural number i let us write i_2 for the binary notation of i.

Since we are only interested in realizing *positive* derivations, we need to define realizability only for positive formulas. Accordingly, the crucial notion $\rho \ \mathbf{r}\ A$ ("ρ realizes A") for $\rho \in \mathbb{W}$ and A a positive formula, is given inductively in the following manner.

$$
\begin{array}{lll}
\rho \ \mathbf{r}\ \mathsf{W}(t) & \text{if} & \mathcal{M}(\lambda\eta) \models t = \overline{\rho}, \\
\rho \ \mathbf{r}\ (t_1 = t_2) & \text{if} & \rho = \epsilon \text{ and } \mathcal{M}(\lambda\eta) \models t_1 = t_2, \\
\rho \ \mathbf{r}\ (A \wedge B) & \text{if} & \rho = \langle \rho_0, \rho_1 \rangle \text{ and } \rho_0 \ \mathbf{r}\ A \text{ and } \rho_1 \ \mathbf{r}\ B, \\
\rho \ \mathbf{r}\ (A \vee B) & \text{if} & \rho = \langle i, \rho_0 \rangle \text{ and either } i = 0 \text{ and } \rho_0 \ \mathbf{r}\ A \text{ or} \\
 & & \qquad\qquad\qquad\quad i = 1 \text{ and } \rho_0 \ \mathbf{r}\ B, \\
\rho \ \mathbf{r}\ (\forall x)A(x) & \text{if} & \rho \ \mathbf{r}\ A(u) \text{ for a fresh variable } u, \\
\rho \ \mathbf{r}\ (\exists x)A(x) & \text{if} & \rho \ \mathbf{r}\ A(t) \text{ for some term } t.
\end{array}
$$

If Δ denotes the sequence $A_1, \dots, A_n$ of positive formulas, then we say that ρ realizes the sequence Δ, in symbols, $\rho \ \mathbf{r}\ \Delta$, if $\rho = \langle i_2, \rho_0 \rangle$ for some $1 \le i \le n$ and $\rho_0 \ \mathbf{r}\ A_i$. Hence, according to the notion $\rho \ \mathbf{r}\ \Delta$, the sequence Δ is understood disjunctively, i.e. as the succedent of a given sequent.

It is important to note that in our definition of realizability, the realizers ρ mainly control information concerning the predicate W and, in addition, the usual information concerning conjunction and disjunction. However, the above notion of realizability *trivializes* quantifiers over arbitrary individuals.

[1]Actually, suitable interpretations for the constants $\mathsf{s}_\ell, \mathsf{p}_\ell, \mathsf{c}_\subseteq, \mathsf{l}_\mathsf{W}, *$ and $\times$ can also be given using the other constants of $\mathcal{L}_\mathsf{W}$.

The following properties concerning substitution will be crucial in the proof of the realizability theorem below. The proof of the following lemma is immediate from the definition of realizability and will therefore be omitted.

Lemma 56 (Substitution) *We have for all positive formulas A, all variables u and all terms s and t:*

1. *If ρ $\mathbf{r}$ $A(t)$ and $\mathcal{M}(\lambda\eta) \models t = s$, then ρ $\mathbf{r}$ $A(s)$.*
2. *If ρ $\mathbf{r}$ $A(u)$, then ρ $\mathbf{r}$ $A(t)$.*

Let us introduce some final piece of notation before we state the realizability theorem for PT. For an $\mathcal{L}_{\mathsf{W}}$ formula A we write $A[\vec{u}]$ in order to express that all the free variables occurring in A are contained in the list $\vec{u}$. The analogous convention is used for finite sequences of $\mathcal{L}_{\mathsf{W}}$ formulas.

Theorem 57 (Realizability for PT^+) *Let $\Gamma \Rightarrow \Delta$ be a sequent of positive formulas with $\Gamma = A_1, \dots, A_n$ and assume that $\mathsf{PT}^+ \mathrel{|\!\!\frac{\quad}{\star}} \Gamma[\vec{u}] \Rightarrow \Delta[\vec{u}]$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in* FPTIME *so that we have for all terms $\vec{s}$ and all $\rho_1, \dots, \rho_n \in \mathbb{W}$:*

$$\textit{For all } 1 \le i \le n : \rho_i \ \mathbf{r} \ A_i[\vec{s}] \quad \Longrightarrow \quad F(\rho_1, \dots, \rho_n) \ \mathbf{r} \ \Delta[\vec{s}].$$

Proof. We will prove our claim by induction on the length of quasi cut-free derivations of sequents of positive formulas in PT^+. In order to show that our realizing functions are in FPTIME we make use of the function algebra characterization of FPTIME given in Theorem 42. It is important that our realizing functions are invariant under substitutions of terms $\vec{s}$ for the free variables $\vec{u}$ in the sequent $\Gamma[\vec{u}] \Rightarrow \Delta[\vec{u}]$. This fact is always immediate and, therefore, in order to simplify notation, we often suppress substitutions in our discussion of the various axioms and rules below.

We start with a discussion of the logical axioms and rules of our sequent calculus LK. In the case of an identity axiom $A \Rightarrow A$ for A being a positive formula, we simply choose the function F with $F(\rho) = \langle 1, \rho \rangle$ as our realizing function so that our claim is immediate.

Let us turn to rules for conjunction introduction on the right and on the left. If our last inference is of the form

$$\frac{\Gamma \Rightarrow A, \Delta \qquad \Gamma \Rightarrow B, \Delta}{\Gamma \Rightarrow A \wedge B, \Delta},$$

and F_0 and F_1 are the two realizing functions for the left and the right premise of this rule, respectively, given to us by the induction hypothesis, then we define the realizing function F for the conclusion of the rule by

$$F(\vec{\rho}) \;=\; \begin{cases} F_0(\vec{\rho}) & \text{if } F_0(\vec{\rho})_0 \neq 1, \\ F_1(\vec{\rho}) & \text{if } F_0(\vec{\rho})_0 = 1 \text{ and } F_1(\vec{\rho})_0 \neq 1, \\ \langle 1, \langle F_0(\vec{\rho})_1, F_1(\vec{\rho})_1\rangle\rangle & \text{otherwise.} \end{cases}$$

In the case of introduction of $\wedge$ on the left, i.e., if we have derived the sequent $\Gamma,\, A \wedge B \;\Rightarrow\; \Delta$ from $\Gamma, A \;\Rightarrow\; \Delta$ or $\Gamma,\, B \;\Rightarrow\; \Delta$, we choose $F(\vec{\rho}, \sigma)$ to be $F_0(\vec{\rho}, (\sigma)_0)$, respectively $F_0(\vec{\rho}, (\sigma)_1)$, for F_0 being the realizing function for the corresponding premise.

Next we discuss the rules for introducing a disjunction on the left and on the right. We first assume that our last inference is of the form

$$\frac{\Gamma \;\Rightarrow\; A,\, \Delta}{\Gamma \;\Rightarrow\; A \vee B,\, \Delta},$$

and we let F_0 denote the function given by the induction hypothesis. Then the realizing function F for the conclusion of this rule is given as follows:

$$F(\vec{\rho}) \;=\; \begin{cases} F_0(\vec{\rho}) & \text{if } F_0(\vec{\rho})_0 \neq 1, \\ \langle 1, \langle 0, F_0(\vec{\rho})_1\rangle\rangle & \text{otherwise.} \end{cases}$$

The dual rule for introducing $\vee$ on the right is treated similarly. Now assume that our derivation ends with the rule

$$\frac{\Gamma,\, A \;\Rightarrow\; \Delta \qquad \Gamma,\, B \;\Rightarrow\; \Delta}{\Gamma,\, A \vee B \;\Rightarrow\; \Delta},$$

and let F_0 and F_1 be our realizing functions for the premises of this rule. Then we can simply define F by

$$F(\vec{\rho}, \sigma) \;=\; \begin{cases} F_0(\vec{\rho}, (\sigma)_1) & \text{if } (\sigma)_0 = 0, \\ F_1(\vec{\rho}, (\sigma)_1) & \text{otherwise.} \end{cases}$$

This ends our discussion of $\vee$ introduction. Observe that we do not have to consider introduction rules for negation and implication, since we are working in the positive fragment of PT^+.

We now address the quantification rules of LK. The introduction rules for universal quantification on the right and on the left have their usual form,

$$\frac{\Gamma \;\Rightarrow\; A(u),\, \Delta}{\Gamma \;\Rightarrow\; (\forall x)A(x),\, \Delta} \qquad\qquad \frac{\Gamma,\, A(t) \;\Rightarrow\; \Delta}{\Gamma,\, (\forall x)A(x) \;\Rightarrow\; \Delta},$$

for u a "fresh" variable and t an arbitrary term. Letting F_0 denote the function realizing the premise of these rules, it is straightforward to see that we can simply take $F = F_0$ for the function realizing the conclusion of the corresponding rule, since our definition of realizability trivializes quantifiers. In the case of the second of the above rules we further use the fact that our notion of realizability is closed under substitution (Lemma 56). Finally, it is easily seen that the choice $F = F_0$ also works equally well for the two introduction rules for the existential quantifiers, namely

$$\frac{\Gamma \Rightarrow A(t), \Delta}{\Gamma \Rightarrow (\exists x)A(x), \Delta} \qquad \frac{\Gamma, A(u) \Rightarrow \Delta}{\Gamma, (\exists x)A(x) \Rightarrow \Delta}.$$

In a further step we have to convince ourselves how to realize the structural rules of LK, namely *weakening, exchange* and *contraction.* As these rules are realized in a rather straightforward manner, we leave the details as an exercise to the devoted reader.

