Ordinal Notations and Well-Orderings in Bounded Arithmetic

Arnold Beckmann[1,3] Samuel R. Buss[1,4] Chris Pollett[2]

June 25, 2002

1 Introduction

Ordinal notations and provability of well-foundedness have been a central tool in the study of the consistency strength and computational strength of formal theories of arithmetic. This development began with Gentzen's consistency proof for Peano arithmetic based on the well-foundedness of ordinal notations up to ϵ_0. Since the work of Gentzen, ordinal notations and provable well-foundedness have been studied extensively for many other formal systems, some stronger and some weaker than Peano arithmetic. In the present paper, we investigate the provability and non-provability of well-foundedness of ordinal notations in very weak theories of bounded arithmetic, notably the theories S^i_2 and T^i_2 with $1 \leq i \leq 2$. We prove several results about the provability of well-foundedness for ordinal notations; our main results state that for the usual ordinal notations for ordinals below ϵ_0 and Γ_0, the theories T^1_2 and S^2_2 can prove the ordinal Σ^b_1-minimization principle over a bounded domain. PLS is the class of functions computed by a polynomial local search to minimize a cost function. It is a corollary of our theorems that the cost function can be allowed to take on ordinal values below Γ_0, without increasing the class PLS.

The historical development of ordinal notations and formal theories of arithmetic is far too extensive for us to survey here. We shall include the basic definitions for ordinal notations of ordinals below ϵ_0 and Γ_0, and the reader can refer to Feferman [6, 7] or the textbooks of Schütte [12] or Pohlers [11] for more details.

Theories of bounded arithmetic are fragments of Peano arithmetic which have induction strongly restricted, firstly to allow induction only on certain

[1]Dept. of Mathematics, Univ. of California, San Diego, La Jolla, CA 92093-0112.

[2]Computer Science Dept., San Jose State Univ., One Washington Square, San Jose, CA 95192. Work performed while at the Dept. of Mathematics, UCLA.

[3]Supported by the Deutschen Akademie der Naturforscher Leopoldina grant #BMBF-LPD 9801-7 with funds from the Bundesministeriums für Bildung, Wissenschaft, Forschung und Technologie.

[4]Supported in part by NSF grants DMS-0100589 and DMS-9803515 and by a cooperative research grant INT-9600919/ME-103 of the NSF (USA) and the MŠMT (Czech Republic).

types of bounded formulas, and secondly, in the case of the theories S_2^i, to allow only length induction instead of the usual successor induction. In order to get meaningful results, we need to work with theories such as S_2^1, T_2^1, S_2^2 and T_2^2 as introduced by the second author in [3]. We presume the reader is familiar with these theories of bounded arithmetic: the necessary background can be found in [3, 4, 8, 10].

Ordinal notations have been extensively used in the study of strong theories, ranging from primitive recursive arithmetic, to Peano arithmetic and to fragments of second-order arithmetic; however, there has been little prior work relating ordinals to proof systems as weak as fragments of bounded arithmetic. This is due, in part, to the fact that the fragments of bounded arithmetic have computational complexity related to (near) feasible classes such as polynomial time and the various levels of the polynomial time hierarchy. There are no good mechanisms available to combine such low-level complexity with ordinal recursion, and thus traditional results on ordinal notations have not transferred to the setting of bounded arithmetic. Sommer [14] investigated the formalizability of abstract ordinal notations in full bounded arithmetic, $I\Delta_0$, and showed that $I\Delta_0$ can represent ordinals up to Γ_0 with the ordinal operations of addition and the Veblen φ function. Beckmann introduced *dynamic ordinals* in fragments of the bounded arithmetic as a tool to analyze predicative bounded arithmetic and to obtain relativized separation results for bounded arithmetic theories [1]. His dynamic ordinals are based on exponential notations for integers, which are similar to the Cantor normal form representation of ordinals less than ϵ_0.

It is known (c.f. [13]) that if one formalizes transfinite induction on ordinals in the usual way, then induction on ordinals below ω^ω (or even induction on just ω^2) is sufficient to give the full strength of primitive recursive arithmetic. Thus, in order to get meaningful ordinal well-foundedness results for fragments of bounded arithmetic it is necessary to limit the strength of the well-foundedness principles. We shall do this by restricting to a finite domain as follows. Let T be a first-order theory of bounded arithmetic and $\prec$ be a binary relation which T-provably defines a total ordering on a domain $D \subseteq \mathbb{N}$. In practice, we will require that $\prec$ and D are polynomial time recognizable predicates, defined by Δ_1^b-formulas. We say that $\prec$ is well-founded on bounded domains provided that whenever $A(x)$ is a predicate and there is a $d \in D$ such that $d \leq t$ and $A(d)$, then there is a $\prec$-least element $d \in D$ satisfying both $d \leq t$ and $A(d)$. In symbols, we write this principle as the formula $WF_A(t)$:

$$\begin{aligned}&(\exists d \leq t)(d \in D \wedge A(d)) \rightarrow \\ &\quad (\exists d \leq t)[d \in D \wedge A(d) \wedge (\forall d' \leq t)(d' \in D \wedge A(d') \rightarrow d' \not\prec d)].\end{aligned} \tag{1}$$

It is of course a triviality that every total order is well-founded on bounded domains, since bounded domains are finite sets. However, if a theory T can prove (1) for all Σ_1^b-formulas $A(x)$ then we say that $\prec$ is *provably well-founded on bounded domains* in T. [5] We let $WF_\prec$ denote the axiom scheme containing the formulas WF_A for all Σ_1^b-formulas A.

[5]One could consider more general concepts, e.g., Σ_i^b-well-foundedness on bounded domains

The use of the domain D in the above definition is merely a convenience; there would be no loss in generality in taking $D = \mathbb{N}$. Indeed, the ordering $\prec$ can always be extended to have domain all integers by making $d \prec x$ for all $d \in D$ and $x \notin D$. The reason for using the domain D is that we will consider natural orderings on ordinal notations for ordinals below ϵ_0 and Γ_0: it will be convenient to take D to be the set of valid Gödel numbers for ordinal notations.

Very recently (and independently of our own work), Sommer has announced sharpened versions of the above-mentioned results from [13]. In our notation, what he has done is modify the quantifier $(\forall d' \le t)$ of formula (1) by replacing t with a bound $f(d)$ which depends on d (and remove the other uses of t). We know no direct connection of this to our well-ordering principles on bounded domains.

The outline of the present paper is as follows. In the next section, we show first that T^2_2 is strong enough to prove the well-foundedness on bounded domains of any Δ^b_1-defined total ordering. On the other hand, we show that if the ordering is defined with an oracle, then T^1_2 and S^2_2 cannot prove the well-foundedness on bounded domains of the ordering. Section 3 considers the usual Cantor normal form representation of ordinals below ϵ_0: First we show that the well-foundedness of this system is not provable in S^1_2 unless $S^1_2 = T^1_2$. Second, we prove that T^1_2 can prove the well-foundedness of these ordinals on bounded domains, by giving an order-preserving embedding of the ordinals less than ϵ_0 restricted to a bounded number of operators into $\mathbb{N}^k$ for some integer k. Section 4 introduces the usual Veblen φ-function notation for ordinals less than Γ_0. It is shown that S^1_2 can Δ^b_1 define the ordering on ordinals below Γ_0 by giving straightforward polynomial time algorithms (this re-obtains, more concretely, some results of Sommer [14]). In addition, S^1_2 can define the normal form on ordinals below Γ_0 and prove that normal forms exist and are unique. After this, we reach the central results of the paper: we prove that T^1_2 and S^2_2 can prove the well-foundedness on bounded domains of usual ordinal notations for ordinals below Γ_0.

Section 5 is a short diversion discussing order-preserving embeddings of ordinals into the set of lexicographically-ordered words over a finite alphabet.

Finally, in section 6, we discuss a corollary of our theorems for the class Polynomial Local Search, PLS, introduced by Johnson, Papadimitriou and Yannakakis [9]. Buss-Krajíček [5] showed that the Σ^b_1-definable functions of T^1_2 are precisely the (multivalued) functions which are definable as the composition of a projection function and a PLS function. We define the classes $(\epsilon_0, \preccurlyeq)$-PLS and $(\Gamma_0, \preccurlyeq)$-PLS by generalizing PLS to allow the cost function to take on ordinal values less than ϵ_0 or Γ_0, respectively. It is a corollary of our theorems on the provability of well-foundedness of ordinal notations in T^1_2, that the classes $(\epsilon_0, \preccurlyeq)$-PLS and $(\Gamma_0, \preccurlyeq)$-PLS are identical to the class PLS.

We thank R. Sommer for useful discussions and remarks, which lead to a considerable simplification of some of the proofs.

defined with A ranging over Σ^b_i-formulas. However, we shall not consider these generalizations in this paper.

2 General orderings

This section states a couple results about general orderings. By a "general ordering" we mean any order defined by a Δ_1^b-formula; by comparison the results of sections 3 and 4 concern specific natural well-orderings based on ordinal notations.

To formalize general well-orderings, it is best to introduce $\prec$ as new relation symbol in the language. Let $\mathrm{Total}(\prec)$ be the formula expressing the condition that $\prec$ is a total ordering, i.e., that $\prec$ is transitive and satisfies trichotomy. This can be formalized by

$$\begin{aligned}(\forall x)(\forall y)(\forall z)[(x \not\prec x) \wedge (x \prec y \vee x = y \vee y \prec x) \\ \wedge (x \prec y \wedge y \prec z \rightarrow x \prec z)]\end{aligned}$$

Note that $\mathrm{Total}(\prec)$ is a $\forall\Delta_0^b(\prec)$-formula. Let $T_2^2(\prec)+\mathrm{Total}(\prec)$ be the theory of bounded arithmetic axiomatized by the BASIC axioms, the successor induction axioms (IND) for $\Sigma_2^b(\prec)$ formulas and the axiom $\mathrm{Total}(\prec)$.

Theorem 1 *The ordering $\prec$ is provably well-founded on bounded domains over $T_2^2(\prec) + \mathrm{Total}(\prec)$.*

Proof Fix $A(x) \in \Sigma_1^b$ (or even, $A \in \Sigma_1^b(\prec)$). Clearly, $WF_A(y)$ is a $\Sigma_2^b(\prec)$-formula. Let T be the theory $T_2^2(\prec) + \mathrm{Total}(\prec)$. Clearly, $T \vdash WF_A(0)$ because the trichotomy of $\prec$ is implied by the axiom $\mathrm{Total}(\prec)$. Also, T proves $WF_A(y) \rightarrow WF_A(y{+}1)$. Thus, by $\Sigma_2^b(\prec)$-IND, $T \vdash (\forall y)WF_A(y)$, which is what we needed to prove. □

For weaker theories, we have:

Theorem 2 *The ordering $\prec$ is not provably well-founded on bounded domains over $S_2^2(\prec) + \mathrm{Total}(\prec)$. In fact, $S_2^2(\prec) + \mathrm{Total}(\prec)$ does not prove $(\forall y)WF_A(y)$ even for the case where $A(x)$ is the universally true formula $x = x$.*

Theorem 2 holds for $T_2^1(\prec) + \mathrm{Total}(\prec)$ as well, since $T_2^1 \subseteq S_2^2$.

Proof Our proof exploits a proof technique of Krajíček [10, Thm. 11.2.5]. Let S be the theory $S_2^2(\prec) + \mathrm{Total}(\prec)$. Let $A(x)$ be the formula "$x = x$". (In fact any formula $A(x)$ which is true for a superpolynomial density of x's could be used.) If S proves $(\forall y)WF_A(y)$, then S can also prove the assertion

$$(\forall y)(\exists x \leq y)(\forall z \leq y)(x \preccurlyeq z). \tag{2}$$

Suppose for the sake of obtaining a contradiction, that S can prove (2). This is a $\forall\Sigma_2^b$-formula; therefore, by the relativization of the 'main theorem' for S_2^2, there is a $\square_2^p(\prec) = P^{NP(\prec)}$-function f which, given an input y, produces a $\prec$-least x below y [3]. The function f runs in polynomial time and uses an $NP(\prec)$-oracle.[6]

[6] One might also let the computation of f query the predicate $\prec$, but this is not necessary, since these queries may be made indirectly through the $NP(\prec)$ oracle.

