Multiply-Recursive Upper Bounds with Higman's Lemma

S. Schmitz and Ph. Schnoebelen

LSV, ENS Cachan & CNRS, Cachan, France
{schmitz,phs}@lsv.ens-cachan.fr

Abstract. We develop a new analysis for the length of controlled bad sequences in well-quasi-orderings based on Higman's Lemma. This leads to tight multiply-recursive upper bounds that readily apply to several verification algorithms for well-structured systems.

1 Introduction

Well-quasi-orderings (wqo's) are an important tool in logic and computer science [13]. They are the key ingredient to a large number of decidability (or finiteness, regularity, ...) results. In constraint solving, automated deduction, program analysis, and many more fields, wqo's usually appear under the guise of specific tools, like Dickson's Lemma (for tuples of integers), Higman's Lemma (for words and their subwords), Kruskal's Tree Theorem and its variants (for finite trees with embeddings), and recently the Robertson-Seymour Theorem (for graphs and their minors). In program verification, wqo's are the basis for *well-structured systems* [1, 10, 11], a generic framework for infinite-state systems.

Complexity. Wqo's are seldom used in complexity analysis. In order to extract complexity upper bounds for an algorithm whose termination proof rests on Dickson's or Higman's Lemma, one must be able to bound the length of so-called "controlled bad sequences" (see Def. 2.4). Here the available results are not very well known in computer science, and their current packaging does not make them easy to read and apply. For applications like the complexity of lossy channel systems [4] that rely on Higman's Lemma over Γ_p^* (the words over a p-letter alphabet), what we really need is something like:

Length Function Theorem. *Let $L_{\Gamma_p^*}(n)$ be the maximal length of bad sequences $w_0, w_1, w_2, \ldots$ over Γ_p^* with $p \geq 2$ s.t. $|w_i| < g^i(n)$ for $i = 0, 1, 2, \ldots$ If the* control function *g is primitive-recursive, then the* length function *$L_{\Gamma_p^*}$ is bounded by a function in $\mathscr{F}_{\omega^{p-1}}$.*[1]

Unfortunately, the literature contains no such clear statement (see the comparison with existing work below).

[1] Here the functions F_α are the ordinal-indexed levels of the Fast-Growing Hierarchy [14], with multiply-recursive complexity starting at level $\alpha = \omega$, i.e., Ackermannian complexity, and stopping just before level $\alpha = \omega^\omega$, i.e., hyper-Ackermannian complexity. The function classes $\mathscr{F}_\alpha$ denote their elementary-recursive closure.

L. Aceto, M. Henzinger, and J. Sgall (Eds.): ICALP 2011, Part II, LNCS 6756, pp. 441–452, 2011.

Our Contribution. We provide a new and *self-contained* proof of the Length Function Theorem, a fundamental result that (we think) deserves a wide audience. The exact statement we prove, Thm. 5.3 below, is rather general: it is parameterized by the control function g and accomodates various combinations of Γ_p^* sets without losing precision. For this we significantly extend the setting we developed for Dickson's Lemma [9]: We rely on iterated residuations with a simple but explicit algebraic framework for handling wqo's and their residuals in a compositional way. Our computations can be kept relatively simple by means of a fully explicit notion of "normed reflection" that captures the over-approximations we use, all the while enjoying good algebraic properties. We also show *how to apply* the Length Function Theorem by deriving precise multiply-recursive upper bounds, parameterized by the alphabet size, for the complexity of lossy channel systems and the Regular Post Embedding Problem (see Sec. 6).

Comparison with Existing Work. (Here, and for easier comparison, we assume that the control function g is the successor function.)

For $\mathbb{N}^k$ (i.e., Dickson's Lemma), Clote gives an explicit upper bound at level $\mathscr{F}_{k+6}$ extracted from complex Ramsey-theoretical results, hence hardly self-contained [8]. This is a simplification over an earlier analysis by McAloon, which leads to a uniform upper bound at level $\mathscr{F}_{k+1}$, but gives no explicit statement nor asymptotic analysis [15]. Both analyses are based on large intervals and extractions, and McAloon's is technically quite involved. With D. and S. Figueira, we improved this to an explicit and tight $\mathscr{F}_k$ [9].

For Γ_p^* (Higman's Lemma), Cichoń and Tahhan Bittar exhibit a reduction method, deducing bounds (for tuples of) words on Γ_p from bounds on the Γ_{p-1} case [7]. Their decomposition is clear and self-contained, with the control function made explicit. It ends up with some inequalities, collected in [7, Sec. 8], from which it is not clear what precisely are the upper bounds one can extract. Following this, Touzet claims a bound of F_{ω^p} [19, Thm. 1.2] with an analysis based on iterated residuations but the proof (given in [18]) is incomplete.

Finally, Weiermann gives an $\mathscr{F}_{\omega^{p-1}}$-like bound for Γ_p^* [20, Coro. 6.3] for sequences produced by term rewriting systems, but his analysis is considerably more involved (as can be expected since it applies to the more general Kruskal Theorem) and one cannot easily extract an explicit proof for his Coro. 6.3.

Regarding lower bounds, it is known that $F_{\omega^{p-1}}$ is essentially tight [6].