We conclude our discussion of the logical axioms and rules by considering the cut rule. Hence, by assumption, there exists a *positive* formula A so that our derivation ends by an application of the rule

$$\frac{\Gamma, A \Rightarrow \Delta \qquad \Gamma \Rightarrow A, \Delta}{\Gamma \Rightarrow \Delta}$$

By induction hypothesis we are given realizing functions F_0 and F_1 for the left and the right premise of this rule, respectively. We now obtain a realizing function F for $\Gamma \Rightarrow \Delta$ by setting

$$F(\vec{\rho}) \;=\; \begin{cases} F_1(\vec{\rho}) & \text{if } F_1(\vec{\rho})_0 \neq 1, \\ F_0(\vec{\rho}, F_1(\vec{\rho})_1) & \text{otherwise.} \end{cases}$$

Let us now turn to the non-logical axioms and rules of PT^+. First of all, it is quite easy to find realizing functions for the positive sequents corresponding to the axioms of $\mathsf{B}(*, \times)$. Instead of discussing all cases in detail we confine ourselves to looking at a few examples.

Clearly, sequents corresponding to true equations in the term model $\mathcal{M}(\lambda\eta)$ such as $\Rightarrow \mathsf{s}t_1t_2t_3 = t_1t_3(t_2t_3)$ are simply realized by the 0-ary function $F = \langle 1, \epsilon \rangle$. Further, for the two sequents given in the previous paragraph for definition by cases on W we can simply take the two realizing functions $F_0(\rho, \sigma, \tau) = \langle 1, \epsilon \rangle$ as well as

$$F_1(\rho, \sigma) \;=\; \begin{cases} \langle 1, \epsilon \rangle & \text{if } \rho = \sigma, \\ \langle 2_2, \epsilon \rangle & \text{otherwise,} \end{cases}$$

respectively. Further, in order to realize the two sequents corresponding to axioms (25) and (28) concerning the totality of word concatenation and word multiplication, namely

$$\mathsf{W}(s),\, \mathsf{W}(t) \;\Rightarrow\; \mathsf{W}(s{*}t), \qquad \mathsf{W}(s),\, \mathsf{W}(t) \;\Rightarrow\; \mathsf{W}(s{\times}t),$$

the two functions $F_0(\rho, \sigma) = \langle 1, \rho{*}\sigma \rangle$ and $F_1(\rho, \sigma) = \langle 1, \rho{\times}\sigma \rangle$ do the job. Also, it is easy to see how to realize the equality axioms. E.g., the sequent $s = t,\, \mathsf{W}(s) \;\Rightarrow\; \mathsf{W}(t)$ can be realized by the function $F(\rho, \sigma) = \sigma$.

Recall that PT^+ also includes an extensionality axiom for our notion of equality $=$, which we have formalized by the sequent $(\forall x)(sx = tx) \;\Rightarrow\; s = t$. Also this sequent is easily seen to realizable by the function $F(\rho) = \langle 1, \epsilon \rangle$.

Let us now turn to the crucial part of the proof, namely the treatment of the rule for $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ notation induction on W. According to the four premises of $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ induction, we have quasi cut-free PT^+ derivations of the four sequents

$$\begin{aligned}
&\Gamma,\, \mathsf{W}(u) \;\Rightarrow\; \mathsf{W}(tu),\, \Delta,\\
&\Gamma \;\Rightarrow\; A(\epsilon),\, \Delta,\\
&\Gamma,\, \mathsf{W}(u),\, A(u) \;\Rightarrow\; A(\mathsf{s}_i u),\, \Delta, \quad (i = 0, 1)
\end{aligned}$$

for $A(u)$ being of the form $(\exists y \leq tu)B(u, y)$ with B positive and W free. Hence, the induction hypothesis guarantees the existence of four FPtime functions F, G_ϵ, G_0, and G_1 on $\mathbb{W}$, so that we have for all $\mathcal{L}_{\mathsf{W}}$ terms $\vec{s}$ and all binary words $\vec{\rho}, \sigma, \tau$,

(1) $\quad \vec{\rho}\ \mathbf{r}\ \Gamma[\vec{s}] \quad \Longrightarrow \quad F(\vec{\rho}, \sigma)\ \mathbf{r}\ \mathsf{W}(t[\vec{s}](\sigma)),\, \Delta[\vec{s}],$[2]

(2) $\quad \vec{\rho}\ \mathbf{r}\ \Gamma[\vec{s}] \quad \Longrightarrow \quad G_\epsilon(\vec{\rho})\ \mathbf{r}\ A[\vec{s}, \epsilon],\, \Delta[\vec{s}],$

(3) $\quad \vec{\rho}\ \mathbf{r}\ \Gamma[\vec{s}],\, \tau\ \mathbf{r}\ A[\vec{s}, \sigma] \quad \Longrightarrow \quad G_i(\vec{\rho}, \sigma, \tau)\ \mathbf{r}\ A[\vec{s}, \mathsf{s}_i\sigma],\, \Delta[\vec{s}] \quad (i = 0, 1)$

It is our crucial aim to find a realizing function for the conclusion of the induction rule, i.e., a polynomial time computable function H so that we have for all $\vec{\rho}, \sigma$ in $\mathbb{W}$,

(4) $\qquad \vec{\rho}\ \mathbf{r}\ \Gamma[\vec{s}] \quad \Longrightarrow \quad H(\vec{\rho}, \sigma)\ \mathbf{r}\ A[\vec{s}, \sigma],\, \Delta[\vec{s}].$

[2]Temporarily in this proof, if $\Gamma = C_1, \ldots, C_m$ is a sequence of formulas contained in the antecedent of a sequent, then we write $\rho_1, \ldots, \rho_m\ \mathbf{r}\ \Gamma$ if for all $1 \leq i \leq m$, $\rho_i\ \mathbf{r}\ C_i$.

Our desired word function H is defined for all $\vec{\rho}$ and σ in $\mathbb{W}$ as follows:

$$
\begin{aligned}
H(\vec{\rho}, \epsilon) &= G_\epsilon(\vec{\rho}), \\
H(\vec{\rho}, \mathsf{s}_i\sigma) &= \begin{cases} H(\vec{\rho}, \sigma) & \text{if } H(\vec{\rho}, \sigma)_0 \neq 1, \\ F(\vec{\rho}, \sigma) & \text{if } H(\vec{\rho}, \sigma)_0 = 1 \text{ and } F(\vec{\rho}, \sigma)_0 \neq 1, \\ G_i(\vec{\rho}, \sigma, H(\vec{\rho}, \sigma)_1) & \text{otherwise.} \end{cases}
\end{aligned}
$$

It is now a matter of routine to check (4) by (meta) notation induction on σ, using our assertions (1)–(3) from the induction hypothesis.

It still remains to check that the function H is indeed in FPTIME. Clearly, H is defined by recursion on notation from functions which are already known to be in FPTIME and, hence, it is sufficient to provide a suitable bound for H; of course it is enough to bound $H(\vec{\rho}, \sigma)$ under the assumption that $\vec{\rho}\ \mathbf{r}\ \Gamma[\vec{s}]$. Looking at our recursive definition of H, it is clear that H stays constant whenever we enter the first or the second case of our three-fold case distinction, so that bounding will be immediate from our discussion below. Further, when setting

$$H(\vec{\rho}, \mathsf{s}_i\sigma) = G_i(\vec{\rho}, \sigma, H(\vec{\rho}, \sigma)_1) \tag{5}$$

in the third case, we know that $H(\vec{\rho}, \sigma)_0 = 1$ *and* $F(\vec{\rho}, \sigma)_0 = 1$. Using (4) and (1) together with our assumption $\vec{\rho}\ \mathbf{r}\ \Gamma[\vec{s}]$ this means in particular that

$$H(\vec{\rho}, \sigma)_1\ \mathbf{r}\ A[\vec{s}, \sigma] \quad \text{and} \quad F(\vec{\rho}, \sigma)_1\ \mathbf{r}\ \mathsf{W}(t[\vec{s}](\sigma)). \tag{6}$$

But now we have to recall that the formula $A[\vec{s}, \sigma]$ has the shape

$$(\exists y \in \mathsf{W})[y \leq t[\vec{s}](\sigma)\ \wedge\ B[\vec{s}, y, \sigma]],$$

with B positive and W free; hence, the only occurrence of W in $A[\vec{s}, \sigma]$ stems from the leading bounded existential quantifier. But the bounding term $t[\vec{s}](\sigma)$ of this quantifier evaluates to $F(\vec{\rho}, \sigma)_1$ in $\mathcal{M}(\lambda\eta)$ according to (6). It is now easy to see that $H(\vec{\rho}, \sigma)_1$ is bounded by a *linear function* L in the length of $F(\vec{\rho}, \sigma)_1$; this only uses some obvious properties of our low level pairing function. It follows from these considerations that if we define $H(\vec{\rho}, \mathsf{s}_i\sigma)$ by (5) according to the third case in our case distinction, then it is clearly bounded. This ends our considerations concerning the bounding of the function H.

We have shown that the conclusion of the $\Sigma^{\mathsf{b}}_{\mathsf{W}}$ notation induction rule can be realized by a FPTIME function H. This ends our discussion of the induction rule and, in fact, also the proof of the realizability theorem for PT^+. □

The following corollary is immediate from our realizability theorem for PT^+ as well as the partial cut elimination theorem for PT^+ (Theorem 54). It shows that the provably total functions of PT^+ are contained in FPTIME.