The $NP(\prec)$ oracle is a nondeterministic oracle Turing machine $M(u)$, which on input u returns *True* iff it has an accepting computation. The machine M is allowed to make queries of the form "$v \prec w$?" to the oracle $\prec$. In addition, f is Σ_2^b-defined by S, and S can prove all the relevant properties of f and the machine M. In particular, S can prove $\forall y \forall z \leq y (f(y) \preccurlyeq z)$.

Since $f(y)$ runs in polynomial time, say in time $p(|y|)$, any query u to M made during the computation of f must have $|u| \leq p(|y|)$. M also is polynomial time, so there is a polynomial q bounding the runtime of M. In particular, any particular computation path of $M(u)$ can make at most $q(|u|)$ queries "$v \prec w$?" to the oracle.

To prove Theorem 2, it will suffice to construct an oracle $\prec$ for which f fails to correctly produce the $\prec$-least $x \leq y$. To do this, choose y sufficiently large, and run the polynomial time algorithm for computing $f(y)$. As we compute $f(y)$, we construct a series of linear orders $\prec_i$, $i = 0, 1, 2, 3, \ldots$. The order $\prec_i$ will have domain D_i and with $D_i \subseteq D_{i+1}$ and each $\prec_i$ will be the restriction of $\prec_{i+1}$ to the domain D_i.

Initially, we set $\prec_0$ to be the empty binary relation with domain $D_0 = \emptyset$. When $f(y)$ makes its $(i+1)$-st query, u, to M, we define $\prec_{i+1}$. To define $\prec_{i+1}$, first consider the case that there is some total, linear order $\prec$ extending $\prec_i$ such that that $M(u)$ accepts relative to $\prec$. Choose, arbitrarily, such a $\prec$ and an accepting computation of $M(u)$ relative to u. Then, let D_{i+1} equal D_i union the set of values v, w such that $M(u)$ made a query "$v \prec w$?" in this accepting computation. Let $\prec_{i+1}$ equal $\prec$ restricted to the domain D_{i+1}. This same computation path will cause $M(u)$ to accept relative to any linear ordering containing $\prec_{i+1}$. In this case, the computation of $f(u)$ then continues with a *True* answer from the oracle. In the second case, there is no such ordering $\prec$: set $\prec_{i+1}=\prec_i$ and $D_{i+1} = D_i$. $M(u)$ will not accept relative to any linear ordering extending $\prec_{i+1}$. In this case, the computation of $f(y)$ is continued with a *False* answer from the oracle query.

Note that in either case, D_{i+1} has at most $2q(|u|) \leq 2q(p(|y|))$ new elements over D_i. At the end of the computation f outputs a value x. The computation made $k \leq p(|y|)$ queries to $M(u)$, so we obtain a linear order $\prec_k$ with domain D_k of cardinality at most $2p(|y|) \cdot q(p(|y|))$. If $\prec$ is any total linear order with domain $\mathbb{N}$ which extends $\prec_k$, then the computation of f relative to $\prec$ will will have the same answers to its oracle queries and hence must output the same value x. If y was sufficiently large, namely if $y > 2p(|y|)q(p(|y|)) + 1$, then we can extend $\prec_k$ to an ordering $\prec$ such that there is some $z \leq y$ with $z \prec x$: this is done by choosing $z \in [0, y] \setminus (D_k \cup \{x\})$ and letting z be the least element of $\prec$. This contradicts the fact that $f(y)$ was to produce the $\prec$-least element $x \leq y$. □

We do not know of any way to extend Theorem 2 to a non-oracle style result. For instance, does there exist a Δ_1^b-definition of a binary predicate $\prec$ such that $S_2^1 \vdash \text{Total}(\prec)$ and such that the theory $S_2^1 + WF_\prec$ includes all the $\forall\Sigma_2^b$-consequences of T_2^2?

3 Ordinals below ϵ_0

3.1 Ordinal notations for ϵ_0

We use $<$ and $\leq$ to denote the usual ordering on ordinals; i.e., $<$ denotes the 'real' semantic concept of ordinal orderings, and we will reserve the symbol $\prec$ for syntactically defined orderings on Gödel numbers of ordinals. We reserve lowercase Greek letters to denote ordinals or Gödel numbers of ordinals. Recall the Cantor normal form for ordinals; i.e., every ordinal $\alpha > 0$ can be written uniquely in the form

$$\alpha = \omega^{\alpha_1} + \omega^{\alpha_2} + \omega^{\alpha_3} + \cdots + \omega^{\alpha_k},$$

where $k \geq 1$ and $\alpha_1 \geq \alpha_2 \geq \alpha_3 \geq \cdots \geq \alpha_k$. This is the basis for the well-known representation of ordinals less than ϵ_0: namely, write an ordinal $\alpha < \epsilon_0$ as a term in Cantor normal form, recursively writing the exponents of ω in the same form.

This gives a syntactic representation of ordinals less than ϵ_0. We need to formalize this syntactic representation in the bounded arithmetic theory S^1_2, by defining a set D which is the set of Gödel numbers of (syntactic representations of) ordinals less than ϵ_0 and a binary formula $\prec$ which defines the ordinal ordering on the Gödel numbers. The formulas D and $\prec$ need to be Δ^b_1-formulas, that is to say, polynomial time computable, and S^1_2 needs to be able to prove that $\prec$ defines a total ordering on the domain D.

The syntactic representation in S^1_2 is developed in two stages: first representing ordinals in non-normal form, and then showing that ordinals can be converted to the normal form. Ordinals will first be represented in a 'basic form' and then to make our results more general, in a 'compact form'. As an example of the difference,

$$\omega^{\omega^0+\omega^0+\omega^0} + \omega^{\omega^0+\omega^0+\omega^0}$$

is in basic form, and its compact form is $\omega^{\omega^0 \cdot 3} \cdot 2$.

Definition The set of *basic forms* for ordinals less than ϵ_0 is the set of expressions inductively defined as follows:

1. 0 is a basic form.
2. If α is a basic form, then so is ω^α. The expression ω^α is called an *ω-term.*
3. If α and β are basic forms, and α is an ω-term and $\beta \neq 0$, then $\alpha + \beta$ is a basic form.

S^1_2 can formalize the notion of basic form by using standard sequence coding methods to define the Gödel number of a basic form. We assume that some efficient method of sequence coding is used for Gödel numbers so that the length of the Gödel number of a basic form α is proportional to the number of symbols in α.

It is immediate from the definition that every non-zero basic form α can be written uniquely in its *additive expansion*

$$\alpha \;=\; \alpha_1 + \alpha_2 + \cdots + \alpha_k,$$

where each α_i is an ω-term. Note that S^1_2 is able to prove the existence and uniqueness of the additive expansion of a basic form. To put basic forms into a normal form, we wish to further require that the ordinals $\alpha_1, \ldots, \alpha_k$ form a non-increasing sequence. To formalize this, we first need to define a syntactic order, denoted $\prec$, on basic forms. Unfortunately, the definition of the $\prec$ is complicated by the fact that basic forms are not (yet) in normal form.

Definition Let α and β be basic forms. We inductively define $\alpha \preccurlyeq \beta$ and $\alpha \not\preccurlyeq \beta$, letting $\alpha \approx \beta$ mean that both $\alpha \preccurlyeq \beta$ and $\beta \preccurlyeq \alpha$, and $\alpha \prec \beta$ mean that $\alpha \preccurlyeq \beta$ and $\alpha \not\approx \beta$. First if $\alpha = 0$, then $\alpha \preccurlyeq \beta$. Also, if $\beta = 0$ and $\alpha \neq 0$, then $\alpha \not\preccurlyeq \beta$. Otherwise, let $\alpha_1 + \cdots + \alpha_k$ and $\beta_1 + \cdots + \beta_\ell$ be the additive expansions of α and β. Then

a. If $k = \ell = 1$, and $\alpha_1 = \omega^{\alpha'}$ and $\beta_1 = \omega^{\beta'}$, then $\alpha \preccurlyeq \beta$ is defined to hold if and only if $\alpha' \preccurlyeq \beta'$.

b. If $\alpha_1 \prec \alpha_i$ for some $i > 1$, then $\alpha \preccurlyeq \beta$ if and only if $\alpha_2 + \ldots + \alpha_k \preccurlyeq \beta$.

c. If $\beta_1 \prec \beta_i$ for some $i > 1$, then $\alpha \preccurlyeq \beta$ if and only if $\alpha \preccurlyeq \beta_2 + \ldots + \beta_\ell$.

d. Otherwise, if none of a., b., or c. apply, then $\alpha \preccurlyeq \beta$ iff either (1) $\alpha_1 \prec \beta_1$ or (2) $\alpha_1 \approx \beta_1$ and $\alpha_2 + \cdots + \alpha_k \preccurlyeq \beta_2 + \cdots + \beta_\ell$. (If either k or ℓ are one, then the empty sum is interpreted as 0.)

It is easy to show that $\preccurlyeq$ is transitive and reflexive and satisfies dichotomy, i.e., for all basic forms α, β, γ,

$$\begin{aligned}
&\alpha \preccurlyeq \alpha, \\
&\alpha \preccurlyeq \beta \lor \beta \preccurlyeq \alpha, \\
&\alpha \preccurlyeq \beta \land \beta \preccurlyeq \gamma \to \alpha \preccurlyeq \gamma.
\end{aligned}$$

Likewise, $\prec$ is transitive and antisymmetric. The proofs of these facts use ordinary integer induction and do not require transfinite induction. To make $\prec$ a linear (non-partial) ordering, we need to mod out by the $\approx$ relation. The best way to do this is identify normal forms for ordinal notations:

Definition The set of *normal basic forms* for ordinals less than ϵ_0 is the set of expressions inductively defined as follows:

1. 0 is a normal basic form.

2. If α is a normal basic form, then so is ω^α.

3′. Let α and β be normal basic forms, α be an ω-term and β be non-zero with β' the leading term in the additive expansion of β. If $\beta' \preccurlyeq \alpha$, then $\alpha + \beta$ is a normal basic form.

It is straightforward to see that S^1_2 can Δ^b_1-define the syntactic concepts of $\prec$, $\preccurlyeq$ on basic forms, for instance using the techniques of [3, Ch. 7] for inductive definitions S^1_2, or of [1, 2] for defining exponential notations. Probably the most straightforward way to define these concepts in S^1_2 is to follow [1, 2] and use the following polynomial time algorithm for determining if $\alpha \preccurlyeq \beta$: Given α and β as inputs, the algorithm first forms all pairs (α', β') with α' a subterm of α and β' a subterm of β. Then, starting with shorter subterms, the algorithm decides for each such pair, whether $\alpha' \prec \beta'$ or $\alpha' \approx \beta'$ or $\beta' \prec \alpha'$. This algorithm can readily be formalized in S^1_2, and based on this algorithm, S^1_2 can prove properties of $\preccurlyeq$ such as transitivity, reflexivity and dichotomy.

In addition, S^1_2 can Δ^b_1-define the set of (Gödel numbers of) normal basic forms. Finally, S^1_2 can prove that $\prec$ is a total ordering on the normal basic forms, satisfying transitivity and trichotomy.

We now define the compact representations for ordinals less than ϵ_0.

Definition The set of *compact forms* for ordinals less than ϵ_0 is the set of expressions inductively defined as follows:

1. 0 is a compact form.
2. If α is a compact form and $n \in \mathbb{N}, n > 0$, then $\omega^\alpha \cdot n$ is a compact form. This is called an *ω-term.*
3. If α and β are compact forms, and α is an ω-term and $\beta \neq 0$, then $\alpha + \beta$ is a compact form.

The definition of $\prec$ for compact forms is complicated by the fact that we not only need to discard additive terms that are 'out of order', but also need to collect together the integer coefficients of the maximum additive term.