Outline of the Paper. All basic notions are recalled in Sec. 2, leading to the Descent Equation (3). Reflections in an algebraic setting are defined in Sec. 3, then transfered in an ordinal-arithmetic setting in Sec. 4. We prove the Main Theorem in Sec. 5, before illustrating its uses in Sec. 6. All the proofs missing from this extended abstract can be found at `http://arxiv.org/abs/1103.4399`.

2 Normed Wqo's and Controlled Bad Sequences

We recall some basic notions of wqo-theory [see e.g. 13]. A *quasi-ordering* (a "qo") is a relation $(A; \leq)$ that is reflexive and transitive. As usual, we write

$x < y$ when $x \leq y$ and $y \not\leq x$, and we denote structures $(A; P_1, \ldots, P_m)$ with just the support set A when this does not lead to ambiguities. Classically, the substructure *induced* by a subset $X \subseteq A$ is $(X; P_{1|X}, \ldots, P_{m|X})$ where, for a predicate P over A, $P_{|X}$ is its trace over X.

A qo A is a *well-quasi-ordering* (a "wqo") if every infinite sequence $x_0, x_1, x_2, \ldots$ contains an infinite increasing subsequence $x_{i_0} \leq x_{i_1} \leq x_{i_2} \cdots$ Equivalently, a qo is a wqo if it is well-founded (has no infinite strictly decreasing sequences) and contains no infinite antichains (i.e., set of pairwise incomparable elements). Every induced substructure of a wqo is a wqo.

Wqo's With Norms. A *norm function* over a set A is a mapping $|.|_A : A \to \mathbb{N}$ that provides every element of A with a positive integer, its *norm*, capturing a notion of size. For $n \in \mathbb{N}$, we let $A_{<n} \stackrel{\text{def}}{=} \{x \in A \mid |x|_A < n\}$ denote the subset of elements with norm below n. The norm function is said to be *proper* if $A_{<n}$ is finite for every n.

Definition 2.1. *A* normed wqo *(a "nwqo") is a wqo* $(A; \leq_A, |.|_A)$ *equipped with a proper norm function.*

There are no special conditions on norms, except being proper. In particular no connection is required between the ordering of elements and their norms. In applications, norms are related to natural complexity measures.

Example 2.2 (Some Basic Wqo's). The set of natural numbers $\mathbb{N}$ with the usual ordering is the smallest infinite wqo. For every $p \in \mathbb{N}$, we single out two p-element wqo's: $\searrow_p$ is the p-element initial segment of $\mathbb{N}$, i.e., the set $\{0, 1, 2, \ldots, p-1\}$ ordered linearly, while Γ_p is the p-letter alphabet $\{a_1, \ldots, a_p\}$ where distinct letters are unordered. We turn them into nwqo's by fixing the following:

$$|k|_{\mathbb{N}} = |k|_{\searrow_p} \stackrel{\text{def}}{=} k\,, \qquad\qquad |a_i|_{\Gamma_p} \stackrel{\text{def}}{=} 0\,. \tag{1}$$

We write $A \equiv B$ when the two nwqo's A and B are *isomorphic* structures. For all practical purposes, isomorphic nwqo's can be identified, following a standard practice that significantly simplifies the notational apparatus we develop in Sec. 3. For the moment, we only want to stress that, in particular, *norm functions must be preserved* by nwqo isomorphisms.

Example 2.3 (Isomorphism Between Basic Nwqo's). On the positive side, $\searrow_0 \equiv \Gamma_0$ and also $\searrow_1 \equiv \Gamma_1$ since $|a_1|_{\Gamma_1} = 0 = |0|_{\searrow_1}$. By contrast $\searrow_2 \not\equiv \Gamma_2$: not only these two have non-isomorphic order relations, they also have different norm functions.

Good, Bad, and Controlled Sequences. A sequence $\mathbf{x} = x_0, x_1, x_2, \ldots$ over a qo is *good* if $x_i \leq x_j$ for some positions $i < j$. It is *bad* otherwise. *Over a wqo*, all infinite sequences are good (equivalently, all bad sequences are finite).

We are interested in the maximal length of bad sequences for a given wqo. Here, a difficulty is that, in general, bad sequences can be arbitrarily long and there is no finite maximal length. However, in our applications we are only interested in bad sequences generated by some algorithmic method, i.e., bad sequences whose complexity is controlled in some way.

Definition 2.4 (Control Functions and Controlled Sequences) *A* control function *is a mapping* $g : \mathbb{N} \to \mathbb{N}$. *For a* size $n \in \mathbb{N}$, *a sequence* $\mathbf{x} = x_0, x_1, x_2, \ldots$ *over a nwqo* A *is* (g,n)-controlled $\overset{\text{def}}{\Leftrightarrow}$

$$\forall i = 0, 1, 2, \ldots : \ |x_i|_A < g^i(n) \ = \ \overbrace{g(g(\ldots g}^{i \text{ times}}(n))) \, .$$

Why n is called a "size" appears with Prop. 2.8 and its proof. A pair (g,n) is just called a *control* for short. We say that a sequence $\mathbf{x}$ is n-*controlled* (or just *controlled*), when g (resp. g and n) is clear from the context. Observe that the empty sequence is always a controlled sequence.

Proposition 2.5. *Let* A *be a nwqo and* (g,n) *a control. There exists a finite maximal length* $L \in \mathbb{N}$ *for* (g,n)*-controlled bad sequences over* A.