Corollary 58 *Let t be a closed $\mathcal{L}_{\mathsf{W}}$ term and assume that*

$$\mathsf{PT}^+ \vdash \mathsf{W}(u_1), \ldots, \mathsf{W}(u_n) \Rightarrow \mathsf{W}(tu_1 \ldots u_n),$$

for distinct variables $u_1, \ldots, u_n$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in FPTIME *so that we have for all words $w_1, \ldots, w_n$ in $\mathbb{W}$,*

$$\mathcal{M}(\lambda\eta) \models t\overline{w}_1 \ldots \overline{w}_n = \overline{F(w_1, \ldots, w_n)}.$$

Proof. Assuming that we have a closed $\mathcal{L}_{\mathsf{W}}$ term t so that the sequent

$$\mathsf{W}(u_1), \ldots, \mathsf{W}(u_n) \Rightarrow \mathsf{W}(tu_1 \ldots u_n)$$

is provable in PT^+, we know that by partial cut elimination, this sequent has a proof using positive cut formulas only. Hence, our theorem provides a function G in FPTIME so that we have for all $\mathcal{L}_{\mathsf{W}}$ terms $s_1, \ldots, s_n$ and all words $\rho_1, \ldots, \rho_n$ in $\mathbb{W}$,

$$G(\rho_1, \ldots, \rho_n)_1 \ \mathbf{r}\ \mathsf{W}(ts_1 \ldots s_n),$$

whenever $\rho_i \,\mathbf{r}\, \mathsf{W}(s_i)$ for all $1 \leq i \leq n$. If we now set for given words $w_1, \ldots, w_n$ in $\mathbb{W}$,

$$s_i = \overline{w}_i, \ \rho_i = w_i, \ \text{ and } F(w_1, \ldots, w_n) = G(w_1, \ldots, w_n)_1,$$

then the assertion of our corollary is immediate. □

This ends our discussion of the realizability theorem for PT^+ and its crucial consequences. Turning to the realizability theorem for PTLS^+, note that the only difference between PT^+ and $\mathsf{PTLS}^+$is the presence of word multiplication $\times$ in PT^+. Hence, the proof of the following theorem is literally the same as the proof of the realizability theorem for PT^+, but since $\times$ does not need to be realized, the corresponding realizing function is indeed in FPTIMELINSPACE, according to the function algebra characterization of FPTIMELINSPACE given in Theorem 42. Again we can derive the desired corollary about the provably total functions of PTLS^+.

Theorem 59 (Realizability for PTLS^+) *Let $\Gamma \Rightarrow \Delta$ be a sequent of positive formulas with $\Gamma = A_1, \ldots, A_n$ and assume that $\mathsf{PTLS}^+ \mid\!\frac{\ }{\star}\, \Gamma[\vec{u}] \Rightarrow \Delta[\vec{u}]$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in* FPtimeLinspace *so that we have for all terms $\vec{s}$ and all $\rho_1, \ldots, \rho_n \in \mathbb{W}$:*

$$\textit{For all } 1 \le i \le n : \rho_i \ \mathbf{r}\ A_i[\vec{s}] \quad \Longrightarrow \quad F(\rho_1, \ldots, \rho_n)\ \mathbf{r}\ \Delta[\vec{s}].$$

Corollary 60 *Let t be a closed $\mathcal{L}_{\mathsf{W}}$ term and assume that*

$$\mathsf{PTLS}^+ \vdash \mathsf{W}(u_1), \ldots, \mathsf{W}(u_n) \Rightarrow \mathsf{W}(tu_1 \ldots u_n),$$

for distinct variables $u_1, \ldots, u_n$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in FPtimeLinspace *so that we have for all words $w_1, \ldots, w_n$ in $\mathbb{W}$,*

$$\mathcal{M}(\lambda\eta) \models t\overline{w}_1 \ldots \overline{w}_n = \overline{F(w_1, \ldots, w_n)}.$$

Let us now discuss the realizability theorems for the two systems PS^+ and LS^+. Indeed, also the proof of these theorems runs very analogous to the proof of the realizability theorem for PT^+. The crucial difference between PS^+ and PT^+ lies in the fact that PS^+ contains lexicographic induction on W, $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$, instead of the schema $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ of notation induction on W present in PT^+. The only difference in the realization of the corresponding rules of inference in the sequent-style setting is that one requires bounded lexicographic recursion (BRL) in order to realize the $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_\ell)$ rule, where, as we have seen above, bounded recursion on notation (BRN) was needed for the realization of the $(\Sigma^{\mathsf{b}}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ induction rule. Otherwise, the proof of the realizability theorem for PS^+ is identical to the one for PT^+. Hence, using the characterization of FPspace stated in Theorem 42, we are thus in a position to spell out the following theorem together with its expected corollary.

Theorem 61 (Realizability for PS^+) *Let $\Gamma \Rightarrow \Delta$ be a sequent of positive formulas with $\Gamma = A_1, \ldots, A_n$ and assume that $\mathsf{PS}^+ \mid\!\frac{\ }{\star}\, \Gamma[\vec{u}] \Rightarrow \Delta[\vec{u}]$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in* FPspace *so that we have for all terms $\vec{s}$ and all $\rho_1, \ldots, \rho_n \in \mathbb{W}$:*

$$\textit{For all } 1 \le i \le n : \rho_i \ \mathbf{r}\ A_i[\vec{s}] \quad \Longrightarrow \quad F(\rho_1, \ldots, \rho_n)\ \mathbf{r}\ \Delta[\vec{s}].$$

Corollary 62 *Let t be a closed $\mathcal{L}_{\mathsf{W}}$ term and assume that*

$$\mathsf{PS}^+ \vdash \mathsf{W}(u_1), \ldots, \mathsf{W}(u_n) \Rightarrow \mathsf{W}(tu_1 \ldots u_n),$$

for distinct variables $u_1, \ldots, u_n$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in FPSPACE *so that we have for all words $w_1, \ldots, w_n$ in $\mathbb{W}$,*

$$\mathcal{M}(\lambda\eta) \models t\overline{w}_1 \ldots \overline{w}_n = \overline{F(w_1, \ldots, w_n)}.$$

As above, if word multiplication $\times$ is absent, then the proof for PS^+ actually produces realizing functions in FLINSPACE. Thus we obtain the following realizability theorem for the system LS^+.

Theorem 63 (Realizability for LS^+) *Let $\Gamma \Rightarrow \Delta$ be a sequent of positive formulas with $\Gamma = A_1, \ldots, A_n$ and assume that $\mathsf{LS}^+ \underset{\star}{\vdash} \Gamma[\vec{u}] \Rightarrow \Delta[\vec{u}]$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in* FLINSPACE *so that we have for all terms $\vec{s}$ and all $\rho_1, \ldots, \rho_n \in \mathbb{W}$:*

$$\text{For all } 1 \leq i \leq n : \rho_i \ \mathbf{r}\ A_i[\vec{s}] \quad \Longrightarrow \quad F(\rho_1, \ldots, \rho_n) \ \mathbf{r}\ \Delta[\vec{s}].$$

Corollary 64 *Let t be a closed $\mathcal{L}_{\mathsf{W}}$ term and assume that*

$$\mathsf{LS}^+ \vdash \mathsf{W}(u_1), \ldots, \mathsf{W}(u_n) \Rightarrow \mathsf{W}(tu_1 \ldots u_n),$$

for distinct variables $u_1, \ldots, u_n$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in FLINSPACE *so that we have for all words $w_1, \ldots, w_n$ in $\mathbb{W}$,*

$$\mathcal{M}(\lambda\eta) \models t\overline{w}_1 \ldots \overline{w}_n = \overline{F(w_1, \ldots, w_n)}.$$

10.4 Putting the pieces together

The results of the previous section, namely Corollary 58, Corollary 60, Corollary 62, and Corollary 64, can now be combined with our lower bound results summarized in Theorem 51. Hence, we have now established the following main theorem concerning the provably total functions of the four systems PT, PTLS, LS, and PS.

Theorem 65 *We have the following proof-theoretic results:*

1. *The provably total functions of PT coincide with* FPTIME.
2. *The provably total functions of PTLS coincide with* FPTIMELINSPACE.
3. *The provably total functions of PS coincide with* FPSPACE.
4. *The provably total functions of LS coincide with* FLINSPACE.

Moreover, this theorem holds true in the presence of totality of application (Tot) *and extensionality of operations* (Ext).

Chapter 11

Further applicative systems

It is the aim of this chapter to consider further natural applicative systems for various classes of computable functions. We start with the system PH which is closely related to the polynomial time hierarchy PH. The second section is concerned with applicative systems for the primitive recursive functions and, finally, in the last section we make some remarks concerning an applicative setting which is of the same strength as Peano arithmetic PA.

In the course of this chapter we will see that the techniques developed in this part of our thesis so far extend in a straightforward manner to various systems considered in the following sections.

11.1 A type two functional for bounded quantification

In this section we consider a natural type two functional π which allows for the elimination of bounded quantifiers. Using the techniques of the previous chapter we will show that the provably total functions of the theory PT augmented by π are exactly the functions on $\mathbb{W}$ in the function polynomial time hierarchy FPH.

It is worth mentioning at this point that the formulation and spirit of the π functional is similar to the non-constructive μ operator which has been studied extensively in the applicative context, cf. the papers Feferman and Jäger [35, 36], Glass and Strahm [43], Jäger and Strahm [68], Marzetta and Strahm [84], and Strahm [123]. In contrast to π, the operator μ tests for *unbounded* quantification and, hence, is much stronger than the π functional. The applicative axiomatization of the two functionals, however, is completely analogous.