Definition Let α and β be compact forms. $\alpha \preccurlyeq \beta$ and $\alpha \not\preccurlyeq \beta$ are inductively defined as follows. As before, let $\alpha \approx \beta$ mean that $\alpha \preccurlyeq \beta$ and $\beta \preccurlyeq \alpha$, and $\alpha \prec \beta$ mean that $\alpha \preccurlyeq \beta$ and $\alpha \not\approx \beta$. Also as before, if $\alpha = 0$, then $\alpha \preccurlyeq \beta$; and if $\beta = 0$ and $\alpha \neq 0$, then $\alpha \not\preccurlyeq \beta$. Otherwise, let $\alpha_1 + \cdots + \alpha_k$ and $\beta_1 + \cdots + \beta_\ell$ be the additive expansions of α and β, where $\alpha_i = \omega^{\alpha'_i} \cdot n_i$ and $\beta_i = \omega^{\beta'_i} \cdot m_i$.

a. If $k = \ell = 1$, then $\alpha \preccurlyeq \beta$ is defined to hold if and only if either (1) $\alpha'_1 \prec \beta'_1$ or (2) $\alpha'_1 \approx \beta'_1$ and $n_1 \leq m_1$.

b. If $\alpha'_1 \prec \alpha'_i$ for some $i > 1$, then $\alpha \preccurlyeq \beta$ if and only if $\alpha_2 + \ldots + \alpha_k \preccurlyeq \beta$.

c. If $\beta'_1 \prec \beta'_i$ for some $i > 1$, then $\alpha \preccurlyeq \beta$ if and only if $\alpha \preccurlyeq \beta_2 + \ldots + \beta_\ell$.

d. Otherwise, if none of a., b. or c. apply, and if $\alpha'_1 \not\approx \beta'_1$, then $\alpha \preccurlyeq \beta$ iff $\alpha'_1 \preccurlyeq \beta'_1$.

e. If none of a., b., c., or d. apply, then let $S = \{i : \alpha'_i \approx \alpha'_1\}$ and let $T = \{i : \beta'_i \approx \beta'_1\}$. Let $n_S = \sum_{i \in S} n_i$ and $m_T = \sum_{i \in T} m_i$. If $n_S < m_T$, then $\alpha \preccurlyeq \beta$. If $n_S > m_T$, then $\alpha \not\preccurlyeq \beta$. If $n_S = m_T$, then $\alpha \preccurlyeq \beta$ iff $\sum_{i > \max(S)} \alpha_i \preccurlyeq \sum_{i > \max(T)} \beta_i$. If a summation is empty, it is interpreted as zero.

To make $\prec$ a linear (non-partial) order, we restrict its domain to compact forms in normal form:

Definition The set of *normal compact forms* for ordinals less than ϵ_0 is the set of expressions inductively defined as follows:

1. 0 is a normal compact form.

2. If α is a normal compact form and $n > 0$, then $\omega^\alpha \cdot n$ is a compact normal form.

3′. Let α and β be normal compact forms, $\alpha = \omega^{\alpha'} \cdot n$ an ω-term and β be non-zero with $\omega^{\beta'} \cdot m$ the leading term in the additive expansion of β. If $\beta' \prec \alpha'$, then $\alpha + \beta$ is a normal compact form.

The predicates $\prec$ and $\preccurlyeq$ for compact forms can be seen to be polynomial time computable based on their inductive definitions, and the bounded arithmetic theory S^1_2 can Δ^b_1-define the syntactic concepts of $\prec$, $\preccurlyeq$ on compact forms. Similarly, S^1_2 can Δ^b_1-define the set of (Gödel numbers of) normal compact forms and the set of normal compact forms is polynomial time recognizable. Finally, S^1_2 can prove that $\prec$ is a total ordering on the normal compact forms, satisfying transitivity and trichotomy.

3.2 Provable well-foundedness on bounded domains

By Theorem 1, the well-foundedness of $\prec$ on bounded domains can be proved in T^2_2. Theorem 4 will show that the well-foundedness on bounded domains of (compact form) ordinal notations below ϵ_0 is provable in T^1_2. Before stating and proving that result, we show that T^1_2 is the weakest fragment of bounded arithmetic which can prove the well-foundedness of $\prec$ on ϵ_0.

Theorem 3 *Let $\prec$ be the ordering on normal basic (or, compact) forms for ordinals less than ϵ_0. Over the base theory S^1_2, $WF_\prec$ implies Σ^b_1-IND.*

Therefore, $S^1_2 \vdash WF_\prec$ implies $S^1_2 = T^1_2$. The latter condition is unlikely to hold, since it implies that $P^{NP}[\log]$ equals P^{NP} [10, Thm. 10.3.1].

Proof Since normal basic forms are essentially a special case of normal compact forms, it suffices to prove the theorem for normal basic forms. The proof is quite simple, we just need to give a natural, polynomial time computable, order preserving, embedding of the integers into the normal basic forms. (The identity mapping $n \mapsto \omega^0 + \omega^0 + \cdots + \omega^0$ from the integers into the normal basic forms cannot be used, since it has exponential growth rate and is not polynomial time.) The mapping we use is as follows: let $n \in \mathbb{N}$ have binary representation $(n_k \cdots n_1 n_0)_2$ with each $n_i \in \{0, 1\}$. Let $S = \{i : n_i = 1\}$ and $i_0, i_1, \ldots i_p$ enumerate S in decreasing order. Let α_i be the basic form representing the ordinal i, $\alpha_i = \omega^0 + \cdots + \omega^0$ (i summands). We define $\iota(0) = 0$ and, for $n > 0$,

$$\iota(n) = \omega^{\alpha_{i_0}} + \omega^{\alpha_{i_1}} + \cdots \omega^{\alpha_{i_p}}.$$

The mapping ι is readily seen to be polynomial time and Σ_1^b-definable in S_2^1. Furthermore, S_2^1 can prove that ι is order-preserving, so that $n < m$ iff $\iota(n) \prec \iota(m)$. S_2^1 can intensionally Δ_1^b-define the range of ι, and can Σ_1^b-define the inverse function ι^{-1}.

To complete the proof, suppose that $A(x)$ is a Σ_1^b-formula: we must show that S_2^1 can prove the least number principle (MIN) for A. Let $A^*(w)$ be the formula $w \in ran(\iota) \wedge A(\iota^{-1}(w))$. Then the formula WF_{A^*} immediately implies the least number principle for $A(x)$. □

The proof really showed that over the base theory S_2^1, $WF_{\prec'}$ implies Σ_1^b-IND, where $\prec'$ is the restriction of $\prec$ to normal basic (or, compact) forms $\alpha \prec \omega^\omega$.

Theorem 4 *Let $\prec$ be the ordering on normal basic (or, compact) forms. Then $T_2^1 \vdash WF_\prec$.*

It is sufficient to prove Theorem 4 for compact forms, since this subsumes the case of basic forms. Before giving the proof, we give an order-preserving embedding from bounded domains of normal compact forms into fixed-length sequences of integers. A compact form may be thought of as a sequence of symbols from the alphabet $\mathfrak{A}$ containing "ω", "+", the integers, and a special 'end of superscript' symbol "$\downarrow$". The length of an ordinal notation is defined to equal the number of symbols in the expression.

Definition The *length* of a compact form α is denoted $\mathrm{lh}(\alpha)$ and is defined by

- $\mathrm{lh}(0) = 1$.
- The length of $\omega^\alpha \cdot n$ equals $\mathrm{lh}(\alpha) + 3$.
- The length of $\alpha + \beta$ equals $\mathrm{lh}(\alpha) + \mathrm{lh}(\beta) + 1$.

We assume that an efficient Gödel encoding is used for compact forms; for instance, based on encoding the sequences $\mathrm{Seq}(\alpha)$ defined next. In particular, we require that $2 \cdot \mathrm{lh}(\alpha) \leq |\alpha|$ holds provably for S_2^1.

Definition $\mathfrak{A}$ is the set $\mathbb{N} \cup \{\omega, +, \downarrow\}$, and $\mathfrak{A}^*$ is the set of finite sequences over $\mathfrak{A}$. We use the infix operator $\frown$ to denote sequence concatenation. Let α be a compact form. Then $\mathrm{Seq}(\alpha)$ is the member of $\mathfrak{A}^*$ defined by:

- $\mathrm{Seq}(0)$ is $\langle 0 \rangle$. That is, the one element sequence containing the symbol 0.
- $\mathrm{Seq}(\omega^\alpha \cdot n)$ equals $\langle \omega \rangle \frown \mathrm{Seq}(\alpha) \frown \langle \downarrow, n \rangle$.
- $\mathrm{Seq}(\alpha + \beta)$ equals $\mathrm{Seq}(\alpha) \frown \langle + \rangle \frown \mathrm{Seq}(\beta)$.

Note that the symbols ω and $\downarrow$ will appear in $\mathrm{Seq}(\alpha)$ in pairs, and are balanced like left and right parentheses. Also, for every $\alpha \neq 0$, the two last elements of of $\mathrm{Seq}(\alpha)$ will be $\downarrow, n$ for some integer n.

Definition The set $\mathfrak{A}$ is linearly ordered by the convention that $\downarrow < + < \mathbb{N} < \omega$ with the integers inheriting their usual ordering. The induced lexicographic ordering on $\mathfrak{A}^*$ is also denoted $<$.

Lemma 5 *For any normal compact forms α and β, $\alpha \prec \beta$ holds if and only if* $\mathrm{Seq}(\alpha) < \mathrm{Seq}(\beta)$.

Proof The proof is by induction on the lengths of α and β. For either α or β equal to zero, this fact is immediate. Otherwise, α will equal either $\omega^{\alpha'} \cdot n$ or $\omega^{\alpha'} \cdot n + \alpha''$ and likewise β will equal either $\omega^{\beta'} \cdot m$ or $\omega^{\beta'} \cdot m + \beta''$. If $\alpha' \prec \beta'$, then the induction hypothesis implies that that $\mathrm{Seq}(\alpha') < \mathrm{Seq}(\beta')$ and therefore $\mathrm{Seq}(\alpha) < \mathrm{Seq}(\beta)$. Similarly, if $\beta' \prec \alpha'$, then $\mathrm{Seq}(\beta) < \mathrm{Seq}(\alpha)$.

So suppose $\alpha' = \beta'$. Then, if either $n < m$ or $m < n$, then $\mathrm{Seq}(\alpha) < \mathrm{Seq}(\beta)$ or $\mathrm{Seq}(\beta) < \mathrm{Seq}(\alpha)$, respectively. Finally, if $\alpha' = \beta'$ and $m = n$, then we apply the induction hypotheses to the forms α'' and β'' if they are both present. $\square$

Let $\mathbb{N}^*$ be the set of finite sequences over the integers under the usual lexicographic ordering. It is simple to give an order-preserving map from $\mathfrak{A}^*$ into $\mathbb{N}^*$. Namely, for $w \in \mathfrak{A}^*$, replace each element of the sequence w by a pair of integers: ω is replaced by $1, 0$; an integer n is replaced by $0, n+2$; the symbol $+$ is replaced with $0, 1$ and $\downarrow$ is replaced with $0, 0$. The resulting sequence has length twice as long as w. For α a compact form, we write $\mathrm{Nat}(\alpha)$ to denote the sequence in $\mathbb{N}^*$ which is the image of the sequence $\mathrm{Seq}(\alpha)$. The previous lemma immediately implies that, for all normal compact forms, $\mathrm{Nat}(\alpha) < \mathrm{Nat}(\beta)$ if and only if $\alpha \prec \beta$. We thus have:

Theorem 6 *The mapping* Nat *is an order-preserving mapping from the set of normal compact forms into $\mathbb{N}^*$. Furthermore, the length of the sequence* $\mathrm{Nat}(\alpha)$ *is* $2 \cdot \mathrm{lh}(\alpha)$.

The theorem is proved from the discussion above. The bound on the length is immediate from the definition. Furthermore, the definition of the function Nat and the proof of the theorem can be carried out in S^1_2.