We write $L_{A,g}(n)$ for this maximal length, a number that depends on all three parameters: A, g and n. However, for complexity analysis, the relevant information is how, for given A and g, the *length function* $L_{A,g} : \mathbb{N} \to \mathbb{N}$ behaves asymptotically, hence our choice of notation. Furthermore, g is a parameter that remains fixed in our analysis and applications, hence it is usually left implicit. **From now on we assume a fixed control function** g and just write $L_A(n)$ for $L_{A,g}(n)$. We further assume that g is *smooth* ($\overset{\text{def}}{\Leftrightarrow}$ $g(x+1) \geq g(x)+1 \geq x+2$ for all x), which is harmless for applications but simplifies computations like (4).

Residuals Wqo's and a Descent Equation. Via residuals one expresses the length function by induction over nwqo's.

Definition 2.6 (Residuals). *For a nwqo* A *and an element* $x \in A$, *the* residual A/x *is the substructure (a nwqo) induced by the subset* $A/x \overset{\text{def}}{=} \{y \in A \mid x \not\leq y\}$ *of elements that are not above* x.

Example 2.7 (Residuals of Basic Nwqo's). For all $k < p$ and $i = 1, \ldots, p$:

$$\mathbb{N}/k = \searrow_p/k = \searrow_k \, , \qquad\qquad \Gamma_p/a_i \equiv \Gamma_{p-1} \, . \tag{2}$$

Proposition 2.8 (Descent Equation)

$$L_A(n) = \max_{x \in A_{<n}} \left\{1 + L_{A/x}(g(n))\right\} . \tag{3}$$

This reduces the L_A function to a finite combination of L_{A_i}'s where the A_i's are residuals of A, hence "smaller" sets. Residuation is well-founded for wqo's: a sequence of successive residuals $A \supsetneq A/x_0 \supsetneq A/x_0/x_1 \supsetneq A/x_0/x_1/x_2 \supsetneq \cdots$ is necessarily finite since $x_0, x_1, x_2, \ldots$ must be a bad sequence. Hence the recursion in the Descent Equation is well-founded and can be used to evaluate $L_A(n)$. This is our starting point for analyzing the behaviour of length functions.

For example, using induction and Eq. (2), the Descent Equation leads to:

$$L_{\Gamma_p}(n) = p \, , \qquad L_{\mathbb{N}}(n) = n \, , \qquad L_{\searrow_p}(n) = \min(n, p) \, . \tag{4}$$

3 An Algebra of Normed Wqo's

The algebraic framework we now develop has two main goals. First it provides a *notation* for denoting the wqo's encountered in algorithmic applications. These wqo's and their norm functions abstract data structures that are built inductively by combining some basic wqo's. Second, it supports a *calculus* for the kind of compositional computations, based on the Descent Equation, we develop next.

The constructions we use in this paper are disjoint sums, cartesian products, and Kleene stars (with Higman's order). These constructions are classic. Here we also have to define how they combine the norm functions:

Definition 3.1 (Sums, Products, Stars Nwqo's) *The* disjoint sum $A_1 + A_2$ *of two nwqos* A_1 *and* A_2 *is the nwqo given by*

$$A_1 + A_2 = \{\langle i, x\rangle \mid i \in \{1,2\} \text{ and } x \in A_i\}\,, \tag{5}$$

$$\langle i, x\rangle \leq_{A_1+A_2} \langle j, y\rangle \overset{\text{def}}{\Leftrightarrow} i = j \text{ and } x \leq_{A_i} y\,, \tag{6}$$

$$|\langle i, x\rangle|_{A_1+A_2} \overset{\text{def}}{=} |x|_{A_i}\,. \tag{7}$$

The cartesian product $A_1 \times A_2$ *of two nwqos* A_1 *and* A_2 *is the nwqo given by*

$$A_1 \times A_2 = \{\langle x_1, x_2\rangle \mid x_1 \in A_1, x_2 \in A_2\}\,, \tag{8}$$

$$\langle x_1, x_2\rangle \leq_{A_1\times A_2} \langle y_1, y_2\rangle \overset{\text{def}}{\Leftrightarrow} x_1 \leq_{A_1} y_1 \text{ and } x_2 \leq_{A_2} y_2\,, \tag{9}$$

$$|\langle x_1, x_2\rangle|_{A_1\times A_2} \overset{\text{def}}{=} \max(|x_1|_{A_1}, |x_2|_{A_2})\,. \tag{10}$$

The Kleene star A^* *of a nwqo* A *is the nwqo given by*

$$A^* \overset{\text{def}}{=} \text{ all finite lists } (x_1 \dots x_n) \text{ of elements of } A\,, \tag{11}$$

$$(x_1 \dots x_n) \leq_{A^*} (y_1 \dots y_m) \overset{\text{def}}{\Leftrightarrow} \begin{cases} x_1 \leq_A y_{i_1} \wedge \dots \wedge x_n \leq_A y_{i_n} \\ \text{for some } 1 \leq i_1 < i_2 < \dots < i_n \leq m \end{cases}\,, \tag{12}$$

$$|(x_1 \dots x_n)|_{A^*} \overset{\text{def}}{=} \max(n, |x_1|_A, \dots, |x_n|_A)\,. \tag{13}$$

It is well-known (and plain) that $A_1 + A_2$ and $A_1 \times A_2$ are indeed wqo's when A_1 and A_2 are. The fact that A^* is a wqo when A is, is a classical result called Higman's Lemma. We let the reader check that the norm functions defined in Equations (7), (10), and (13), are proper and turn $A_1 + A_2$, $A_1 \times A_2$ and A^* into nwqo's. Finally, we note that nwqo isomorphism is a congruence for sum, product and Kleene star.