As usual, a function F on the binary words $\mathbb{W}$ is defined to be in the *(function) polynomial time hierarchy* FPH if F is computable in polynomial time using finitely many oracles from the Meyer-Stockmeyer polynomial time hierarchy PH on $\mathbb{W}$. It is well-known how to extend Cobham's function algebra characterization of FPTIME so as to capture FPH: one simply closes the Cobham algebra under bounded quantification. In the sequel we let (BQ) denote the operator which maps an $(n{+}1)$-ary function F on $\mathbb{W}$ to the $(n{+}1)$-ary function $\mathsf{BQ}(F)$, which is given for all $\vec{x}, y \in \mathbb{W}$ as follows:

$$\mathsf{BQ}(F)(\vec{x}, y) \;:=\; \begin{cases} 0 & \text{if } (\exists z \leq y) F(\vec{x}, z) = 0, \\ 1 & \text{otherwise.} \end{cases}$$

The following theorem is folklore, cf. Clote's survey article [22] on function algebras and computation models.

Theorem 66 *We have the following function algebra characterization:*

$$[\epsilon, \mathsf{I}, \mathsf{s}_0, \mathsf{s}_1, *, \times; \mathsf{COMP}, \mathsf{BRN}, \mathsf{BQ}] = \mathrm{FP}\textsc{h}.$$

For the formulation of our type two functional for bounded quantification in the applicative setting, we assume that the applicative language $\mathcal{L}_\mathsf{W}$ is extended by a new constant π. The axioms for π are divided into $(\pi.1)$ and $(\pi.2)$: the first axiom claims that for a given total operation f on W and an $a \in \mathsf{W}$, it is always the case that $\pi f a$ is a word whose length is bounded by the length of a; the second axioms expresses, in addition, that $\pi f a$ is a zero of f provided that there exists a word $x \leq a$ with $fx = 0$. Hence, given that $f : \mathsf{W} \to \mathsf{W}$ and $a \in \mathsf{W}$, we have that indeed $(\exists x \leq a)(fx = 0)$ *is equivalent to* $f(\pi f a) = 0$, i.e., bounded quantifiers can be eliminated by means of π.

The type two functional π for bounded quantification

$(\pi.1)$ $\quad f : \mathsf{W} \to \mathsf{W} \wedge a \in \mathsf{W} \;\to\; \pi f a \in \mathsf{W} \wedge \pi f a \leq a$

$(\pi.2)$ $\quad f : \mathsf{W} \to \mathsf{W} \wedge a \in \mathsf{W} \wedge (\exists x \leq a)(fx = 0) \;\to\; f(\pi f a) = 0$

We now define the $\mathcal{L}_\mathsf{W}$ theory PH to be simply PT plus the two axioms $(\pi.1)$ and $(\pi.2)$. We aim at showing that the provably total functions of PH are exactly the functions in the function polynomial time hierarchy FPH.

Clearly, we can make use of the function algebra characterization of FPH given in the theorem above in order to show that the provably total functions of PH contain FPH: with the help of π we have closure under bounded quantification and, moreover, due to Lemma 46 we know that in PH closure under bounded recursion on notation is available. Hence, we can state the following theorem.

Theorem 67 *The provably total functions of* PH *include* FPH*.*

Indeed, let us mention that it is possible to show that Ferreira's system Σ^b_∞-NIA (cf. Ferreira [39]) or, equivalently, Buss' system S_2 (cf. Buss [15]) are directly contained in PH.

In order to show that the lower bound stated in the above theorem is sharp, we can make use in a straightforward manner of the partial cut elimination and realizability techniques introduced in the previous chapter. In the following we sketch the main new steps of this procedure.

As above, we provide an upper bound directly for the system PH^+, i.e., the extension of PH by totality and extensionality. The Gentzen-style reformulation of PH^+ simply extends the Gentzen-style version of PT^+ by two new rules corresponding to the axioms (π.1) and (π.2) for π. As expected, in these rules u denotes a fresh variable.

$$\frac{\Gamma,\, \mathsf{W}(u) \;\Rightarrow\; \mathsf{W}(tu),\, \Delta}{\Gamma,\, \mathsf{W}(s) \;\Rightarrow\; \mathsf{W}(\pi ts) \wedge \pi ts \leq s,\, \Delta}\ (\pi.1)$$

$$\frac{\Gamma,\, \mathsf{W}(u) \;\Rightarrow\; \mathsf{W}(tu),\, \Delta}{\Gamma,\, \mathsf{W}(s),\, (\exists x \leq s)(tx = 0) \;\Rightarrow\; t(\pi ts) = 0,\, \Delta}\ (\pi.2)$$

We observe that the main formulas of both rules are *positive*, so that the partial cut elimination theorem for PT^+ (Theorem 54) readily extends to PH^+. Hence, we can assume that PH^+ derivations of sequents of positive formulas contain cuts with positive cut formulas only.

In the sequel we want to use the same notion of realizability as in the previous chapter. Hence, we have to extend our open term model $\mathcal{M}(\lambda\eta)$ so as to incorporate the new constant π. The informal interpretation of πfa is simply the least $x \leq a$ so that $fx = 0$, if such an x exists, and ϵ otherwise.[1] Formally

[1] Leastness is always understood in the sense of the lexicographic ordering of the full binary tree. In the sequel we use the notation $(\mu x \leq a)R(x)$ to denote the least $x \leq a$ satisfying $R(x)$ if it exists, and ϵ otherwise.

in $\mathcal{M}(\lambda\eta)$, we can either write down appropriate reduction rules for π or use recursion in $\mathcal{M}(\lambda\eta)$ in order to define π directly. The realizability theorem for PH^+ is now spelled out in the expected manner.

Theorem 68 (Realizability for PH^+) *Let $\Gamma \Rightarrow \Delta$ be a sequent of positive formulas with $\Gamma = A_1, \dots, A_n$ and assume that $\mathsf{PH}^+ \vdash_{\star} \Gamma[\vec{u}] \Rightarrow \Delta[\vec{u}]$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in* FPH *so that we have for all terms $\vec{s}$ and all $\rho_1, \dots, \rho_n \in \mathbb{W}$:*

$$\textit{For all } 1 \leq i \leq n : \rho_i \ \mathbf{r} \ A_i[\vec{s}] \quad \Longrightarrow \quad F(\rho_1, \dots, \rho_n) \ \mathbf{r} \ \Delta[\vec{s}].$$

Proof. In addition to the proof of the realizability theorem for PT^+ we only have to show how to deal with the two rules $(\pi.1)$ and $(\pi.2)$. For that purpose let us assume that we have a quasi cut free derivation of the sequent

$$\Gamma, \mathsf{W}(u) \Rightarrow \mathsf{W}(tu), \Delta,$$

and let F_0 denote the function in FPH which is given to us by the induction hypothesis. In case of $(\pi.1)$ it is not difficult to check that the following function F can be used as a realizing function for the conclusion of this rule.

$$F(\vec{\rho}, \sigma) \ = \ \begin{cases} \langle 1, \langle (\mu\tau \leq \sigma . F_0(\vec{\rho}, \tau)_1 = 0), \epsilon \rangle \rangle & \text{if } (\forall \tau \leq \sigma) F_0(\vec{\rho}, \tau)_0 = 1, \\ F_0(\vec{\rho}, (\mu\tau \leq \sigma) F_0(\vec{\rho}, \tau)_0 \neq 1) & \text{otherwise} \end{cases}$$

It is easy to see that F is in FPH, since the functions in the polynomial time hierarchy are clearly closed under bounded minimization. In the case of the rule for $(\pi.2)$ the realizing function F for its conclusion can be chosen as follows. Again it is easy to see that this F is in FPH.

$$F(\vec{\rho}, \sigma, \sigma') \ = \ \begin{cases} \langle 1, \epsilon \rangle & \text{if } (\forall \tau \leq \sigma) F_0(\vec{\rho}, \tau)_0 = 1, \\ F_0(\vec{\rho}, (\mu\tau \leq \sigma) F_0(\vec{\rho}, \tau)_0 \neq 1) & \text{otherwise} \end{cases}$$

This ends our short discussion of the proof of the realizability theorem for the system PH^+. □

As above, we can now derive the following crucial corollary.

Corollary 69 *Let t be a closed $\mathcal{L}_{\mathsf{W}}$ term and assume that*

$$\mathsf{PH}^+ \vdash \mathsf{W}(u_1), \dots, \mathsf{W}(u_n) \Rightarrow \mathsf{W}(tu_1 \dots u_n),$$

for distinct variables $u_1, \dots, u_n$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in FPH *so that we have for all words $w_1, \dots, w_n$ in $\mathbb{W}$,*

$$\mathcal{M}(\lambda\eta) \models t\overline{w}_1 \dots \overline{w}_n = \overline{F(w_1, \dots, w_n)}.$$

This last corollary combined with Theorem 67 yields the following main result of this section.

Theorem 70 *The provably total functions of* PH *coincide with* $\mathrm{FP{\small H}}$. *In addition, this theorem holds true in the presence of totality of application* (Tot) *and extensionality of operations* (Ext).

11.2 Positive induction equals primitive recursion

In this section we briefly examine the effect of replacing our bounded induction principles ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\mathsf{W}}$) and ($\Sigma^{\mathsf{b}}_{\mathsf{W}}$-$\mathsf{I}_{\ell}$) by the schema of induction for arbitrary *positive* formulas. We will show that the corresponding applicative framework characterizes exactly the class of primitive recursive functions. This result is previously due to Cantini [18][2]. However, the proof given here is new and quite different from the techniques used by Cantini.