We are now ready to prove Theorem 4. We argue informally in T^1_2 to prove $WF_A(y)$, for A a Σ^b_1-formula. Fix the value of y. Given an ordinal $\beta \leq y$, every integer in the sequence $\mathrm{Nat}(\beta)$ is less than $2^{|y|}$. Thus, for any $\beta \leq y$, we can encode $\mathrm{Nat}(\beta)$ by the single integer

$$n_\beta \ := \ \sum_{i=0}^{|y|} 2^{(|y|-i)|y|} (\mathrm{Nat}(\beta))_i,$$

where the notation $(-)_i$ extracts the i-th entry of sequence, starting with $i = 0$ for the first entry, and $(\mathrm{Nat}(\beta))_i$ is to equal 0 when i is greater than or equal the length of $\mathrm{Nat}(\beta)$. The mapping $\beta \mapsto n_\beta$, from normal compact forms to integers is still order-preserving, at least for ordinals β with $|\beta| \leq |y|$. Let $A^*(n)$ be the formula expressing

$$n = n_\beta \text{ for some } \beta \leq y \text{ such that } A(\beta).$$

By Σ_1^b-minimization (which is provable in T_2^1), there is a $<$-least n such that $A^*(n)$ holds. From $n = n_\beta$, there is a simple polynomial time procedure to obtain β. Therefore, T_2^1 proves that there is a $\prec$-least $\beta \leq y$ such that $A(\beta)$ holds.

That completes the proof of Theorem 4. □

The construction from the proof of Theorem 6 can also be applied directly to normal basic form ordinals. For basic form ordinals, Seq gives an order preserving map from the set of normal basic forms into finite sequences (words) over the alphabet $\{\downarrow, +, 0, 1, \omega\}$, where the sequences are ordered lexicographically. (In fact, the 1's may be omitted w.l.o.g.)

This is a theme we shall return to briefly in section 5.

4 Ordinals below Γ_0

4.1 Ordinal notations for Γ_0

Recall the binary Veblen function φ (cf. [6, 11]), which can be defined in the following way: Letting $\varphi_\alpha(\beta)$ denote $\varphi\alpha\beta$, then φ_0 is the enumeration of the "additiven Hauptzahlen" (i.e. $\varphi_0(\beta) = \omega^\beta$), and, for $\alpha > 0$, φ_α is the enumeration of the simultaneous fixed points of all $\varphi_{\alpha'}$ with $\alpha' < \alpha$, i.e. of $\{\beta : (\forall\alpha' < \alpha)\varphi_{\alpha'}(\beta) = \beta\}$. Now Γ_α is the αth element in the enumeration of $\{\gamma : \varphi\gamma 0 = \gamma\}$. Then each ordinal $\alpha < \Gamma_0$ either is 0 or can be written uniquely as

$$\alpha = \varphi\alpha_1\beta_1 + \ldots + \varphi\alpha_k\beta_k \tag{3}$$

with $\alpha_i, \beta_i < \varphi\alpha_i\beta_i$ and $\varphi\alpha_1\beta_1 \geq \ldots \geq \varphi\alpha_k\beta_k$. This is the basis for the well-known representation of ordinals less than Γ_0: namely, write an ordinal $\alpha < \Gamma_0$ in the form (3), recursively writing the subterms in the same form. It gives a syntactic representation of ordinals less than Γ_0. We need to formalize this syntactic representation in the bounded arithmetic theory S_2^1, by defining a set D which is the set of Gödel numbers of (syntactic representations of) ordinals less than Γ_0 and a binary formula $\prec$ which defines the ordinal ordering on the Gödel numbers. The formulas D and $\prec$ need to be Δ_1^b-formulas, that is to say, polynomial time computable, and S_2^1 needs to be able to prove that $\prec$ defines a total ordering on the domain D.

The syntactic representation in S_2^1 is developed in two stages: first representing ordinals in non-normal form, and then showing that ordinals can be converted to the normal form.

Definition The set of *basic forms* for ordinals less than Γ_0 is the set of expressions inductively defined as follows:

1. 0 is a basic form.
2. If α, β are basic forms, then so is $(\varphi\alpha\beta)$. We call the expression $(\varphi\alpha\beta)$ a *φ-term.*

3. If α and β are basic forms, and α is a φ-term, then $(\alpha+\beta)$ is a basic form.

We henceforth omit parentheses and write just $\varphi\alpha\beta$ for $(\varphi\alpha\beta)$ whenever there is no ambiguity.

As in the case of ϵ_0, S^1_2 can formalize the notion of basic form. It is immediate from the definition that every basic form α can be written uniquely in its *additive expansion*

$$\alpha \;=\; \alpha_1 + \alpha_2 + \cdots + \alpha_k, \tag{4}$$

where each α_i for $i < k$ is a φ-term, and α_k either is zero or a φ-term. Note that S^1_2 is able to prove the existence and uniqueness of the additive expansion of a basic form. The fact that we allow (only) the last term in the additive expansion (4) to equal 0 will be a technical convenience later when we define decompositions of normal forms. This does mean that we cannot directly combine two basic forms α and β using addition: instead we define $\alpha+\beta$ to be a basic term by defining: (a) $0+\beta$ is the same as β and (b) for $\alpha \neq 0$, $\alpha+\beta$ is defined by writing α in its additive expansion (4), letting $\ell = k$ if $\alpha_k \neq 0$ and $\ell = k-1$ if $\alpha_k = 0$, then letting $\alpha+\beta$ equal $\alpha_1 + \cdots + \alpha_\ell + \beta$.

To put basic forms into a normal form, we wish to further require that any additive expansion (4) has the α_i's in non-increasing order, and that all φ-terms $\varphi\alpha\beta$ satisfy $\alpha, \beta < \varphi\alpha\beta$. To formalize this, we first need to formalize a syntactic order, denoted $\prec$, on basic forms.

Definition Let α and β be basic forms. We inductively define $\alpha \preccurlyeq \beta$ and $\alpha \not\preccurlyeq \beta$, letting $\alpha \approx \beta$ mean that both $\alpha \preccurlyeq \beta$ and $\beta \preccurlyeq \alpha$, and $\alpha \prec \beta$ mean that $\alpha \preccurlyeq \beta$ and $\alpha \not\approx \beta$. First if $\alpha = 0$, then $\alpha \preccurlyeq \beta$. Also, if $\beta = 0$ and $\alpha \neq 0$, then $\alpha \not\preccurlyeq \beta$. Otherwise, let $\alpha_1 + \cdots + \alpha_k$ and $\beta_1 + \cdots + \beta_\ell$ be the additive expansions of α and β. Then

a. If $k = \ell = 1$, $\alpha_1 = \varphi\alpha'_1\alpha'_2$ and $\beta_1 = \varphi\beta'_1\beta'_2$, then $\alpha \preccurlyeq \beta$ is defined to hold if and only if either (1) $\alpha'_1 \prec \beta'_1$ and $\alpha'_2 \preccurlyeq \beta_1$; or (2) $\alpha'_1 \approx \beta'_1$ and $\alpha'_2 \preccurlyeq \beta'_2$; or (3) $\beta'_1 \prec \alpha'_1$ and $\alpha_1 \preccurlyeq \beta'_2$ holds.

b. If $\alpha_1 \prec \alpha_i$ for some $i > 1$, then $\alpha \preccurlyeq \beta$ if and only if $\alpha_2 + \ldots + \alpha_k \preccurlyeq \beta$.

c. If $\beta_1 \prec \beta_i$ for some $i > 1$, then $\alpha \preccurlyeq \beta$ if and only if $\alpha \preccurlyeq \beta_2 + \ldots + \beta_\ell$.

d. Otherwise, if none of a., b. or c. apply, then $\alpha \preccurlyeq \beta$ iff either (1) $\alpha_1 \prec \beta_1$ or (2) $\alpha_1 \approx \beta_1$ and $\alpha_2 + \cdots + \alpha_k \preccurlyeq \beta_2 + \cdots + \beta_\ell$. (If either k or ℓ are one, then the empty sum is interpreted as 0.)

The predicates $\prec$ and $\preccurlyeq$ for basic forms can be seen to be polynomial time computable based on their inductive definitions, and the bounded arithmetic theory S^1_2 can Δ^b_1-define the syntactic concepts of $\prec$, $\preccurlyeq$ on basic forms. It is easy to show that $\preccurlyeq$ is transitive and reflexive and satisfies dichotomy. Likewise, $\prec$ is transitive and antisymmetric. The proofs of these facts use ordinary integer induction and do not require transfinite induction and can be carried out in S^1_2. In fact, S^1_2 can prove the following facts which we need for later work:

Proposition 7 (S^1_2) *For all basic forms α, β, γ,*

1. $\alpha \preccurlyeq \beta \vee \beta \preccurlyeq \alpha$. *(Dichotomy)*
2. $\alpha \preccurlyeq \beta \wedge \beta \preccurlyeq \gamma \rightarrow \alpha \preccurlyeq \gamma$. *(Transitivity)*
3. $\alpha \prec \beta \wedge \beta \prec \gamma \rightarrow \alpha \prec \gamma$. *(Transitivity)*
4. $\alpha \preccurlyeq \alpha + \beta \wedge \beta \preccurlyeq \alpha + \beta$.
5. $0 \prec \varphi\alpha\beta$.
6. $\alpha \prec \beta \rightarrow \varphi\alpha(\varphi\beta\gamma) \approx \varphi\beta\gamma$.
7. $\beta \prec \gamma \leftrightarrow \varphi\alpha\beta \prec \varphi\alpha\gamma$.
8. $\alpha \prec \beta \leftrightarrow \varphi\alpha 0 \prec \varphi\beta 0$.
9. $\beta \preccurlyeq \varphi\alpha\beta$.
10. $\alpha \prec \varphi\alpha 0$. *Also,* $\alpha \prec \varphi\alpha\beta$.
11. $\beta \not\approx 0 \rightarrow 1 \preccurlyeq \beta$.
12. $\alpha \prec \beta \rightarrow \alpha + 1 \preccurlyeq \beta$.
13. $\varphi\beta\gamma \neq \varphi 00 \wedge \alpha \prec \varphi\beta\gamma \rightarrow \alpha + 1 \prec \varphi\beta\gamma$.
14. $\alpha \prec \beta \leftrightarrow \varphi\alpha(\gamma+1) \prec \varphi\beta(\gamma+1)$.

The expression "1" is an abbreviation for $\varphi 00$. We shall not carry out the proofs of the assertions in Proposition 7. The proofs use induction and should be carried out in the order the assertions are listed in the proposition.

The ordering $\prec$ is only a partial order on basic forms. To make $\prec$ a linear (non-partial) ordering, we need to mod out by the $\approx$ relation. The best way to do this is identify normal forms for ordinal notations. Our definition of a normal form will be slightly nonstandard for technical reasons. The difference is that additive expansions will always end with the term 0.

Definition The set of *normal forms*, also represented by NF, for ordinals less than Γ_0 is the set of expressions inductively defined as follows:

1. 0 is a normal form.
2. Let α, β and γ be normal forms, $\alpha, \beta \prec \varphi\alpha\beta$ and γ' be the leading term in the additive expansion of γ. If $\gamma' \preccurlyeq \varphi\alpha\beta$, then $\varphi\alpha\beta + \gamma$ is a normal form.

The formulation of (2) has the effect that any φ-term is part of a summation that ends with "+0". This is a technical convenience which has the advantage that any non-zero normal form β can be (uniquely) expressed in the form

$$\beta = \varphi\alpha_1(\varphi\alpha_2(\ldots(\varphi\alpha_{b-1}(\varphi\alpha_b 0 + \gamma_b) + \gamma_{b-1})\ldots) + \gamma_2) + \gamma_1$$

where of course some γ_i's may be 0.