Notation (0 and 1) *We let* $\mathbf{0}$ *and* $\mathbf{1}$ *be short-hand notations for, respectively,* Γ_0 *(the empty nwqo) and* Γ_1 *(the singleton nwqo with the* 0 *norm).*

This is convenient for writing down the following algebraic properties:

Proposition 3.2 *The following isomorphisms hold:*

$$A + B \equiv B + A\,, \qquad A + (B + C) \equiv (A + B) + C\,,$$
$$A \times B \equiv B \times A\,, \qquad A \times (B \times C) \equiv (A \times B) \times C\,,$$
$$\mathbf{0} + A \equiv A\,, \qquad \mathbf{1} \times A \equiv A\,,$$
$$\mathbf{0} \times A \equiv \mathbf{0}\,, \qquad (A + A') \times B \equiv (A \times B) + (A' \times B)\,,$$
$$\mathbf{0}^* \equiv \mathbf{1}\,, \qquad \mathbf{1}^* \equiv \mathbb{N}\,.$$

In view of these properties, we freely write $A \cdot k$ and A^k for the k-fold sums and products $A + \cdots + A$ and $A \times \cdots \times A$. Observe that $A \cdot k \equiv A \times \Gamma_k$.

Reflecting Normed Wqo's. Reflections are the main comparison/abstraction tool we shall use. They let us simplify instances of the Descent Equation by replacing all A/x for $x \in A_{<n}$ by a single (or a few) A' that is smaller than A but large enough to reflect all considered A/x's.

Definition 3.3 *A* nwqo reflection *is a mapping $h : A \to B$ between two nwqo's that satisfies the two following properties:*

$$\forall x, y \in A : h(x) \leq_B h(y) \text{ implies } x \leq_A y \,,$$
$$\forall x \in A : |h(x)|_B \leq |x|_A \,.$$

In other words, a nwqo reflection is an order reflection that is also norm-decreasing (not necessarily strictly).

We write $h : A \hookrightarrow B$ when h is a nwqo reflection and say that B *reflects* A. This induces a relation between nwqos, written $A \hookrightarrow B$.

Reflection is transitive since $h : A \hookrightarrow B$ and $h' : B \hookrightarrow C$ entails $h' \circ h : A \hookrightarrow C$. It is also reflexive, hence reflection is a quasi-ordering. Any nwqo reflects its substructures since $Id : X \hookrightarrow A$ when X is a substructure of A. Thus $\mathbf{0} \hookrightarrow A$ for any A, and $\mathbf{1} \hookrightarrow A$ for any non-empty A.

Example 3.4 Among the basic nwqos from Example 2.2, we note the following relations (or absences thereof). For any $p \in \mathbb{N}$, $\searrow_p \hookrightarrow \Gamma_p$, while $\Gamma_p \not\hookrightarrow \searrow_p$ when $p \geq 2$. The reflection of substructures yields $\searrow_p \hookrightarrow \mathbb{N}$ and $\Gamma_p \hookrightarrow \Gamma_{p+1}$. Obviously, $\mathbb{N} \not\hookrightarrow \searrow_p$ and $\Gamma_{p+1} \not\hookrightarrow \Gamma_p$.

Reflections preserve controlled bad sequences. Let $h : A \hookrightarrow B$, consider a sequence $\mathbf{x} = x_0, x_1, \ldots, x_l$ over A, and write $h(\mathbf{x})$ for $h(x_0), h(x_1), \ldots, h(x_l)$, a sequence over B. Then $h(\mathbf{x})$ is bad when $\mathbf{x}$ is, and n-controlled when $\mathbf{x}$ is. Hence:

$$A \hookrightarrow B \text{ implies } L_A(n) \leq L_B(n) \text{ for all } n \,. \tag{14}$$

Reflections are compatible with product, sum, and Kleene star.

Proposition 3.5 (Reflection is a Preconguence)

$$A \hookrightarrow A' \text{ and } B \hookrightarrow B' \text{ imply } A + B \hookrightarrow A' + B' \text{ and } A \times B \hookrightarrow A' \times B' \,, \tag{15}$$
$$A \hookrightarrow A' \text{ implies } A^* \hookrightarrow A'^* \,. \tag{16}$$

Computing and Reflecting Residuals. We may now tackle our first main problem: computing residuals A/x. This is done by induction over the structure of A.

Proposition 3.6 (Inductive Rules For Residuals)

$$(A + B)/\langle 1, x\rangle = (A/x) + B \,, \quad (A + B)/\langle 2, x\rangle = A + (B/x) \,, \tag{17}$$
$$(A \times B)/\langle x, y\rangle \hookrightarrow \big[(A/x) \times B\big] + \big[A \times (B/y)\big] \,, \tag{18}$$
$$A^*/(x_1 \ldots x_n) \hookrightarrow \Gamma_n \times A^n \times (A/x_1)^* \times \cdots \times (A/x_n)^* \,, \tag{19}$$
$$\Gamma^*_{p+1}/(x_1 \ldots x_n) \hookrightarrow \Gamma_n \times (\Gamma^*_p)^n \,. \tag{20}$$

Equation (20) is a refinement of (19) in the case of finite alphabets.