The primitive recursive functions $\mathrm{FP{\small RIM}}$ on $\mathbb{W}$ are generated from the usual initial functions by closing under composition and recursion on notation (RN), where (RN) is simply (BRN) without the bounding condition. Hence, using our function algebra notation, $\mathrm{FP{\small RIM}}$ is defined to be the function algebra $[\epsilon, \mathsf{I}, \mathsf{s}_0, \mathsf{s}_1; \mathsf{COMP}, \mathsf{RN}]$. Denoting by ($\mathsf{RL}$) the corresponding schema of unbounded lexicographic recursion, it is well known that indeed

$$[\epsilon, \mathsf{I}, \mathsf{s}_\ell; \mathsf{COMP}, \mathsf{RL}] = [\epsilon, \mathsf{I}, \mathsf{s}_0, \mathsf{s}_1; \mathsf{COMP}, \mathsf{RN}].$$

Hence, it does not matter whether we use lexicographic or notation recursion in the context of unbounded recursion schemas.

Let us now turn to a natural applicative framework PR capturing $\mathrm{FP{\small RIM}}$. The schema of positive notation induction on W, (Pos-I_{W}), includes for each formula $A(x)$ in the class Pos,

$$(\mathsf{Pos}\text{-}\mathsf{I}_{\mathsf{W}}) \qquad A(\epsilon) \wedge (\forall x \in \mathsf{W})(A(x) \rightarrow A(\mathsf{s}_0 x) \wedge A(\mathsf{s}_1 x)) \;\rightarrow\; (\forall x \in \mathsf{W}) A(x)$$

The schema of positive lexicographic induction on W, (Pos-I_ℓ), is stated accordingly. The applicative theory PR is now defined to be the theory B plus

[2]Actually, Cantini establishes a slightly stronger theorem in the sense that he also allows *negative equations* to occur in induction formulas.

positive notation induction on W, $(\mathsf{Pos\text{-}I_W})$. Observe that we do not include $*$ and $\times$ in PR as these are easily definable as we will see now.

As can be expected, it is possible represent recursion on notation in PR in a very direct and natural way, by referring to the recursion theorem of B and exploiting $(\mathsf{Pos\text{-}I_W})$. In particular, we obtain in a straightforward manner the following unbounded analogue of Lemma 46; it's proof is an obvious adaptation of the proof of Lemma 46 and, therefore, is left to the reader.

Lemma 71 *There exists a closed $\mathcal{L}_\mathsf{W}$ term $\tilde{\mathsf{r}}_\mathsf{W}$ so that* PR *proves*

$$f : \mathsf{W} \to \mathsf{W} \wedge g : \mathsf{W}^3 \to \mathsf{W} \quad \to$$

$$\begin{cases} \tilde{\mathsf{r}}_\mathsf{W} fg : \mathsf{W}^2 \to \mathsf{W} \wedge \\ x \in \mathsf{W} \wedge y \in \mathsf{W} \wedge y \neq \epsilon \wedge h = \tilde{\mathsf{r}}_\mathsf{W} fg \to \\ \qquad hx\epsilon = fx \wedge hxy = gxy(hx(\mathsf{p_W} y)) \end{cases}$$

Corollary 72 *The provably total functions of* PR *include* FPRIM.

Indeed, PR does not only establish the convergence of each primitive recursive function, but it also interprets in a straightforward manner the subsystem of Peano arithmetic PA which is based on the schema of complete induction for Σ_1 formulas. The latter system is well-known to be a conservative extension with respect to Π_2 statements of primitive recursive arithmetic PRA as was shown by Parsons [90].

Before we turn to the upper bound of PR let us quickly address the question of whether it matters if we include $(\mathsf{Pos\text{-}I_W})$ or $(\mathsf{Pos\text{-}I_\ell})$ in our definition of PR. As our discussion above concerning the corresponding function algebras suggests, there should be no difference, and indeed this is confirmed by the following lemma.

Lemma 73 *We have that* $(\mathsf{Pos\text{-}I_\ell})$ *and* $(\mathsf{Pos\text{-}I_W})$ *are equivalent over our base theory* B.

Proof. Let us briefly sketch this equivalence. Firstly, the fact that $(\mathsf{Pos\text{-}I_\ell})$ entails $(\mathsf{Pos\text{-}I_W})$ over B is shown by literally the same proof as in Lemma 48. For the reverse direction we work informally in B plus $(\mathsf{Pos\text{-}I_W})$, aiming at deriving each instance of $(\mathsf{Pos\text{-}I_\ell})$. Let $A(x)$ be a positive formula and assume

$$A(\epsilon) \wedge (\forall x \in \mathsf{W})(A(x) \to A(\mathsf{s}_\ell x)). \tag{1}$$

Now we first note that the length function $|\cdot|$ used in the proof of Lemma 48 is easily definable in $\mathsf{B} + (\mathsf{Pos}\text{-}\mathsf{I}_{\mathsf{W}})$, due to our previous lemma. Now we define the positive formula $B(x)$ to be simply $A(|x|)$ and readily observe that (1) entails

$$(2) \qquad B(\epsilon) \wedge (\forall x \in \mathsf{W})(B(x) \to B(\mathsf{s}_0 x) \wedge B(\mathsf{s}_1 x)).$$

From (2) and $(\mathsf{Pos}\text{-}\mathsf{I}_{\mathsf{W}})$ we have thus shown $(\forall x \in \mathsf{W})A(|x|)$. The final step consists now in finding a term $\mathsf{exp} : \mathsf{W} \to \mathsf{W}$, so that the theory $\mathsf{B} + (\mathsf{Pos}\text{-}\mathsf{I}_{\mathsf{W}})$ proves $(\forall x \in \mathsf{W})(|\mathsf{exp}x| = x)$. The definition of exp is straightforward by the previous lemma and we leave the details to the reader. Using exp we are now able to show that

$$(3) \qquad (\forall x \in \mathsf{W})A(|x|) \to (\forall x \in \mathsf{W})A(x),$$

and this suddenly completes the proof of the fact that $(\mathsf{Pos}\text{-}\mathsf{I}_{\mathsf{W}})$ entails $(\mathsf{Pos}\text{-}\mathsf{I}_{\ell})$ over the base theory B. $\square$

Clearly, this last lemma shows that in the theory PR we have available the lexicographic analogue of Lemma 71.

The final part of this section is devoted to showing that the provably total functions of PR do not go beyond the primitive recursive functions $\mathrm{FP}\textsc{rim}$ on $\mathbb{W}$. Again our realizability techniques work in a perspicuous manner. We first reformulate the system PR^+, i.e., $\mathsf{PR} + (\mathsf{Tot}) + (\mathsf{Ext})$, in sequent style. Positive induction on notation $(\mathsf{Pos}\text{-}\mathsf{I}_{\mathsf{W}})$ is stated as a rule in the same way as for the system PT^+, but of course without the premise concerning the totality of a bounding function. Partial cut elimination for PR^+ works as before. As to the realizability theorem, its proof is literally the same as the proof of the realizability theorem for PT^+, with the only difference that in the treatment of the notation induction rule, we have no bounding information available and, hence, we can only conclude that the corresponding function is primitive recursive.

Theorem 74 (Realizability for PR^+) *Let $\Gamma \Rightarrow \Delta$ be a sequent of positive formulas with $\Gamma = A_1, \ldots, A_n$ and assume that $\mathsf{PR}^+ \underset{\star}{\vdash} \Gamma[\vec{u}] \Rightarrow \Delta[\vec{u}]$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in $\mathrm{FP}\textsc{rim}$ so that we have for all terms $\vec{s}$ and all $\rho_1, \ldots, \rho_n \in \mathbb{W}$:*

$$\textit{For all } 1 \leq i \leq n : \rho_i \ \mathbf{r} \ A_i[\vec{s}] \quad \Longrightarrow \quad F(\rho_1, \ldots, \rho_n) \ \mathbf{r} \ \Delta[\vec{s}].$$

Corollary 75 *Let t be a closed $\mathcal{L}_{\mathsf{W}}$ term and assume that*

$$\mathsf{PR}^+ \vdash \mathsf{W}(u_1), \dots, \mathsf{W}(u_n) \Rightarrow \mathsf{W}(tu_1 \dots u_n),$$

for distinct variables $u_1, \dots, u_n$. Then there exists a function $F : \mathbb{W}^n \to \mathbb{W}$ in FPRIM *so that we have for all words $w_1, \dots, w_n$ in $\mathbb{W}$,*

$$\mathcal{M}(\lambda\eta) \models t\overline{w}_1 \dots \overline{w}_n = \overline{F(w_1, \dots, w_n)}.$$

From this corollary and Corollary 72 we are now in a position to state the following crucial theorem concerning the provably total functions of PR. As we have noted above, (a slight strengthening of) this theorem has previously been obtained by Cantini [18], using a quite different argument.

Theorem 76 *The provably total functions of* PR *coincide with* FPRIM. *In addition, this theorem holds true in the presence of totality of application* (Tot) *and extensionality of operations* (Ext).

11.3 Full induction and Peano arithmetic

A further natural strengthening of our applicative framework consists in allowing induction on W for arbitrary formulas in the language $\mathcal{L}_{\mathsf{W}}$. Using known techniques, it easily follows that the so-obtained applicative systems have the same proof-theoretic strength as Peano arithmetic PA.

By ($\mathcal{L}_{\mathsf{W}}$-I_{W}) and ($\mathcal{L}_{\mathsf{W}}$-I_{ℓ}) we denote the schema of notation induction and lexicographic induction on W, respectively, for arbitrary formulas of our applicative language $\mathcal{L}_{\mathsf{W}}$. With the same argument as in Lemma 73 above one establishes that ($\mathcal{L}_{\mathsf{W}}$-I_{W}) and ($\mathcal{L}_{\mathsf{W}}$-I_{ℓ}) are equivalent over the base theory B. For an interpretation of $\mathsf{B} + (\mathcal{L}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ or $\mathsf{B} + (\mathcal{L}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell})$ in Peano arithmetic PA, one makes use of an inner model construction, formalizing the standard recursion-theoretic model *PRO* of B, cf. e.g. Feferman and Jäger [35] for a similar argument. The so-obtained interpretation yields that the provably total functions of $\mathsf{B} + (\mathcal{L}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ and $\mathsf{B} + (\mathcal{L}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell})$ are exactly the α recursive functions for α less than PA's proof-theoretic ordinal ε_0.