S^1_2 can Δ^b_1-define the set of (Gödel numbers of) normal forms and the set of normal forms is polynomial time recognizable. Further, S^1_2 can prove that $\prec$ is a total ordering on the normal forms, satisfying transitivity and trichotomy, and that the following properties hold:

Proposition 8 (S_2^1) *Let "1" abbreviate the normal form $\varphi 00 + 0$.*

1. *$\forall \alpha, \beta \in NF$, if $\alpha \approx \beta$, then $\alpha = \beta$.*
2. *$\prec$ is a total order on normal forms (satisfies trichotomy and transitivity).*
3. *$\forall \alpha, \beta \in NF$, $\alpha + 1$ and $\varphi\alpha 0 + 0$ and $\varphi\alpha(\beta + 1) + 0$ are in NF.*
4. *If $\alpha_1 + \cdots + \alpha_k \in NF$, then $\alpha_1 \not\prec \alpha_i$ for all i.*
5. *If $\alpha = \beta + \gamma \in NF$ and $\beta \prec \delta$ and β and δ are φ-terms, then $\alpha \prec \delta$.*
6. *For all basic forms α, there is a $\beta \in NF$ such that $\alpha \approx \beta$.*
7. *If $\alpha \in NF$ and β is a proper subterm of α, then $\beta \prec \alpha$.*

From items 1 and 6, every basic form corresponds to a unique normal form. The proof of the proposition is carried out similarly to the proof of the previous proposition. In order to prove 1, one first proves that

$$\beta \prec \varphi\alpha\beta \wedge \delta \prec \varphi\gamma\delta \wedge \varphi\alpha\beta \approx \varphi\gamma\delta \rightarrow \alpha \approx \gamma \wedge \beta \approx \delta.$$

4.2 Decompositions

By Theorem 1 we know that the well-foundedness of $\prec$ on bounded domains can be proved in T_2^2. We will prove that the well-foundedness on bounded domains of ordinal notations below Γ_0 is provable in T_2^1. We do this by describing a $\square_1^p$-algorithm that computes an order-preserving embedding of Γ_0 into words, i.e. finite sequences, over some finite alphabet together with the lexicographic ordering. This embedding will map bounded domain ordinals to fixed length words (i.e., the length will be bounded polynomially in the binary length of the bound to the ordinal domain). Using the same technique as in Section 3 we can code fixed length words in an order-preserving fashion by natural numbers. This algorithm will be formalizable in S_2^1, and S_2^1 will prove that it is order-preserving. Hence the task of finding a $\prec$-minimal β satisfying a Σ_1^b-formula $A(\beta)$ can be reduced to finding a $<$-minimal n satisfying some Σ_1^b-formula $A^*(n)$, which can be solved in T_2^1.

Our idea for computing the order-preserving embedding is to find certain maximal elements in the decomposition of ordinals. E.g., assume β is a normal form. As mentioned above, β has the form

$$\beta = \varphi\alpha_1(\varphi\alpha_2(\ldots(\varphi\alpha_b 0 + \gamma_b)\ldots) + \gamma_2) + \gamma_1. \tag{5}$$

Let α^* be the maximal element of $\alpha_1, \ldots, \alpha_b$ with respect to $\prec$. Lemma 9 will show that

$$\varphi\alpha^* 0 \preccurlyeq \beta \prec \varphi(\alpha^* + 1)0,$$

hence α^* is the most important subterm β concerning $\prec$, which will come (lexicographically) before others. We will recursively apply our algorithm to α^*, which is simpler, because the length of α^* is smaller than that of β. Then the algorithm will make recursive calls to compute the other components of β, namely, the parts of β which are not in the subterm α^*.

In order to formalize this idea we start by defining the decomposition of non-zero normal forms in Γ_0 as shown in equation (5).

Definition (Decomposition of β) Each normal form $\beta \in \Gamma_0 \setminus \{0\}$ can uniquely be decomposed in the following way: Write $\beta_0 := \beta$. Let $\beta_i \neq 0$ be recursively defined, then we define α_{i+1}, β_{i+1} and γ_{i+1} by $\beta_i = \varphi\alpha_{i+1}\beta_{i+1} + \gamma_{i+1}$. Let b be the index of the last β_i, i.e. $\beta_b = 0$. This expresses β as in equation (5). Then we define

$$\begin{aligned} \alpha^* &:= \mathrm{m}_1(\beta) := \max\nolimits_{\prec}\{\alpha_1, \ldots, \alpha_b\} \\ i^* &:= \min\{i \ :\ \alpha_i = \alpha^*\} \\ \beta^* &:= \mathrm{m}_2(\beta) := \beta_{i^*} \end{aligned}$$

We call $b; \vec{\alpha}; \vec{\beta}; \vec{\gamma}; \alpha^*; \beta^*; i^*$ the *decomposition* of β. As β is non-zero and a normal form we immediately have $b > 0$, $i^* > 0$ and that $\vec{\alpha}$, $\vec{\beta}$, $\vec{\gamma}$ are all normal forms.

Example: Let $\beta = \varphi 2(\varphi 10 + 1) + 0$. The natural numbers occurring in this term stand for the normal forms representing that number, e.g. 1 abbreviates the normal form $\varphi 00 + 0$. Then we decompose:

$$\begin{array}{lll} & \beta_0 = \varphi 2(\varphi 10 + 1) + 0 & \\ \alpha_1 = 2 & \beta_1 = \varphi 10 + 1 & \gamma_1 = 0 \\ \alpha_2 = 1 & \beta_2 = 0 & \gamma_2 = 1 \end{array}$$

Hence $b = 2$, $\alpha^* = 2$, $i^* = 1$, $\beta^* = \varphi 10 + 1$.

In section 4.3, we present an algorithm for computing an order preserving embedding of ordinals below Γ_0 into words over a finite alphabet. The intuitive idea behind the embedding algorithm is that it first computes the components α^* and β^* of the decomposition of β, and recursively outputs the embedding of α^* and β^*. The algorithm then treats $\rho = \varphi\alpha^*\beta^*$ as a fixed subterm which has been fully processed, and continues computing the embedding for the rest of the ordinal β recursively. This recursive computation above the fixed subterm ρ works with ordinals β which have decomposition containing a β_i equal to $\rho + \gamma$ for some γ. We call the set of ordinals β satisfying this condition a *context*. To formally define contexts, we define classes $\tilde{\Gamma}_0[\rho]$ and $\Gamma_0[\rho]$ as follows:

Definition The variables $\alpha, \beta, \gamma, \delta$ range over normal forms. The class $\tilde{\Gamma}_0[\rho]$ is inductively defined by:

1. Let $\gamma \neq 0$. If $\varphi\alpha(\rho + \gamma) + \delta$ is in normal form, then it is in $\tilde{\Gamma}_0[\rho]$.
2. Let $\beta \in \tilde{\Gamma}_0[\rho]$. If $\varphi\alpha\beta + \gamma$ is in normal form, then it is in $\tilde{\Gamma}_0[\rho]$.

$\Gamma_0[\rho]$ contains $\tilde{\Gamma}_0[\rho]$ plus all normal form ordinals $\rho + \gamma$.

Before giving a detailed description of the embedding algorithm we establish various properties of decompositions and contexts needed to prove the algorithm's correctness.

Lemma 9 (S^1_2) *Let $\beta \in NF \setminus \{0\}$ and α^*, β^* be as above.*

1. $\varphi\alpha^*0 \preceq \beta \prec \varphi(\alpha^*+1)0$.

2. $\varphi\alpha^*\beta^* \preceq \beta \prec \varphi\alpha^*(\beta^*+1)$.

3. $\beta \in \Gamma_0[\varphi\alpha^*\beta^*]$. In particular, if $i^* > 1$, then $\gamma_{i^*} \neq 0$.

Proof Since $\beta \succcurlyeq \beta_{i^*-1} \succcurlyeq \varphi\alpha^*\beta^* \succcurlyeq \varphi\alpha^*0$, we have the first inequalities in 1. and 2. To prove the second inequality in 1., we show $\beta_j \prec \varphi(\alpha^*+1)0$, for $j \leq b$, by induction on j from b down to 0. In the base case, $j = b$, and $\beta_b = 0$ so the assertion is obvious. For the induction step, let $j < b$. The induction hypothesis implies

$$\varphi\alpha_{j+1}\beta_{j+1} \prec \varphi\alpha_{j+1}(\varphi(\alpha^*+1)0) \approx \varphi(\alpha^*+1)0$$

since $\alpha_{j+1} \preccurlyeq \alpha^*$ by the definition of α^*. Hence $\beta_j = \varphi\alpha_{j+1}\beta_{j+1} + \gamma_{j+1} \prec \varphi(\alpha^*+1)0$.

The second inequality in 2. is proved by a similar use of induction on j. In the base case, $j = i^* - 1$; then $\beta_j = \varphi\alpha^*\beta^* + \gamma_{i^*} \prec \varphi\alpha^*(\beta^*+1)$, since $\varphi\alpha^*\beta^* \prec \varphi\alpha^*(\beta^*+1)$. For the induction step, let $j < i^* - 1$. By the induction hypothesis we have

$$\varphi\alpha_{j+1}\beta_{j+1} \prec \varphi\alpha_{j+1}(\varphi\alpha^*(\beta^*+1)) \approx \varphi\alpha^*(\beta^*+1)$$

since $\alpha_{j+1} \prec \alpha^*$ by the definition of α^*. Therefore $\beta_j = \varphi\alpha_{j+1}\beta_{j+1} + \gamma_{j+1} \prec \varphi\alpha^*(\beta^*+1)$.

The condition 3. is nearly obvious. The only thing to check is that γ_{i^*} is non-zero if β does not have the form $\varphi\alpha^*\beta^* + \gamma_{i^*}$. But in this case $i^* > 1$, hence we have $\beta_{i^*-2} = \varphi\alpha_{i^*-1}(\varphi\alpha^*\beta^* + \gamma_{i^*}) + \gamma_{i^*-1}$. By definition of α^* we have $\alpha_{i^*-1} \prec \alpha^*$. If γ_{i^*} were zero, then we would have $\varphi\alpha_{i^*-1}(\varphi\alpha^*\beta^* + \gamma_{i^*}) \approx \varphi\alpha^*\beta^* + \gamma_{i^*}$ contradicting β_{i^*-2} being a normal form. ◻

We next generalize the notion of decomposition to decomposition of ordinals β in $\tilde{\Gamma}_0[\rho]$. This is called *decomposition above* ρ and the main difference in the process of forming the decomposition of β above ρ is that the process stops when the subterm ρ is reached. That is to say, the decomposition of β above ρ will be an expression of the form

$$\beta = \varphi\alpha_1(\varphi\alpha_2(\ldots(\varphi\alpha_b(\rho + \gamma_{b+1}) + \gamma_b)\ldots) + \gamma_2) + \gamma_1. \tag{6}$$

We will then define $\alpha^{*,\rho}$ to equal the $\prec$-maximal element of $\alpha_1, \ldots, \alpha_b$ and this will satisfy

$$\varphi\alpha^{*,\rho}(\rho+1) \preccurlyeq \beta \prec \varphi(\alpha^{*,\rho}+1)(\rho+1);$$

that is to say, $\alpha^{*,\rho}$ is the most important subterm of β above ρ (see Lemmas 10 and 13).

Definition (Decomposition of β above ρ) Let ρ be a φ-term. Each $\beta \in \tilde{\Gamma}_0[\rho]$ can uniquely be decomposed in the following way. Let $\beta_0 := \beta$. Inductively define α_{i+1}, β_{i+1} and γ_{i+1} by $\beta_i = \varphi\alpha_{i+1}\beta_{i+1} + \gamma_{i+1}$. Let b be the index such that β_b is of the form $\rho + \gamma_{b+1}$. This expresses β in the form of equation (6).

Then we define

$$\begin{aligned} \alpha^{*,\rho} &:= \mathrm{m}_1^\rho(\beta) := \max\nolimits_\prec\{\alpha_1, \dots, \alpha_b\} \\ i^{*,\rho} &:= \min\{i \,:\, \alpha_i = \alpha^{*,\rho}\} \\ \beta^{*,\rho} &:= \mathrm{m}_2^\rho(\beta) := \beta_{i^{*,\rho}} \end{aligned}$$

We say that $b; \vec{\alpha}; \vec{\beta}; \vec{\gamma}; \alpha^{*,\rho}; \beta^{*,\rho}; i^{*,\rho}$ is the *decomposition of* β *above* ρ. As β is a normal form in $\tilde{\Gamma}_0[\rho]$ we immediately have $b > 0$, $i^* > 0$, $\gamma_{b+1} \neq 0$ and that all $\vec{\alpha}$, $\vec{\beta}$, $\vec{\gamma}$ are normal forms.