Since it provides reflections instead of isomorphisms, Prop. 3.6 is not meant to support exact computations of A/x by induction over the structure of A. More to the point, it yields over-approximations that are sufficiently precise for our purposes while bringing important simplifications when we have to reflect (the max of) all A/x for all $x \in A_{<n}$.

4 Reflecting Residuals in Ordinal Arithmetic

We now introduce an *ordinal* notation for nwqo's. The purpose is twofold. Firstly, the ad-hoc techniques we use for evaluating, reflecting, and comparing residual nwqo's are more naturally stated within the language of ordinal arithmetic. Secondly, these ordinals will be essential for bounding L_A using functions in subrecursive hierarchies. For these developments, we restrict ourselves to *exponential* nwqo's, i.e., nwqo's obtained from finite Γ_p's with sums, products, and *Kleene star restricted to the* Γ_p*'s*. Modulo isomorphism, $\mathbb{N}^k \equiv \prod_{i=1}^{k} \Gamma_1^*$ is exponential.

Ordinal Terms. We use Greek letters like $\alpha, \beta, \ldots$ to denote ordinal terms in Cantor Normal Form (CNF) built using 0, addition, and ω-exponentiation (we restrict ourselves to ordinals $< \varepsilon_0$). A term α has the general form $\alpha = \omega^{\beta_1} + \omega^{\beta_2} + \cdots + \omega^{\beta_m}$ with $\beta_1 \geq \beta_2 \geq \cdots \geq \beta_m$ (ordering defined below) and where we distinguish between three cases: α is 0 if $m = 0$, α is a *successor* if ($m > 0$ and) $\beta_m = 0$, α is a *limit* if $\beta_m \neq 0$ (in the following, λ will always denote a limit, and we write $\alpha + 1$ rather than $\alpha + \omega^0$ for a successor). We say that α is principal (additive) if $m = 1$.

Ordering among our ordinals is defined inductively by

$$\alpha < \alpha' \overset{\text{def}}{\Leftrightarrow} \begin{cases} \alpha = 0 \text{ and } \alpha' \neq 0, \text{ or} \\ \alpha = \omega^\beta + \gamma,\ \alpha' = \omega^{\beta'} + \gamma' \text{ and } \begin{cases} \beta < \beta', \text{ or} \\ \beta = \beta' \text{ and } \gamma < \gamma'. \end{cases} \end{cases} \tag{21}$$

We let $\mathrm{CNF}(\alpha)$ denote the set of ordinal terms $<\alpha$.

For $c \in \mathbb{N}$, $\omega^\beta \cdot c$ denotes the c-fold addition $\omega^\beta + \cdots + \omega^\beta$. We sometimes write terms under a "strict" form $\alpha = \omega^{\beta_1} \cdot c_1 + \omega^{\beta_2} \cdot c_2 + \cdots + \omega^{\beta_m} \cdot c_m$ with $\beta_1 > \beta_2 > \cdots > \beta_m$, where the c_i's, called *coefficients*, must be > 0.

Recall the definitions of the *natural sum* $\alpha \oplus \alpha'$ and *natural product* $\alpha \otimes \alpha'$ of two terms in $\mathrm{CNF}(\varepsilon_0)$:

$$\sum_{i=1}^{m} \omega^{\beta_i} \oplus \sum_{j=1}^{n} \omega^{\beta'_j} \overset{\text{def}}{=} \sum_{k=1}^{m+n} \omega^{\gamma_k}\,, \qquad \sum_{i=1}^{m} \omega^{\beta_i} \otimes \sum_{j=1}^{n} \omega^{\beta'_j} \overset{\text{def}}{=} \bigoplus_{i=1}^{m} \bigoplus_{j=1}^{n} \omega^{\beta_i \oplus \beta'_j}\,,$$

where $\gamma_1 \geq \cdots \geq \gamma_{m+n}$ is a rearrangement of $\beta_1, \ldots, \beta_m, \beta'_1, \ldots, \beta'_n$. For $\alpha \in \mathrm{CNF}(\omega^{\omega^\omega})$, the decomposition $\alpha = \sum_{i=1}^{m} \omega^{\beta_i}$ uses β_i's that are in $\mathrm{CNF}(\omega^\omega)$, i.e., of the form $\beta_i = \sum_{j=1}^{k_i} \omega^{p_{i,j}}$ (with each $p_{i,j} < \omega$) so that ω^{β_i} is $\bigotimes_{j=1}^{k_i} \omega^{\omega^{p_{i,j}}}$. A term ω^{ω^p} is called a *principal multiplicative*.