The interpretation of $\mathsf{B} + (\mathcal{L}_{\mathsf{W}}\text{-}\mathsf{I}_{\mathsf{W}})$ or $\mathsf{B} + (\mathcal{L}_{\mathsf{W}}\text{-}\mathsf{I}_{\ell})$ can also be strengthened so as to include the axiom of totality (Tot) and the axiom of extensionality (Ext). In this case, one simply formalizes the standard term model $\mathcal{M}(\lambda\eta)$

of $\mathsf{B} + (\mathsf{Tot}) + (\mathsf{Ext})$ in PA, cf. Cantini [17] or Jäger and Strahm [68] for more details.

Let us conclude this section by noting that similar inner model constructions are of no use in order to establish upper bounds e.g. for the system PR: the reason is that in induction formulas in PR arbitrary unbounded universal quantifiers over individuals are allowed, which makes an embedding in, say, primitive recursive arithmetic PRA extended by Σ_1 induction impossible.

Conclusion of Part II

In this part of our habilitation thesis we have presented a series of natural applicative systems of various bounded complexities. In particular, we have elucidated frameworks for the functions on binary words computable in polynomial time, polynomial time *and* linear space, polynomial space, linear space, as well as the polynomial time hierarchy. Our systems can be viewed as natural applicative analogues of various bounded arithmetics; this is witnessed by the fact that the latter are directly embeddable into various applicative settings. A further distinguished feature of applicative theories is that they allow for a very direct treatment of higher types issues: we have seen that even higher order systems such as Cook and Urquhart's PV^ω are directly contained in the applicative theory PT for the polynomial time computable functions.

Apart from the world of bounded recursion schemas, bounded arithmetic and bounded applicative theories there is the world of so-called *tiered systems* in the sense of Cook and Bellantoni (cf. e.g. [10]) and Leivant (cf. e.g. [77, 79]). Crucial for this approach to characterizing complexities is a strictly predicative regime which distinguishes between different uses of variables in induction and recursion schemas, thus severely restricting the definable or provably total functions in various unbounded formalisms. In our applicative setting such a "predicativization" amounts to distinguishing between (at least) two sorts or types of binary words W_0 and W_1, say, where induction over W_1 is allowed for formulas which are positive and do not contain W_1, cf. Cantini [19, 21] for such systems.

Unarguably, the tiered approach to complexity has led to numerous highly interesting and intrinsic recursion-theoretic and also proof-theoretic characterizations of complexity classes, which might lead to new subrecursive programming paradigms. Also, higher type issues have recently been a subject of interest in this area, cf. e.g. Leivant [78], Bellantoni, Niggl, Schwichtenberg

[11], and Hofmann [50]. In spite of its elegance, it has to be mentioned that the tiered or ramified approach also has its drawbacks. First of all, there is the general observation that reasoning in a system with ramifications can be very difficult: for example, dealing with two tiers W_0 and W_1 only, one has to take into account four kinds of functions from binary words to binary words, which are not closed under composition, of course. Secondly, the strict predicative regime disallows the direct formulation of many natural algorithms, especially those obtained by various kinds of nested recursions, cf. Hofmann [49] for a discussion. And thirdly, it is not at all clear how modern tiered systems relate to the more traditional bounded subsystems of first and higher order arithmetic.

Taking up these points of criticism in the context of the bounded world, of course one has to pay a price in order to avoid ramifications and to deal only with one type W of binary words. Namely, the systems discussed in this thesis include initial functions such as word concatenation and word multiplication as well as recursions and inductions need to be bounded. On the other hand, nesting recursions is generally easy and in many cases it is also not difficult to provide the necessary bounding information. Hence, both the bounded and the tiered approach have their pros and cons. Summing up, in our opinion it is worth exploring the bounded *and* the ramified world, and it would be especially interesting to find out more about the exact relationship between these two worlds.

Coming back to the work and results achieved in this thesis, let us briefly address some directions for future research. Certainly, there is the need to further study and elucidate the role of higher type functionals in the various settings that we have been considering in this paper. Recently, we have done a first step in this direction and shown that indeed

the provably total type two functionals of PT *coincide with the basic feasible functionals of type two*,

and we conjecture that this result holds at all higher types. The proof that a provably total type two functional of PT is basic feasible is simply a refinement of the realizability theorem for PT established above. Details will be given in a publication under preparation.

Finally, a further important research project consists in considering extensions of the applicative systems of this thesis by adding suitable versions of flexible typing and naming in the spirit of explicit mathematics in order to answer the question of what type existence principles can live in a, say, feasible setting of explicit mathematics. We believe that the formalisms designed in this thesis should help in finding suitable versions of "bounded explicit mathematics".

Bibliography

[1] Aczel, P. The strength of Martin-Löf's type theory with one universe. Tech. rep., Dept. of Philosophy, University of Helsinki, 1977.

[2] Arai, T. Some results on cut-elimination, provable well-orderings, induction and reflection. *Annals of Pure and Applied Logic 95* (1998), 93–184.

[3] Arai, T. An introduction to the finitary analyses of proof figures. In *Sets and Proofs*, S. B. Cooper and J. Truss, Eds. Cambridge University Press, 1999, pp. 1–25.

[4] Avigad, J. On the relationship between ATR_0 and $\widehat{ID}_{<\omega}$. *Journal of Symbolic Logic 61*, 3 (1996), 768–779.

[5] Balcázar, J. L., Díaz, J., and Gabarró, J. *Structural Complexity I.* EATCS Monographs on Theoretical Computer Science. Springer Verlag, Berlin, 1988.

[6] Barendregt, H. P. *The Lambda Calculus*, revised ed. North Holland, Amsterdam, 1984.

[7] Beckmann, A. *Separating fragments of bounded arithmetic.* PhD thesis, Universität Münster, 1996.

[8] Beeson, M. J. *Foundations of Constructive Mathematics: Metamathematical Studies.* Springer, Berlin, 1985.

[9] Beeson, M. J. Proving programs and programming proofs. In *Logic, Methodology and Philosophy of Science VII*, Barcan Marcus et. al., Ed. North Holland, Amsterdam, 1986, pp. 51–82.

[10] Bellantoni, S., and Cook, S. A new recursion-theoretic characterization of the poly-time functions. *Computational Complexity 2* (1992), 97–110.

[11] Bellantoni, S., Niggl, K.-H., and Schwichtenberg, H. Higher type recursion, ramification and polynomial time. *Annals of Pure and Applied Logic 104*, 1–3 (2000), 17–30.

[12] Bovet, D., and Crescenzi, P. *Introduction to the Theory of Complexity.* Prentice Hall, 1994.

[13] Buchholz, W. An intuitionistic fixed point theory. *Archive for Mathematical Logic 37* (1997), 21–27.

[14] Buchholz, W., Feferman, S., Pohlers, W., and Sieg, W. *Iterated Inductive Definitions and Subsystems of Analysis: Recent Proof-Theoretical Studies*, vol. 897 of *Lecture Notes in Mathematics.* Springer, Berlin, 1981.

[15] Buss, S. R. *Bounded Arithmetic.* Bibliopolis, Napoli, 1986.

[16] Cantini, A. On the computational content of theories of operations with total application, June 1995. Handwritten notes.

[17] Cantini, A. *Logical Frameworks for Truth and Abstraction.* North-Holland, Amsterdam, 1996.

[18] Cantini, A. Proof-theoretic aspects of self-referential truth. In *Tenth International Congress of Logic, Methodology and Philosophy of Science, Florence, August 1995*, Maria Luisa Dalla Chiara et. al., Ed., vol. 1. Kluwer, September 1997.

[19] Cantini, A. Characterizing poly-time with an intuitionistic theory based on combinatory logic and safe induction. Preprint, Firenze, 1999. 14 pages.

[20] Cantini, A. Feasible operations and applicative theories based on $\lambda\eta$. *Mathematical Logic Quarterly 46*, 3 (2000), 291–312.

[21] Cantini, A. Polytime, combinatory logic and positive safe induction. Preprint, Firenze, 2000. 27 pages.

[22] Clote, P. Computation models and function algebras. In *Handbook of Computability Theory*, E. Griffor, Ed. Elsevier, 1999, pp. 589–681.

[23] Cobham, A. The intrinsic computational difficulty of functions. In *Logic, Methodology and Philosophy of Science II*. North Holland, Amsterdam, 1965, pp. 24–30.

[24] Cook, S. The complexity of theorem proving procedures. In *3rd Annual ACM Symposium on Theory of Computing*. 1971.

[25] Cook, S. A., and Kapron, B. M. Characterizations of the basic feasible functionals of finite type. In *Feasible Mathematics*, S. R. Buss and P. J. Scott, Eds. Birkhäuser, Basel, 1990, pp. 71–95.

[26] Cook, S. A., and Urquhart, A. Functional interpretations of feasibly constructive arithmetic. *Annals of Pure and Applied Logic 63*, 2 (1993), 103–200.

[27] Feferman, S. A language and axioms for explicit mathematics. In *Algebra and Logic*, J. Crossley, Ed., vol. 450 of *Lecture Notes in Mathematics*. Springer, Berlin, 1975, pp. 87–139.