Example: Let $\beta = \varphi 2(\varphi 1(\rho + 2) + 1) + 0$. Then we decompose:

$$\begin{array}{lll} & \beta_0 = \varphi 2(\varphi 1(\rho + 2) + 1) + 0 & \\ \alpha_1 = 2 & \beta_1 = \varphi 1(\rho + 2) + 1 & \gamma_1 = 0 \\ \alpha_2 = 1 & \beta_2 = \rho + 2 & \gamma_2 = 1 \\ & & \gamma_3 = 2 \end{array}$$

Hence $b = 2$, $\alpha^{*,\rho} = 2$, $i^{*,\rho} = 1$, $\beta^{*,\rho} = \varphi 1(\rho + 2) + 1$.

At first glance, one might think that the earlier defined decomposition is the same as a decomposition above 0. However, this is not true, because $\beta \in \tilde{\Gamma}_0[\rho]$ includes the important information that the additive component next to ρ is not zero. Therefore, we would not get $\beta_b = 0$ if we allowed $\rho = 0$.

The next lemma gives some properties of the components $\alpha^{*,\rho}$ and $\beta^{*,\rho}$ of $\beta \in \tilde{\Gamma}_0[\rho]$ which are needed for proving that our embedding algorithm produces an order-preserving embedding.

Lemma 10 (S^1_2) *Let $\beta \in \tilde{\Gamma}_0[\rho]$ for some φ-term ρ, and $\alpha^{*,\rho}$, $\beta^{*,\rho}$ be as above.*

1. *$\varphi\alpha^{*,\rho}(\rho + 1) \preceq \beta \prec \varphi(\alpha^{*,\rho} + 1)(\rho + 1)$.*
2. *$\varphi\alpha^{*,\rho}\beta^{*,\rho} \preceq \beta \prec \varphi\alpha^{*,\rho}(\beta^{*,\rho} + 1)$.*
3. *$\beta^{*,\rho} \in \Gamma_0[\rho] \setminus \{\rho + 0\}$ and $\beta \in \Gamma_0[\varphi\alpha^{*,\rho}\beta^{*,\rho}]$.*

Proof By $\beta \succcurlyeq \beta_{i^{*,\rho}-1} \succcurlyeq \varphi\alpha^{*,\rho}\beta^{*,\rho} \succcurlyeq \varphi\alpha^{*,\rho}(\rho + \gamma_{b+1}) \succcurlyeq \varphi\alpha^{*,\rho}(\rho + 1)$, we have the first inequalities in 1., respectively, 2. To prove the second inequality in 1., we show $\beta_j \prec \varphi(\alpha^{*,\rho} + 1)(\rho + 1)$ for $j \leq b$ by induction on $j = b, \dots, 0$. In the base case, $j = b$, and we have $\beta_b = \rho + \gamma_{b+1} \prec \varphi(\alpha^{*,\rho} + 1)(\rho + 1)$. The induction step, $j < b$, is similar to the analogous case in the proof of Lemma 9.

The second inequality in 2. is proved similarly. For 3., $\beta \in \Gamma_0[\varphi\alpha^{*,\rho}\beta^{*,\rho}]$ is proven in the same way as part 3. of Lemma 9. The other assertion is obvious. $\square$

We have described contexts $\tilde{\Gamma}_0[\rho] \subset \Gamma_0[\rho]$. We will need some properties which allow us to decide whether an element of $\Gamma_0[\rho]$ is in $\tilde{\Gamma}_0[\rho]$ or not. The next lemma will provide us with this.

Lemma 11 (S^1_2) *Let $\beta, \eta \in \Gamma_0[\rho]$ for some φ-term ρ.*

1. $\beta \in \tilde{\Gamma}_0[\rho] \quad \Leftrightarrow \quad \beta \succcurlyeq \varphi 0(\rho+1)$.
2. $\beta \in \tilde{\Gamma}_0[\rho],\ \beta \preccurlyeq \eta \quad \Rightarrow \quad \eta \in \tilde{\Gamma}_0[\rho]$.

Proof 1.: Assume $\beta \in \tilde{\Gamma}_0[\rho]$. Let $b; \vec{\alpha}; \vec{\beta}; \vec{\gamma}; \alpha^{*,\rho}; \beta^{*,\rho}; i^{*,\rho}$ be the ρ-decomposition of β. Then $\beta \succcurlyeq \varphi\alpha_b(\rho + \gamma_{b+1}) \succcurlyeq \varphi 0(\rho+1)$. To prove the converse, assume $\beta \in \Gamma_0[\rho] \setminus \tilde{\Gamma}_0[\rho]$. Then $\beta = \rho + \gamma_1 \prec \varphi 0(\rho+1)$.

2.: Assume $\eta \notin \tilde{\Gamma}_0[\rho]$ and $\beta \in \tilde{\Gamma}_0[\rho]$. Using 1. twice we obtain $\eta \prec \varphi 0(\rho+1) \preccurlyeq \beta$. □

The next lemma states that the main components from decompositions respect the ordering of ordinals.

Lemma 12 (S^1_2) *Let β, η be normal forms with $0 \prec \beta \preccurlyeq \eta$.*

1. $\mathrm{m}_1(\beta) \preccurlyeq \mathrm{m}_1(\eta)$.
2. $\mathrm{m}_1(\beta) = \mathrm{m}_1(\eta) \Rightarrow \mathrm{m}_2(\beta) \preccurlyeq \mathrm{m}_2(\eta)$.

Proof With $\beta, \eta \in \mathrm{NF} \setminus \{0\}$ we associate decompositions $b; \vec{\alpha}; \vec{\beta}; \vec{\gamma}; \alpha^*; \beta^*; i^*$ and $n; \vec{\xi}; \vec{\eta}; \vec{\mu}; \xi^*; \eta^*; j^*$, respectively. Then the lemma states:

1. $\alpha^* \preccurlyeq \xi^*$.
2. $\alpha^* = \xi^* \Rightarrow \beta^* \preccurlyeq \eta^*$.

We prove these by induction on $b+n$. Since $\beta \preccurlyeq \eta$, we obtain $\varphi\alpha_1\beta_1 \preccurlyeq \varphi\xi_1\eta_1$. If $\varphi\alpha_1\beta_1 \approx \varphi\xi_1\eta_1$, then $\varphi\alpha_1\beta_1 = \varphi\xi_1\eta_1$ as $\beta, \eta \in \mathrm{NF}$, thus the assertions are obvious. Hence we assume $\varphi\alpha_1\beta_1 \prec \varphi\xi_1\eta_1$. The proofs of 1. and 2. split into three cases (I)-(III) depending on the reason that $\varphi\alpha_1\beta_1 \prec \varphi\xi_1\eta_1$ holds.

(I) Suppose $\alpha_1 \prec \xi_1$ and $\beta_1 \prec \varphi\xi_1\eta_1$.

1.: If $i^* = 1$ we have $\alpha^* = \alpha_1 \prec \xi_1 \preccurlyeq \xi^*$. Otherwise $i^* > 1$ and $\beta_1 \neq 0$, hence

$$\alpha^* = \mathrm{m}_1(\beta_1) \overset{(*)}{\preccurlyeq} \mathrm{m}_1(\varphi\xi_1\eta_1) = \xi^*$$

using induction hypothesis at $(*)$.

2.: Suppose $\alpha^* = \xi^*$. Then $i^* > 1$, since otherwise $\alpha^* = \alpha_1 \prec \xi_1 \preccurlyeq \xi^* = \alpha^*$. Therefore, $\beta_1 \neq 0$, $\alpha^* = \mathrm{m}_1(\beta_1)$, $\beta^* = \mathrm{m}_2(\beta_1)$, hence $\mathrm{m}_1(\beta_1) = \alpha^* = \xi^* = \mathrm{m}_1(\varphi\xi_1\eta_1)$, thus the induction hypothesis shows

$$\beta^* = \mathrm{m}_2(\beta_1) \preccurlyeq \mathrm{m}_2(\varphi\xi_1\eta_1) = \eta^*.$$

(II) Suppose $\alpha_1 = \xi_1$ and $\beta_1 \prec \eta_1$.

1.: If $i^* = 1$ we have $\alpha^* = \alpha_1 = \xi_1 \preccurlyeq \xi^*$. If $i^* > 1$, then $\beta_1 \neq 0$ and hence $\eta_1 \neq 0$. Hence by the induction hypothesis

$$\alpha^* = \mathrm{m}_1(\beta_1) \preccurlyeq \mathrm{m}_1(\eta_1) \preccurlyeq \xi^*.$$

2.: Suppose $\alpha^* = \xi^*$. If $i^* = 1$ we obtain $\xi_1 = \alpha_1 = \alpha^* = \xi^*$, hence $j^* = 1$, too. Thus

$$\beta^* = \beta_1 \prec \eta_1 = \eta^*.$$

Otherwise $i^*, j^* > 1$, and again $\beta_1 \neq 0$ and $\eta_1 \neq 0$. By the induction hypothesis we first obtain

$$\alpha^* = \mathrm{m}_1(\beta_1) \preccurlyeq \mathrm{m}_1(\eta_1) = \xi^* = \alpha^*,$$

hence $\mathrm{m}_1(\beta_1) = \mathrm{m}_1(\eta_1)$, and applying the induction hypothesis again yields

$$\beta^* = \mathrm{m}_2(\beta_1) \preccurlyeq \mathrm{m}_2(\eta_1) = \eta^*.$$

(III) Suppose $\alpha_1 \succ \xi_1$ and $\varphi\alpha_1\beta_1 \prec \eta_1$. So $\eta_1 \neq 0$.

1.: By induction hypothesis we obtain

$$\alpha^* = \mathrm{m}_1(\varphi\alpha_1\beta_1) \preccurlyeq \mathrm{m}_1(\eta_1) \preccurlyeq \xi^*.$$

2.: Suppose $\alpha^* = \xi^*$, then $j^* > 1$, because otherwise $\xi^* = \xi_1 \prec \alpha_1 \preccurlyeq \alpha^* = \xi^*$. Therefore, $\xi^* = \mathrm{m}_1(\eta_1)$, $\eta^* = \mathrm{m}_2(\eta_1)$, hence $\mathrm{m}_1(\varphi\alpha_1\beta_1) = \alpha^* = \xi^* = \mathrm{m}_1(\eta_1)$. Thus the induction hypothesis shows

$$\beta^* = \mathrm{m}_2(\varphi\alpha_1\beta_1) \preccurlyeq \mathrm{m}_2(\eta_1) = \eta^*.$$

□

Similar properties are needed for decompositions above ρ.

Lemma 13 (S^1_2) *Let $\beta, \eta \in \tilde{\Gamma}_0[\rho]$ with $\beta \preccurlyeq \eta$.*

1. $\mathrm{m}^\rho_1(\beta) \preccurlyeq \mathrm{m}^\rho_1(\eta)$

2. $\mathrm{m}^\rho_1(\beta) = \mathrm{m}^\rho_1(\eta) \Rightarrow \mathrm{m}^\rho_2(\beta) \preccurlyeq \mathrm{m}^\rho_2(\eta)$.

Proof The proof of these assertions is simply a translation of the proof of Lemma 12 using the following Translation:

$$\begin{array}{rcl} \text{decomposition} & \rightsquigarrow & \text{decomposition above } \rho \\ {}^* & \rightsquigarrow & {}^{*,\rho} \\ \cdots \neq 0 & \rightsquigarrow & \cdots \in \tilde{\Gamma}_0[\rho] \end{array}$$

The only additional thing we have to ensure is that the induction hypothesis is always applicable; that is, that the terms under consideration are in $\tilde{\Gamma}_0[\rho]$. □

4.3 Algorithms

In this section, we formulate precisely the embedding algorithms and show that they are in $\square^p_1$.