We map exponential nwqo's to ordinals in $\mathrm{CNF}(\omega^{\omega^\omega})$ using their *maximal order type* [12]. Formally $o(A)$ is defined by

$$o(\Gamma_p) \stackrel{\text{def}}{=} p\,, \qquad o(\Gamma_0^*) \stackrel{\text{def}}{=} \omega^0\,, \qquad o(\Gamma_{p+1}^*) \stackrel{\text{def}}{=} \omega^{\omega^p}\,, \tag{22}$$

$$o(A+B) \stackrel{\text{def}}{=} o(A) \oplus o(B)\,, \qquad o(A \times B) \stackrel{\text{def}}{=} o(A) \otimes o(B)\,. \tag{23}$$

Conversely, there is a *canonical exponential nwqo* $C(\alpha)$ for each α in $\mathrm{CNF}(\omega^{\omega^\omega})$:

$$C\Big(\omega^{\beta_1} + \cdots + \omega^{\beta_m}\Big) = C\Big(\bigoplus_{i=1}^{m} \bigotimes_{j=1}^{k_i} \omega^{\omega^{p_{i,j}}}\Big) \stackrel{\text{def}}{=} \sum_{i=1}^{m} \prod_{j=1}^{k_i} \Gamma^*_{(p_{i,j}+1)}\,. \tag{24}$$

Then, o and C are bijective inverses (modulo isomorphism of nwqo's), compatible with sums and products. This correspondence equates between terms that, on one side, denote partial orderings with norms, and on the other side, ordinals in $\mathrm{CNF}(\omega^{\omega^\omega})$.

Derivatives. We aim to replace the "all A/x for $x \in A_{<n}$" by a computation of "some derived $\alpha' \in \partial_n\alpha$" where $\alpha = o(A)$, see Thm. 4.1 below. For this purpose, the definition of derivatives is based on the inductive rules in Prop. 3.6.

Let $n > 0$ be some norm. We start with principal ordinals and define

$$D_n\Big(\omega^{\omega^p}\Big) \stackrel{\text{def}}{=} \begin{cases} n-1 & \text{if } p = 0, \\ \omega^{(\omega^{p-1}\cdot(n-1))} \cdot (n-1) & \text{otherwise.} \end{cases} \tag{25}$$

$$D_n\left(\omega^{\omega^{p_1}+\cdots+\omega^{p_k}}\right) \stackrel{\text{def}}{=} \bigoplus_{j=1}^{k} \left(D_n\Big(\omega^{\omega^{p_j}}\Big) \otimes \bigotimes_{\ell \neq j} \omega^{\omega^{p_\ell}} \right)\,. \tag{26}$$

Now, with any $\alpha \in \mathrm{CNF}(\omega^{\omega^\omega})$, we associate the set of its *derivatives* $\partial_n\alpha$ with

$$\partial_n\Big(\sum_{i=1}^{m} \omega^{\beta_i}\Big) \stackrel{\text{def}}{=} \Big\{ D_n\big(\omega^{\beta_i}\big) \oplus \sum_{\ell \neq i} \omega^{\beta_\ell} \;\Big|\; i = 1, \ldots, m \Big\}\,. \tag{27}$$

This yields, for example, and assuming $p, k > 0$:

$$D_n(1) = 0,\ D_n(\omega) = n-1,\ D_n\big(\omega^{\omega^p \cdot k}\big) = \omega^{[\omega^p\cdot(k-1)+\omega^{p-1}\cdot(n-1)]} \cdot k(n-1)\,, \tag{28}$$

$$\partial_n 0 = \emptyset,\ \partial_n 1 = \{0\},\ \partial_n\omega = \{n-1\},\ \partial_n(\omega^\beta \cdot (k+1)) = \{\omega^\beta \cdot k \oplus D_n(\omega^\beta)\}. \tag{29}$$

Thus $\partial_n\alpha$ can be a singleton even when α is not principal, e.g., $\partial_n(p+1) = \{p\}$. We sometimes write $\alpha\ \partial_n\ \alpha'$ instead of $\alpha' \in \partial_n\alpha$, seeing ∂_n as a relation. Note that $\partial_n\alpha \subseteq \mathrm{CNF}(\alpha)$, hence $\partial \stackrel{\text{def}}{=} \bigcup_{n<\omega} \partial_n$ is well-founded.

Theorem 4.1 (Reflection by Derivatives). *Let $x \in A_{<n}$ for some exponential A. Then there exists $\alpha' \in \partial_n o(A)$ s.t. $A/x \hookrightarrow C(\alpha')$.*

Combining with equations (3) and (14), we obtain:

$$L_{C(\alpha)}(n) \leq \max_{\alpha' \in \partial_n\alpha} \big\{1 + L_{C(\alpha')}(g(n))\big\}\,. \tag{30}$$

5 Classifying L Using Subrecursive Hierarchies

For α in $\mathrm{CNF}(\omega^{\omega^\omega})$, define

$$M_\alpha(n) \stackrel{\text{def}}{=} \max_{\alpha' \in \partial_n \alpha} \{1 + M_{\alpha'}(g(n))\} \ . \tag{31}$$

(Recall that ∂ is well-founded, thus (31) is well-defined). Comparing with (30), we see that M_α bounds the length function: $M_\alpha(n) \geq L_{C(\alpha)}(n)$.

This defines an ordinal-indexed family of functions $(M_\alpha)_{\alpha \in \mathrm{CNF}(\omega^{\omega^\omega})}$ similar to some classical subrecursive hierarchies, with the added twist of the max operation—see [2, 16] for somewhat similar hierarchies. This is a real issue and one cannot replace a "$\max_{\alpha \in \ldots}\{M_\alpha(x)\}$" with "$M_{\sup\{\alpha \in \ldots\}}(x)$" since M_α is not always bounded by $M_{\alpha'}$ when $\alpha < \alpha'$. E.g., $M_{n+2}(n) = n+2 > M_\omega(n) = n+1$.