[28] Feferman, S. Recursion theory and set theory: a marriage of convenience. In *Generalized recursion theory II, Oslo 1977*, J. E. Fenstad, R. O. Gandy, and G. E. Sacks, Eds., vol. 94 of *Stud. Logic Found. Math.* North Holland, Amsterdam, 1978, pp. 55–98.

[29] Feferman, S. Constructive theories of functions and classes. In *Logic Colloquium '78*, M. Boffa, D. van Dalen, and K. McAloon, Eds. North Holland, Amsterdam, 1979, pp. 159–224.

[30] Feferman, S. Iterated inductive fixed-point theories: application to Hancock's conjecture. In *The Patras Symposion*, G. Metakides, Ed. North Holland, Amsterdam, 1982, pp. 171–196.

[31] Feferman, S. Polymorphic typed lambda-calculi in a type-free axiomatic framework. In *Logic and Computation*, W. Sieg, Ed., vol. 106 of *Contemporary Mathematics*. American Mathematical Society, Providence, Rhode Island, 1990, pp. 101–136.

[32] Feferman, S. Logics for termination and correctness of functional programs. In *Logic from Computer Science*, Y. N. Moschovakis, Ed., vol. 21 of *MSRI Publications*. Springer, Berlin, 1991, pp. 95–127.

[33] Feferman, S. Reflecting on incompleteness. *Journal of Symbolic Logic 56*, 1 (1991), 1–49.

[34] Feferman, S. Logics for termination and correctness of functional programs II: Logics of strength PRA. In *Proof Theory*, P. Aczel, H. Simmons, and S. S. Wainer, Eds. Cambridge University Press, Cambridge, 1992, pp. 195–225.

[35] Feferman, S., and Jäger, G. Systems of explicit mathematics with non-constructive μ-operator. Part I. *Annals of Pure and Applied Logic 65*, 3 (1993), 243–263.

[36] Feferman, S., and Jäger, G. Systems of explicit mathematics with non-constructive μ-operator. Part II. *Annals of Pure and Applied Logic 79*, 1 (1996), 37–52.

[37] Ferreira, F. *Polynomial Time Computable Arithmetic and Conservative Extensions.* PhD thesis, Pennsylvania State University, 1988.

[38] Ferreira, F. Polynomial time computable arithmetic. *Contemporary Mathematics 106* (1990), 137–156.

[39] Ferreira, F. Stockmeyer induction. In *Feasible Mathematics*, S. Buss and P. Scott, Eds. Birkhäuser, 1990, pp. 161–180.

[40] Friedman, H., McAloon, K., and Simpson, S. A finite combinatorial principle which is equivalent to the 1-consistency of predicative analysis. In *Patras Symposion*, G. Metakides, Ed. North Holland, Amsterdam, 1982, pp. 197–230.

[41] Gentzen, G. Neue Fassung des Widerspruchsfreiheitsbeweises für die reine Zahlentheorie. *Forschungen zur Logik und zur Grundlegung der exakten Wissenschaften, Neue Folge 4* (1938), 19–44.

[42] Girard, J.-Y. *Proof Theory and Logical Complexitiy.* Bibliopolis, Napoli, 1987.

[43] Glass, T., and Strahm, T. Systems of explicit mathematics with non-constructive μ-operator and join. *Annals of Pure and Applied Logic 82* (1996), 193–219.

[44] Gödel, K. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme. *Monatshefte für Mathematik und Physik 38* (1931), 173–198.

[45] Griffor, E., and Rathjen, M. The strength of some Martin-Löf type theories. *Archive for Mathematical Logic 33* (1994), 347–385.

[46] Hájek, P., and Pudlák, P. *Metamathematics of First-Order Arithmetic*. Perspectives in Mathematical Logic. Springer, 1993.

[47] Hayashi, S., and Nakano, H. *PX: A Computational Logic*. MIT Press, Cambridge, MA, 1988.

[48] Hindley, J. R., and Seldin, J. P. *Introduction to Combinators and λ-Calculus*. Cambridge University Press, 1986.

[49] Hofmann, M. Linear types and non-size-increasing polynomial time computation. In *LICS '99, Trento* (1999), IEEE.

[50] Hofmann, M. Type systems for polynomial-time computation. Habilitation Thesis, Darmstadt, 1999. Appeared as LFCS Technical Report ECS-LFCS-99-406.

[51] Hopcroft, J., and Ullman, J. *Introduction to Automata Theory, Languages, and Computation*. Addision-Wesley, 1979.

[52] Irwin, R., Kapron, B., and Royer, J. On characterizations of the basic feasible functionals, Part I. *Journal of Functional Programming*. to appear.

[53] Jäger, G. Metapredicative and explicit Mahlo: a proof-theoretic perspective. To appear in *Proceedings Logic Colloquium 2000*.

[54] Jäger, G. Beweistheorie von *KPN*. *Archiv für mathematische Logik und Grundlagenforschung 20* (1980), 53–64.

[55] Jäger, G. Theories for iterated jumps, 1980. Handwritten notes.

[56] Jäger, G. A well-ordering proof for Feferman's theory T_0. *Archiv für mathematische Logik und Grundlagenforschung 23* (1983), 65–77.

[57] Jäger, G. The strength of admissibility without foundation. *The Journal of Symbolic Logic 49*, 3 (1984), 867–879.

[58] Jäger, G. *Theories for Admissible Sets: A Unifying Approach to Proof Theory*. Bibliopolis, Napoli, 1986.

[59] Jäger, G. Induction in the elementary theory of types and names. In *Computer Science Logic '87*, E. Börger, H. Kleine Büning, and M.M. Richter, Eds., vol. 329 of *Lecture Notes in Computer Science*. Springer, Berlin, 1988, pp. 118–128.

[60] Jäger, G. Type theory and explicit mathematics. In *Logic Colloquium '87*, H.-D. Ebbinghaus, J. Fernandez-Prida, M. Garrido, M. Lascar, and M. R. Artalejo, Eds. North Holland, Amsterdam, 1989, pp. 117–135.

[61] Jäger, G. Fixed points in Peano arithmetic with ordinals. *Annals of Pure and Applied Logic 60*, 2 (1993), 119–132.

[62] Jäger, G., Kahle, R., Setzer, A., and Strahm, T. The proof-theoretic analysis of transfinitely iterated fixed point theories. *Journal of Symbolic Logic 64*, 1 (1999), 53–67.

[63] Jäger, G., Kahle, R., and Strahm, T. On applicative theories. In *Logic and Foundations of Mathematics*, A. Cantini, E. Casari, and P. Minari, Eds. Kluwer, 1999, pp. 83–92.

[64] Jäger, G., Kahle, R., and Studer, T. Universes in explicit mathematics. *Annals of Pure and Applied Logic*. To appear.

[65] Jäger, G., and Pohlers, W. Eine beweistheoretische Untersuchung von $(\Delta^1_2\text{-}\mathsf{CA}) + (\mathsf{BI})$ und verwandter Systeme. In *Sitzungsberichte der Bayerischen Akademie der Wissenschaften, Mathematisch-naturwissenschaftliche Klasse*. 1982, pp. 1–28.

[66] Jäger, G., and Strahm, T. Upper bounds for metapredicative Mahlo in explicit mathematics and admissible set theory. *Journal of Symbolic Logic*. To appear.

[67] Jäger, G., and Strahm, T. Second order theories with ordinals and elementary comprehension. *Archive for Mathematical Logic 34*, 6 (1995), 345–375.

[68] Jäger, G., and Strahm, T. Totality in applicative theories. *Annals of Pure and Applied Logic 74*, 2 (1995), 105–120.

[69] Jäger, G., and Strahm, T. Some theories with positive induction of ordinal strength $\varphi\omega 0$. *Journal of Symbolic Logic 61*, 3 (1996), 818–842.

[70] Jäger, G., and Strahm, T. The proof-theoretic strength of the Suslin operator in applicative theories. Preprint, Aug. 1999. To appear in *Festschrift for Solomon Feferman*.

[71] Jäger, G., and Strahm, T. Fixed point theories and dependent choice. *Archive for Mathematical Logic 39* (2000), 493–508.

[72] Jäger, G., and Studer, T. Extending the system T_0 of explicit mathematics: the limit and Mahlo axioms. Submitted for publication.

[73] Kahle, R. *Applikative Theorien und Frege-Strukturen.* PhD thesis, Institut für Informatik und angewandte Mathematik, Universität Bern, 1997.

[74] Kahle, R. Uniform limit in explicit mathematics with universes. Tech. Rep. IAM-97-002, Institut für Informatik und angewandte Mathematik, Universität Bern, 1997.

[75] Kapron, B., and Cook, S. A new characterization of type 2 feasibility. *SIAM Journal on Computing 25* (1996), 117–132.

[76] Krajíček, J. *Bounded Arithmetic, Propositional Logic, and Complexity Theory*, vol. 60 of *Encyclopedia of Mathematics and its Applications.* Cambridge University Press, 1995.

[77] Leivant, D. A foundational delineation of poly-time. *Information and Computation 110* (1994), 391–420.

[78] Leivant, D. Predicative recurrence in finite type. In *Logical Foundations of Computer Science*, A. Nerode and Y. Matiyasevich, Eds.,

vol. 813 of *Lecture Notes in Computer Science*. Springer, 1994, pp. 227–239.

[79] Leivant, D. Ramified recurrence and computational complexity I: Word recurrence and poly-time. In *Feasible Mathematics II*, P. Clote and J. Remmel, Eds. Birkhäuser, 1994, pp. 320–343.

[80] Martin-Löf, P. An intuitionistic theory of types: predicative part. In *Logic Colloquium '73*, H. Rose and J. Sheperdson, Eds. North Holland, 1975, pp. 73–118.