Definition The *length* of a basic form α is denoted $\mathrm{lh}(\alpha)$ and is defined by

- $\mathrm{lh}(0) = 1$.
- The length of $\varphi\alpha\beta$ equals $\mathrm{lh}(\alpha) + \mathrm{lh}(\beta) + 1$.
- The length of $\alpha + \beta$ equals $\mathrm{lh}(\alpha) + \mathrm{lh}(\beta) + 1$.

Algorithms: The algorithms are given by simultaneous recursion.

1. Algorithm `dec1`(β). Let β be a normal form.

`Query:`	$\beta = 0$?
`Yes:`	`Output` $\mathfrak{a} = \mathbf{0}$
`No:`	`Compute` α^*`,` β^* `from the decomposition of` β
	$\mathfrak{b} =$ `dec1`(α^*)
	$\mathfrak{c} =$ `dec1`(β^*)
	$\mathfrak{d} =$ `dec2`$(\varphi\alpha^*\beta^*, \beta)$
	`Output` $\mathfrak{a} = (\mathfrak{b}\mathfrak{c}\mathfrak{d})$

2. Algorithm `dec2`(ρ, β). Let ρ be a φ-term and $\beta \in \Gamma_0[\rho]$.

`Query:`	`Does` β `have the form` $\rho + \gamma$?
`Yes:`	$\mathfrak{b} =$ `dec1`(γ)
	`Output` $\mathfrak{a} = (*\mathfrak{b})$
`No:`	`Compute` $\alpha^{*,\rho}$`,` $\beta^{*,\rho}$ `from the decomposition of` β `above` ρ
	$\mathfrak{b} =$ `dec1`$(\alpha^{*,\rho})$
	$\mathfrak{c} =$ `dec2`$(\rho, \beta^{*,\rho})$
	$\mathfrak{d} =$ `dec2`$(\varphi\alpha^{*,\rho}\beta^{*,\rho}, \beta)$
	`Output` $\mathfrak{a} = (\mathfrak{b}\mathfrak{c}\mathfrak{d})$

Before proving that the algorithms are order-preserving we first show that they indeed are $\square^p_1$-algorithms. In order to show that the runtime of these algorithms is polynomially bounded, we first need to show that the number of recursive calls is small. The next lemma implies that this number is always bounded by $|\beta|$.

Lemma 14 (S^1_2) *Let β be in normal form, then the number of recursive calls in algorithm* `dec1`(β) *is* $\mathrm{lh}(\beta) - 1$. *If ρ is a φ-term and $\beta \in \Gamma_0[\rho]$, then the number of recursive calls in algorithm* `dec2`(ρ, β) *is* $\mathrm{lh}(\beta) - \mathrm{lh}(\rho) - 1$.

Proof Let nrc be the total number of (nested) recursive calls a routine or a subroutine needs until it finishes. We show by simultaneous length-induction on k that for normal forms β

$$\mathrm{lh}(\beta) \leq k \quad \Rightarrow \quad nrc(\texttt{dec1}(\beta)) = \mathrm{lh}(\beta) - 1, \tag{7}$$

and that for φ-terms ρ and $\beta \in \Gamma_0[\rho]$

$$\mathrm{lh}(\beta) - \mathrm{lh}(\rho) \leq k \quad \Rightarrow \quad nrc(\texttt{dec2}(\rho, \beta)) = \mathrm{lh}(\beta) - \mathrm{lh}(\rho) - 1. \tag{8}$$

Case `dec1`(β). We study the behaviour of the algorithm. If the answer to `Query` is `Yes`, then $\beta = 0$, hence $nrc = 0$ and $\mathrm{lh}(\beta) = 1$. Thus the assertion follows.

Otherwise the answer to `Query` is `No`. We compute

$$\begin{aligned} nrc &= nrc(\texttt{dec1}(\alpha^*)) + nrc(\texttt{dec1}(\beta^*)) + nrc(\texttt{dec2}(\varphi\alpha^*\beta^*, \beta)) + 3 \\ &= (\mathrm{lh}(\alpha^*) - 1) + (\mathrm{lh}(\beta^*) - 1) + (\mathrm{lh}(\beta) - \mathrm{lh}(\varphi\alpha^*\beta^*) - 1) + 3 \\ &= \mathrm{lh}(\beta) - 1. \end{aligned}$$

Case $\texttt{dec2}(\rho, \beta)$. If the answer to `Query` is `Yes`, then by the induction hypothesis we obtain $nrc = nrc(\texttt{dec1}(\gamma)) + 1 = \mathrm{lh}(\gamma) = \mathrm{lh}(\beta) - \mathrm{lh}(\rho) - 1$ as $\beta = \rho + \gamma$.

Otherwise the answer to `Query` is `No`. We compute

$$\begin{aligned} nrc &= nrc(\texttt{dec1}(\alpha^{*,\rho})) + nrc(\texttt{dec2}(\rho, \beta^{*,\rho})) + nrc(\texttt{dec2}(\varphi\alpha^{*,\rho}\beta^{*,\rho}, \beta)) + 3 \\ &= (\mathrm{lh}(\alpha^{*,\rho}) - 1) + (\mathrm{lh}(\beta^{*,\rho}) - \mathrm{lh}(\rho) - 1) + (\mathrm{lh}(\beta) - \mathrm{lh}(\varphi\alpha^{*,\rho}\beta^{*,\rho}) - 1) + 3 \\ &= \mathrm{lh}(\beta) - \mathrm{lh}(\rho) - 1. \end{aligned}$$

As phrased above the induction is on a Π^b_2-formula. To formalize the proof in S^1_2, we fix a particular β_0 and prove the lemma holds for β_0. For this, it is enough to prove that (7) and (8) hold for all β and ρ which are subterms of β_0. Quantifying over all subtems requires only a sharply bounded quantifier, so only Σ^b_1-PIND is needed.

This finishes the proof of Lemma 14. □

Using the previous lemma it is easy to obtain a polynomial runtime bound on the algorithms `dec1` and `dec2`. Thus we have established:

Theorem 15 (S^1_2) *`dec1` and `dec2` are $\Box^p_1$-algorithms.*

In order to argue that the algorithms are order-preserving we first have to fix an ordering on the outputs. Let $\mathcal{A}$ be the alphabet $\{\mathbf{0}, \mathbf{(}, \mathbf{)}, \mathbf{*}\}$. The set of finite words over $\mathcal{A}$ is denoted by $\mathcal{A}^*$. Let $\mathfrak{G}$ be the grammar over $\mathcal{A}$ defined by

$$\mathfrak{G} \quad := \quad \mathbf{0} \mid \mathbf{(}\mathfrak{G}\mathfrak{G}\mathfrak{G}\mathbf{)} \mid \mathbf{(*}\mathfrak{G}\mathbf{)}.$$

It is obvious that the algorithms output words from $\mathfrak{G}$. Furthermore, if β is nonzero then $\texttt{dec1}(\beta)$ starts with '**(**', and if ρ is a φ-term and $\beta \in \Gamma_0[\rho]$ then $\texttt{dec2}(\rho, \beta)$ also starts with '**(**'. Words over $\mathcal{A}$ are ordered lexicographically by fixing an ordering $<$ on the four symbols of $\mathcal{A}$:

$$\mathbf{*} \; < \; \mathbf{0} \; < \; \mathbf{(} \; < \; \mathbf{)}.$$

We use $<^l$ to denote the induced lexicographical ordering on $\mathcal{A}^*$.

Lemma 16 (S^1_2) *Let $u, v \in \mathfrak{G}$, then $u <^l v$ implies $ux <^l vy$ for all $x, y \in \mathcal{A}^*$.*

Proof If this is not the case, then u must be a proper initial subword of v. It is not hard to conclude that then not both $u, v \in \mathfrak{G}$ can hold. To this end let $nlp(u)$ ($nrp(u)$) be the number of left (right) parenthesis in u. If $u \in \mathfrak{G}$ then $nlp(u) = nrp(u)$. If $u \in \mathfrak{G}$ and v is a proper initial subword of u different from the empty word, then $nlp(v) > nrp(v)$, which can easily be seen by induction on the number of symbols in u. Hence, if $u, v \in \mathfrak{G}$ then v cannot be a proper initial subword of u. □

We now show that S^1_2 can prove that the algorithms compute order-preserving maps.

Theorem 17 *(S^1_2)* `dec1` *and* `dec2` *are order-preserving.*

In order to show this we will prove

1. Let $\beta, \eta \in \mathrm{NF}$, then

$$\beta \prec \eta \quad \Leftrightarrow \quad \mathtt{dec1}(\beta) <^l \mathtt{dec1}(\eta).$$

2. Let ρ be a φ-term and $\beta, \eta \in \Gamma_0[\rho]$, then

$$\beta \prec \eta \quad \Leftrightarrow \quad \mathtt{dec2}(\rho, \beta) <^l \mathtt{dec2}(\rho, \eta).$$

In both cases it is enough to show "$\Rightarrow$", e.g. because then

$$\beta \not\prec \eta \Rightarrow \eta \preccurlyeq \beta \Rightarrow \mathtt{dec1}(\eta) \leq^l \mathtt{dec1}(\beta) \Rightarrow \mathtt{dec1}(\beta) \not<^l \mathtt{dec1}(\eta)$$

We show 1. and 2. simultaneously by induction on $\mathrm{lh}(\beta)$, respectively, $\mathrm{lh}(\beta) - \mathrm{lh}(\rho)$. More exactly, as in the proof of Lemma 14, we show for all β, η, ρ occuring as subterms of some fixed β_0, that if $\beta \in \mathrm{NF}$ then

$$\mathrm{lh}(\beta) \leq k \wedge \beta \prec \eta \Rightarrow \mathtt{dec1}(\beta) <^l \mathtt{dec1}(\eta),$$

and if ρ is a φ-term and $\beta, \eta \in \Gamma_0[\rho]$ then

$$\mathrm{lh}(\beta) - \mathrm{lh}(\rho) \leq k \wedge \beta \prec \eta \Rightarrow \mathtt{dec2}(\rho, \beta) <^l \mathtt{dec2}(\rho, \eta)$$

by length-induction on k.

For 1., assume $\beta \prec \eta$. If $\beta = 0$ then

$$\mathtt{dec1}(\beta) = \mathbf{0} <^l \boldsymbol{(} <^l \mathtt{dec1}(\eta).$$

Otherwise $\beta \neq 0$, and by Lemma 12, $\alpha^* := \mathrm{m}_1(\beta) \preccurlyeq \mathrm{m}_1(\eta) =: \xi^*$.

A. If $\alpha^* \prec \xi^*$ then the induction hypothesis shows $\mathtt{dec1}(\alpha^*) <^l \mathtt{dec1}(\xi^*)$, hence the assertion follows using the previous lemma.

B. Otherwise $\alpha^* = \xi^*$, hence $\beta^* := \mathrm{m}_2(\beta) \preccurlyeq \mathrm{m}_2(\eta) =: \eta^*$ by Lemma 12.

   (a) If $\beta^* \prec \eta^*$ then the induction hypothesis shows $\mathtt{dec1}(\beta^*) <^l \mathtt{dec1}(\eta^*)$, hence the assertion follows again using the previous Lemma.

   (b) Otherwise $\beta^* = \eta^*$. Let $\rho := \varphi\alpha^*\beta^* = \varphi\xi^*\eta^*$. Hence $\beta, \eta \in \Gamma_0[\rho]$ by Lemma 9. The induction hypothesis now yields $\mathtt{dec2}(\rho, \beta) <^l \mathtt{dec2}(\rho, \eta)$, hence the assertion follows.

For 2., let $\beta, \eta \in \Gamma_0[\rho]$, ρ some φ-term, and assume $\beta \prec \eta$.

A. Assume β has the form $\rho + \gamma$.

   (a) Assume η has the form $\rho+\delta$. Then $\gamma \prec \delta$ and by induction hypothesis $\mathtt{dec1}(\gamma) <^l \mathtt{dec1}(\delta)$, hence the assertion follows.