Subrecursive Hierarchies have been introduced as generators of classes of functions. For instance, writing $\mathscr{F}_\alpha$ for the class of functions elementary-recursive in the function F_α of the *fast growing hierarchy*, we can characterize the set of primitive-recursive functions as $\bigcup_{k<\omega} \mathscr{F}_k$, or that of multiply-recursive functions as $\bigcup_{\beta<\omega^\omega} \mathscr{F}_\beta$ [14].

Let us introduce (slight generalizations of) several classical hierarchies from [14, 7]. Those hierarchies are defined through assignments of *fundamental sequences* $(\lambda_x)_{x<\omega}$ for limit ordinals $\lambda < \varepsilon_0$, verifying $\lambda_x < \lambda$ for all x and $\lambda = \sup_x \lambda_x$. A standard assignment is defined by:

$$(\gamma + \omega^{\beta+1})_x \stackrel{\text{def}}{=} \gamma + \omega^\beta \cdot (x+1)\,, \qquad (\gamma + \omega^\lambda)_x \stackrel{\text{def}}{=} \gamma + \omega^{\lambda_x}\,, \tag{32}$$

where γ can be 0. Note that, in particular, $\omega_x = x+1$. Given an assignment of fundamental sequences, one can define the (x-indexed) *predecessor* $P_x(\alpha) < \alpha$ of an ordinal $\alpha \neq 0$ as

$$P_x(\alpha+1) \stackrel{\text{def}}{=} \alpha\,, \qquad P_x(\lambda) \stackrel{\text{def}}{=} P_x(\lambda_x)\,. \tag{33}$$

Given a fixed smooth control function h, the *Hardy hierarchy* $(h^\alpha)_{\alpha<\varepsilon_0}$ is then defined by

$$h^0(x) \stackrel{\text{def}}{=} x, \qquad h^{\alpha+1}(x) \stackrel{\text{def}}{=} h^\alpha(h(x)), \qquad h^\lambda(x) \stackrel{\text{def}}{=} h^{\lambda_x}(x)\,. \tag{34}$$

A closely related hierarchy is the *length hierarchy* $(h_\alpha)_{\alpha<\varepsilon_0}$ defined by

$$h_0(x) \stackrel{\text{def}}{=} 0, \qquad h_{\alpha+1}(x) \stackrel{\text{def}}{=} 1 + h_\alpha(h(x)), \qquad h_\lambda(x) \stackrel{\text{def}}{=} h_{\lambda_x}(x)\,. \tag{35}$$

Last of all, the *fast growing hierarchy* $(f_\alpha)_{\alpha<\varepsilon_0}$ is defined through

$$f_0(x) \stackrel{\text{def}}{=} h(x), \qquad f_{\alpha+1}(x) \stackrel{\text{def}}{=} f_\alpha^{\omega_x}(x), \qquad f_\lambda \stackrel{\text{def}}{=} f_{\lambda_x}(x)\,. \tag{36}$$

Standard versions of these hierarchies are usually defined by setting h as the successor function, in which case they are denoted H^α, H_α, and F_α resp.

Lemma 5.1. *For all $\alpha \in \mathrm{CNF}(\omega^{\omega^\omega})$ and $x \in \mathbb{N}$,*

1. *$h_\alpha(x) = 1 + h_{P_x(\alpha)}(h(x))$ when $\alpha > 0$,*
2. *$h_\alpha(x) \leq h^\alpha(x) - x$,*
3. *$h^{\omega^\alpha \cdot r}(x) = f_\alpha^r(x)$ for all $r < \omega$,*
4. *if h is eventually bounded by F_γ, then f_α is eventually bounded by $F_{\gamma+\alpha}$.*

Bounding the Length Function. Item 1 of Lem. 5.1 shows that M_α and h_α have rather similar expressions, based on derivatives for M_α and predecessors for h_α; they are in fact closely related:

Proposition 5.2. *For all α in $\mathrm{CNF}(\omega^{\omega^\omega})$, there is a constant k s.t. for all $n > 0$, $M_{\alpha,g}(n) \leq h_\alpha(kn)$ where $h(x) \stackrel{\text{def}}{=} x \cdot g(x)$.*

Proposition 5.2 translates for $n, p > 0$ into an

$$L_{\Gamma_p^*,g}(n) \leq h_{\omega^{\omega^{p-1}}}((p-1)n) \quad \text{for } h(x) \stackrel{\text{def}}{=} x \cdot g(x) \tag{37}$$

upper bound on bad (g,n)-controlled sequences in Γ_p^*. We believe (37) answers a wish expressed by Cichoń and Tahhan Bittar in their conclusion [7]: "an appropriate bound should be given by the function $h_{\omega^{\omega^{p-1}}}$, for some reasonable h."