[81] Martin-Löf, P. *Intutionistic Type Theory*, vol. 1 of *Studies in Proof Theory*. Bibliopolis, 1984.

[82] Marzetta, M. *Predicative Theories of Types and Names*. PhD thesis, Institut für Informatik und angewandte Mathematik, Universität Bern, 1993.

[83] Marzetta, M. Universes in the theory of types and names. In *Computer Science Logic '92*, E. Börger et al., Ed., vol. 702 of *Lecture Notes in Computer Science*. Springer, Berlin, 1993, pp. 340–351.

[84] Marzetta, M., and Strahm, T. The μ quantification operator in explicit mathematics with universes and iterated fixed point theories with ordinals. *Archive for Mathematical Logic 37*, 5+6 (1998), 391–413.

[85] Melhorn, K. Polynomial and abstract subrecursive classes. *Journal of Computer and System Science 12* (1976), 147–178.

[86] Odifreddi, P. *Classical Recursion Theory*. North-Holland, Amsterdam, 1989.

[87] Odifreddi, P. *Classical Recursion Theory Vol. II*. North-Holland, 1999.

[88] Palmgren, E. On universes in type theory. In *Twentyfive Years of Constructive Type Theory*, G. Sambin and J. Smith, Eds. Oxford University Press, 1998.

[89] Papadimitriou, C. *Computational Complexity*. Addison-Wesley, 1994.

[90] Parsons, C. On a number theoretic choice schema and its relation to induction. In *Intuitionism and Proof Theory, Proceedings of the Summer Conference at Buffalo N.Y., 1968*, J. Myhill, A. Kino, and R. Vesley, Eds. North Holland, Amsterdam, 1970, pp. 459–473.

[91] Pezzoli, E. On the computational complexity of type 2 functionals. In *Computer Science Logic '97*, vol. 1414 of *Lecture Notes in Computer Science*. Springer, 1998, pp. 373–388.

[92] Pohlers, W. *Proof Theory: An Introduction*, vol. 1407 of *Lecture Notes in Mathematics*. Springer, Berlin, 1989.

[93] Pohlers, W. Subsystems of set theory and second order number theory. In *Handbook of Proof Theory*, S. R. Buss, Ed. North Holland, 1998, pp. 209–335.

[94] Rathjen, M. Proof-theoretic analysis of KPM. *Archive for Mathematical Logic 30* (1991), 377–403.

[95] Rathjen, M. Collapsing functions based on recursively large ordinals: A wellordering proof for KPM. *Archive for Mathematical Logic 33* (1994).

[96] Rathjen, M. The realm of ordinal analysis. In *Sets and Proofs*, S. B. Cooper and J. Truss, Eds. Cambridge University Press, 1999, pp. 219–279.

[97] Rathjen, M. The strength of Martin-Löf type theory with a superuniverse. Part I. *Archive for Mathematical Logic 39*, 1 (2000), 1–39.

[98] Rathjen, M. The superjump in Martin-Löf type theory. In *Logic Colloquium '98*, S. Buss, P. Hájek, and P. Pudlák, Eds., vol. 13 of *Lecture Notes in Logic*. Association for Symbolic Logic, 2000, pp. 363–386.

[99] Rathjen, M. The strength of Martin-Löf type theory with a superuniverse. Part II. *Archive for Mathematical Logic*. To appear.

[100] Richter, W., and Aczel, P. Inductive definitions and reflecting properties of admissible ordinals. In *Generalized recursion theory*, J. Fenstad and P. Hinman, Eds. North Holland, Amsterdam, 1974, pp. 301–381.

[101] Ritchie, R. W. Classes of predictably computable functions. *Transactions of the American Mathematical Society 106* (1963), 139–173.

[102] Royer, J. Semantics vs. syntax vs. computations: Machine models for type-2 polynomial-time bounded functionals. *Journal of Computer and System Science 54* (1997), 424–436.

[103] Rüede, C. Metapredicative subsystems of analysis. Submitted for publication.

[104] Rüede, C. The proof-theoretic analysis of Σ^1_1 transfinite dependent choice. Submitted for publication.

[105] Rüede, C. Universes in metapredicative analysis. Submitted for publication.

[106] Rüede, C. *Metapredicative subsystems of analysis*. PhD thesis, Institut für Informatik und angewandte Mathematik, Univeristät Bern, 2000.

[107] Rüede, C., and Strahm, T. Intuitionistic fixed point theories for strictly positive operators, 2000. Submitted for publication.

[108] Schlüter, A. An extension of Leivant's characterization of polytime by predicative arithmetic. Preprint, Stanford Univeristy, 1995. 13 pages.

[109] Schütte, K. Kennzeichnung von Ordnungszahlen durch rekursiv erklärte Funktionen. *Mathematische Annalen 127* (1954), 15–32.

[110] Schütte, K. *Proof Theory*. Springer, Berlin, 1977.

[111] Seth, A. *Complexity Theory of Higher Type Functionals*. PhD thesis, Tata Institute of Fundamental Research, Bombay, 1994.

[112] Setzer, A. *Proof-theoretical analysis of Martin-Löf type theory with W type and one universe*. PhD thesis, LMU Munich, 1993.

[113] Setzer, A. Extending Martin-Löf type theory by one Mahlo universe. *Archive for Mathematical Logic 39*, 3 (2000), 155–181.

[114] Simpson, S. G. Σ^1_1 and Π^1_1 transfinite induction. In *Logic Colloquium '80*, D. van Dalen, D. Lascar, and J. Smiley, Eds. North Holland, Amsterdam, 1982, pp. 239–253.

[115] Simpson, S. G. *Subsystems of Second Order Arithmetic*. Perspectives in Mathematical Logic. Springer-Verlag, 1998.

[116] Stärk, R. F. Call-by-value, call-by-name and the logic of values. In *Computer Science Logic CSL '96: Selected Papers*, no. 1258 in Lecture Notes in Computer Science. Springer, March 1997, pp. 431–445.

[117] Strahm, T. Theories with self-application and computational complexity. Submitted for publication.

[118] Strahm, T. Wellordering proofs for metapredicative Mahlo. Submitted for publication.

[119] Strahm, T. *On the Proof Theory of Applicative Theories*. PhD thesis, Institut für Informatik und angewandte Mathematik, Universität Bern, 1996.

[120] Strahm, T. Polynomial time operations in explicit mathematics. *Journal of Symbolic Logic 62*, 2 (1997), 575–594.

[121] Strahm, T. First steps into metapredicativity in explicit mathematics. In *Sets and Proofs*, S. B. Cooper and J. Truss, Eds. Cambridge University Press, 1999, pp. 383–402.

[122] Strahm, T. Autonomous fixed point progressions and fixed point transfinite recursion. In *Logic Colloquium '98*, S. Buss, P. Hájek, and P. Pudlák, Eds., vol. 13 of *Lecture Notes in Logic*. Association for Symbolic Logic, 2000, pp. 449–464.

[123] Strahm, T. The non-constructive μ operator, fixed point theories with ordinals, and the bar rule. *Annals of Pure and Applied Logic 104*, 1–3 (2000), 305–324.

[124] Studer, T. *The mathematics of objects*. PhD thesis, Universität Bern, 2001. To appear.

[125] Talcott, C. A theory for program and data type specification. *Theoretical Computer Science 104* (1992), 129–159.

[126] Thompson, D. B. Subrecursiveness: machine independet notions of computability in restricted time and storage. *Mathematical Systems Theory 6* (1972), 3–15.

[127] Troelstra, A., and van Dalen, D. *Constructivism in Mathematics*, vol. I. North-Holland, Amsterdam, 1988.

[128] Troelstra, A., and van Dalen, D. *Constructivism in Mathematics*, vol. II. North Holland, Amsterdam, 1988.

[129] Tupailo, S. Realization of constructive set theory into explicit mathematics: a lower bound for impredicative Mahlo universe. Submitted for publication.

List of symbols

The following list of symbols is divided into three separate tables: formal systems, axioms and rules, and other symbols. The symbols in all three tables are given in the order of their appearance in the text.

A Formal systems

ETJ, 21	explicit elementary type theory plus join
EIN, 27	explicit inaccessibility
EMA, 27	explicit Mahloness
PA, 32	Peano arithmetic
ACA_0, 32	restricted arithmetical comprehension
$\Pi^1_0\text{-}\mathsf{CA}_{<\varepsilon_0}$, 33, 80	arithmetical comprehension iterated below ε_0
ATR_0, 33	restricted arithmetical transfinite recursion
S_n, 41	n-hyperuniverses
PA_Ω, $\mathsf{PA}^{\mathsf{r}}_\Omega$, 49, 64	Jäger's PA with ordinals
OAD, 51	ordinal theory (admissible)
OIN, 52	ordinal theory (recursively inaccessible)
OMA, 52	ordinal theory (Mahlo)
H, 65	basic semiformal system for ordinal theories
$\mathsf{H}[\alpha]$, 66	H plus α many admissibles
KPi^0, 67, 80	recursive inaccessibility without foundation
$\mathsf{OMA}^{\mathsf{T}}$, 71	Tait-style reformulation of OMA
$\mathsf{H}[S, n, \alpha]$, 74	$\mathsf{H}[S]$ plus α many n-inaccessibles
$\Sigma^1_1\text{-}\mathsf{AC}_0$, $\Sigma^1_1\text{-}\mathsf{AC}$, 79, 80	restricted and full Σ^1_1 axiom of choice
KPu^0, 80	Kripke Platek set theory without foundation
ML, 80	Martin-Löf type theory
HA, 80	Heyting arithmetic

B Axioms and rules

C Other symbols