   (b) Otherwise η does not have the form $\rho + \delta$. But then

$$\mathtt{dec2}(\rho, \beta) = (\mathbf{*} \ldots <^l (\mathbf{0} <^l \mathtt{dec2}(\rho, \eta).$$

B. Otherwise β is not of the form $\rho + \gamma$. By Lemma 11.2. it follows that also η is not of this form. Hence the answers to both `Query`'s are `No`. Furthermore, $\beta, \eta \in \tilde{\Gamma}_0[\rho]$, thus we can apply Lemma 13 to obtain $\alpha^{*,\rho} := \mathrm{m}_1^\rho(\beta) \preccurlyeq \mathrm{m}_1^\rho(\eta) =: \xi^{*,\rho}$.

   (a) If $\alpha^{*,\rho} \prec \xi^{*,\rho}$ then the induction hypothesis shows $\mathtt{dec1}(\alpha^{*,\rho}) <^l \mathtt{dec1}(\xi^{*,\rho})$, hence the assertion follows.

   (b) Otherwise $\alpha^{*,\rho} = \xi^{*,\rho}$. Then $\beta^{*,\rho} := \mathrm{m}_2^\rho(\beta) \preccurlyeq \mathrm{m}_2^\rho(\eta) =: \eta^{*,\rho}$ by Lemma 13.

      i. If $\beta^{*,\rho} \prec \eta^{*,\rho}$ then the induction hypothesis shows $\mathtt{dec2}(\rho, \beta^{*,\rho}) <^l \mathtt{dec2}(\rho, \eta^{*,\rho})$, hence the assertion follows.

      ii. Otherwise $\beta^{*,\rho} = \eta^{*,\rho}$. Let $\tilde{\rho} := \varphi\alpha^{*,\rho}\beta^{*,\rho} = \varphi\xi^{*,\rho}\eta^{*,\rho}$. Hence $\beta, \eta \in \Gamma_0[\tilde{\rho}]$ by Lemma 10. The induction hypothesis now yields $\mathtt{dec2}(\tilde{\rho}, \beta) <^l \mathtt{dec2}(\tilde{\rho}, \eta)$, hence the assertion follows.

That finishes the proof of Theorem 17.

Theorem 18 *Let $\prec$ be the ordering on normal forms. Then $T_2^1 \vdash WF_\prec$.*

Proof The task of finding a $\prec$-minimal $\beta \leq t$ with $A(\beta)$, $A \in \Sigma_1^b$, can now be reduced to finding a $<^l$-minimal $n \leq t'$ such that

$$A^*(n) \equiv (\exists \beta \leq t)(\mathtt{dec1}(\beta) = n \wedge A(\beta)) \quad \in \Sigma_1^b$$

holds. The bound t' must be large enough so that $\mathtt{dec1}(\beta) \leq t'$ whenever $\beta \leq t$; examination of the algorithms and Lemma 14 shows immediately that $\mathtt{dec1}(\beta)$ has length linearly bounded by $\mathrm{lh}(\beta)$, so $t' = t^{O(1)}$ suffices. Using the same technique as in section 3 for coding fixed length words order-preserving into natural numbers this task can be solved by $\Sigma_1^b - Min$, which is at hand in T_2^1. $\square$

5 Embedding into lexicographic orderings

Let A be a finite, ordered set of cardinality at least 2, and A^* be the set of words (finite sequences) over A with the lexicographic ordering. Similarly, $\mathbb{N}^*$ denotes the set of finite sequences of non-negative integers ordered lexicographically. The proofs of the Theorems 4 and 18 involved giving order-preserving embeddings of notations for ordinals less than ϵ_0 and Γ_0 into $\mathbb{N}^*$ or a set A^*. In this section, we consider the question of how general these order preserving embeddings are, and whether they always must exist.

First, we prove a simple fact showing that $\mathbb{N}^*$ can always be replaced by A^*.

Proposition 19 *There is an order-preserving embedding of N^* into A^*. Furthermore, this embedding is polynomial time computable.*

Proof Without loss of generality, A has the symbols $0 < 1$ as members. If $n \geq 0$, let $b_n \in \{0,1\}^*$ be the binary representation of n. Let 1^k be the word which consists of k 1's. The desired embedding σ is defined by letting $\sigma(\langle n_1, n_2, \ldots, n_k\rangle)$ equal $1^{|n_1|}0b_{n_1}1^{|n_2|}0b_{n_2}\cdots 1^{|n_k|}0b_{n_k}$. It is easy to verify that σ has the necessary properties. $\square$

Next we characterize which orderings can be embedded into A^* in an order-preserving fashion. Note that A^* is countable, so only countable orderings can be embedded into A^*.

Theorem 20 *Any countable ordering can be embedded into A^*.*

Proof Obviously any finite ordering can be embedded into A^*. So it suffices to assume that the ordering, denoted $\prec$, has domain the set $\mathbb{N}$. Let $A = \{0,1\}$. We shall define an ordering preserving embedding $\sigma : \mathbb{N} \mapsto A^*$. The values $\sigma(n)$ will be defined so that they always end with the symbol 1, i.e., $\sigma(n)$ is in the regular set $(0 \cup 1)^*1$. Define $pred(\alpha 1) = \alpha 01$ and $succ(\alpha 1) = \alpha 11$. We let $|\alpha|$ denote the number of symbols in the word α.

Inductively define $\sigma(n)$ as follows. First, $\sigma(0) = 1$. To define $\sigma(n+1)$, let i be the $\prec$-greatest element from $\{0,\ldots,n\}$ such that $i \prec (n+1)$, if any such i exists, and dually let j be the $\prec$-least element from $\{0,\ldots n\}$ such that $j \succ (n+1)$. In case i does not exist or $\sigma(i)$ is a proper subsequence of $\sigma(j)$, let $\sigma(n+1) = pred(\sigma(j))$. Otherwise, j does not exist or $\sigma(i)$ is not a proper subsequence of $\sigma(j)$, and in this case $\sigma(n+1) = succ(\sigma(i))$.

We leave to the reader the straightforward proofs that σ is one-to-one and order-preserving. $\square$

The above proof immediately implies that if $\prec$ is recursive (respectively, polynomial time), then σ is recursive (respectively, exponential time). Because of the exponential time complexity, this general result is not good enough to imply anything about provability of well-orderings over bounded domains in bounded arithmetic.

6 Ordinal cost functions for PLS

The class PLS of polynomial local search functions was originally defined by Johnson, Papadimitriou and Yannakakis [9] and contains functions which search for a minimal cost (equivalently, maximal cost) feasible solution. Buss and Krajíček [5] used this class to characterize the provably total functions of T^1_2 as being the set of functions which can be defined as the composition of a projection function and a PLS function.

We give a quick sketch of the definition of the class PLS and the reader can refer to above references for more detailed definitions.

Definition A *PLS problem* consists of a cost function c, a neighborhood function N, and a polynomially bounded set of feasible solutions, defined by a predicate F. For an input x, the set $\{s : F(x,s)\}$ is the set of *feasible solutions*, the mapping $s \mapsto c(x,s)$ assigns a cost to each feasible solution, and the mapping $s \mapsto N(x,s)$ maps feasible solutions to feasible solutions. The functions c and N and the predicate F must be polynomial time computable. The function defined by the PLS problem is the (multivalued) function f defined by $f(x) = y$ iff $F(x,y)$ and $c(x, N(x,y)) \not< c(x,y)$.

We can generalize PLS to allow the cost function to take on ordinal values instead of integer values. (It is for this reason that we defined PLS problems as a minimization problems rather than maximization problems.) Let α denote an ordinal, such as ϵ_0 or Γ_0, with a system of ordinal notations so that the set of valid ordinal notations is polynomial time recognizable and so that the induced ordering, $\preccurlyeq$, on ordinal notations is polynomial time. The class of $(\alpha, \preccurlyeq)$-PLS problems is defined identically to the class PLS except that the condition $c(x, N(x,y)) \not< c(x,y)$ is replaced by $c(x, N(x,y)) \not\prec c(x,y)$.

We shall see below that PLS and $(\epsilon_0, \preccurlyeq)$-PLS and $(\Gamma_0, \preccurlyeq)$-PLS are identical in that they contain exactly the same functions. To establish this result in its strongest form we need the following proposition.

Proposition 21 *Let $h(x)$ be a (multivalued) function such that $h = g \circ f$ where g is a polynomial time function and f is a PLS function. Further suppose that the graph of h, $\{(x,y) : h(x) = y\}$ is polynomial time. Then h is a PLS function.*

Proof Let f be the PLS function defined by c, N and F. We must define c', N' and F' that define the function h. These are defined as follows:

$$\begin{aligned}
c'(x,s) &= \begin{cases} 0 & \text{if } h(x) = s \\ c(x,s) + 1 & \text{otherwise} \end{cases} \\
N'(x,s) &= \begin{cases} s & \text{if } h(x) = s \\ g(s) & \text{if } h(x) = g(s) \text{ and } h(x) \neq s \\ N(x,s) & \text{otherwise} \end{cases} \\
F'(x,s) &\Leftrightarrow F(x,s) \vee h(x) = s
\end{aligned}$$

It is clear by inspection that c', N' and F' define h as a PLS problem. □

Theorem 22 *The classes $(\epsilon_0, \preccurlyeq)$-PLS and $(\Gamma_0, \preccurlyeq)$-PLS are both equal to PLS.*

To prove this theorem, first note that any PLS function is clearly in $(\epsilon_0, \preccurlyeq)$-PLS and in $(\Gamma_0, \preccurlyeq)$-PLS by using the construction from the proof of Theorem 3 to transform integer values of a cost function into ordinal notations. For the other direction, suppose that f is a function in $(\epsilon_0, \preccurlyeq)$-PLS or $(\Gamma_0, \preccurlyeq)$-PLS. By Theorem 4 or 18, T^1_2 can prove that for all inputs x, there is a feasible solution s of minimum cost, and hence of minimal cost. Therefore, by Buss-Krajíček [5], f can be expressed as the composition of a projection function and a PLS function. Since the graph of f is polynomial time, Proposition 21 implies that f is also a PLS function.

References

[1] A. Beckmann, *Separating Fragments of Bounded Arithmetic*, PhD thesis, Univ. of Münster, 1996.

[2] ——, *Notations for exponentiation*. Submitted for publication, 1999.

[3] S. R. Buss, *Bounded Arithmetic*, Bibliopolis, 1986. Revision of 1985 Princeton University Ph.D. thesis.

[4] ——, *Axiomatizations and conservation results for fragments of bounded arithmetic*, in Logic and Computation, proceedings of a Workshop held Carnegie-Mellon University, 1987, vol. 106 of Contemporary Mathematics, American Mathematical Society, 1990, pp. 57–84.

[5] S. R. Buss and J. Krajíček, *An application of Boolean complexity to separation problems in bounded arithmetic*, Proc. London Math. Society, 69 (1994), pp. 1–21.

[6] S. Feferman, *Systems of predicative analysis I*, Journal of Symbolic Logic, 29 (1964), pp. 1–30.

[7] ——, *Systems of predicative analysis II: Representations of ordinals*, Journal of Symbolic Logic, 30 (1968), pp. 193–220.

[8] P. Hájek and P. Pudlák, *Metamathematics of First-order Arithmetic*, Perspectives in Mathematical Logic, Springer-Verlag, Berlin, 1993.

[9] D. S. Johnson, C. H. Papadimitriou, and M. Yannakakis, *How easy is local search?*, J. Comput. System Sci., 37 (1988), pp. 79–100.

[10] J. Krajíček, *Bounded Arithmetic, Propositional Calculus and Complexity Theory*, Cambridge University Press, 1995.

[11] W. Pohlers, *Proof Theory: An Introduction*, Lecture Notes in Mathematics, #1409, Springer-Verlag, Berlin, 1989.

[12] K. Schütte, *Proof Theory*, Grundlehren der mathematischen Wissenschaften #225, Springer-Verlag, Berlin, 1977.

[13] R. Sommer, *Transfinite Induction and Hierarchies Generated by Transfinite Recursion within Peano Arithmetic*, PhD thesis, U.C. Berkeley, 1990.

[14] ——, *Ordinal arithmetic in $I\Delta_0$*, in Arithmetic, Proof Theory and Computational Complexity, P. Clote and J. Krajíček, eds., Oxford University (Clarendon) Press, 1993, pp. 320–363.