It remains to translate the bound of Prop. 5.2 into a more intuitive and readily usable one. Combined with items 2–4 of Lem. 5.1, Prop. 5.2 allows us to state a fairly general result in terms of the $(\mathscr{F}_\alpha)_\alpha$ classes in the two most relevant cases (of which both the Length Function Theorem given in the introduction and, if $\gamma \geq 2$, the $\mathscr{F}_{\gamma+k}$ bound given for $\mathbb{N}^k$ in [9], are consequences):

Theorem 5.3 (Main Theorem). *Let g be a smooth control function eventually bounded by a function in $\mathscr{F}_\gamma$, and let A be an exponential nwqo with maximal order type $< \omega^{\beta+1}$. Then $L_{A,g}$ is bounded by a function in*

- *$\mathscr{F}_\beta$ if $\gamma < \omega$ (e.g. if g is primitive-recursive) and $\beta \geq \omega$,*
- *$\mathscr{F}_{\gamma+\beta}$ if $\gamma \geq 2$ and $\beta < \omega$.*

6 Refined Complexity Bounds for Verification Problems

This section provides two *examples* where our Main Theorem leads to precise multiply-recursive complexity upper bounds for problems that were known to be decidable but not primitive-recursive. Our choice of examples is guided by our close familiarity with these problems (in fact, they have been our initial motivation for looking at subrecursive hierarchies) and by their current role as master problems for showing Ackermann complexity lower bounds in several areas of verification. (A more explicit vademecum for potential users of the Main Theorem can be found in [9].)

Lossy Channel Systems. The wqo associated with a lossy channel system $S = (Q, M, C, \Delta)$ is the set $A_S \stackrel{\text{def}}{=} Q \times (M^*)^C$ of its configurations, ordered with embedding (see details in [4]). Here Q is a set of q control locations, M is a size-m message alphabet and C is a set of c channels. Hence, we obtain $A_S \equiv q \cdot \left(\Gamma_m^*\right)^c$. For such lossy systems [17], reachability, safety and termination can be decided by algorithms that only need to explore bad sequences over A_S. In particular, S has a non-terminating run from configuration s_{init} iff it has a run of length

$L_{A_S}(|s_{\text{init}}|)$, and the shortest run (if one exists) reaching s_{final} from s_{init} has length at most $L_{A_S}(|s_{\text{final}}|)$. Here the sequences (runs of S, forward or backward) are controlled with $g = \mathit{Succ}$. Now, since $o(A_S) = \omega^{(\omega^{m-1}\cdot c)} \cdot q$, Thm. 5.3 gives an overall complexity at level $\mathscr{F}_{\omega^{(m-1)\cdot c}}$, which is the most precise upper bound so far for lossy channel systems.

Regarding lower bounds, the construction in [4] proves a $\mathscr{F}_{\omega^K}$ lower bound for systems using $m = K + 2$ different symbols, $c = 2$ channels, and a quadratic $q \in O(K^2)$ number of states. If emptiness tests are allowed (an harmless extension for lossy systems, see [17]) one can even get rid of the # separator symbol in that construction (using more channels instead) and we end up with $m = K + 1$ and $c = 4$. Thus the demonstrated upper and lower bounds are very close, and tight when considering the recursivity-multiplicity level.

$\mathsf{PEP}^{\text{reg}}$, *the Regular Post Embedding Problem,* is an abstract problem that relaxes Post's Correspondence Problem by replacing the equality "$u_{i_1} \dots u_{i_n} = v_{i_1} \dots v_{i_n}$" with embedding "$u_{i_1} \dots u_{i_n} \leq_{\Gamma^*} v_{i_1} \dots v_{i_n}$" (all this under a "$\exists i_1, \dots, i_n$ in some regular R" quantification). It was introduced in [3] where decidability was shown thanks to Higman's Lemma. Non-trivial reductions between $\mathsf{PEP}^{\text{reg}}$ and lossy channel systems exist. Due to its abstract nature, $\mathsf{PEP}^{\text{reg}}$ is a potentially interesting master problem for proving hardness at multiply-recursive and hyper-Ackermannian, i.e., $\mathscr{F}_{\omega^\omega}$, levels (see refs in [5]).

A pumping lemma was proven in [5], which relies on the L_A function, and from which we can now derive more precise complexity upper bounds. Precisely, the proof of Lem. 7.3 in [5] shows that if a $\mathsf{PEP}^{\text{reg}}$ instance admits a solution $\sigma = i_1 \dots i_n$ longer than some bound H then that solution is not the shortest. Here H is defined as $2 \cdot L_{(\Gamma^* \cdot n)}(0)$ for a n that is at most exponential in the size of the instance. Since the control function is linear, Thm. 5.3 yields an $\mathscr{F}_{\omega^{p-1}}$ complexity upper bound for $\mathsf{PEP}^{\text{reg}}$ on a p-letter alphabet (and a hyper-Ackermannian $\mathscr{F}_{\omega^\omega}$ when the alphabet is not fixed). This motivates a closer consideration of lower bounds (left as future work, e.g., by adapting [4]).

7 Concluding Remarks

We proved a general version of the Main Theorem promised in the introduction. Our proof relies on two main components: an algebraic framework for normed wqo's and normed reflections on the one hand, leading on the other hand to descending relations between ordinals that can be captured in subrecursive hierarchies. This setting accommodates all "exponential" wqo's, i.e., finite combinations of Γ_p^*'s. This lets us derive upper bounds for controlled bad sequences when using Higman's Lemma on finite alphabets.

We hope that our framework will extend smoothly beyond exponential wqo's and may also accept additional wqo constructions like powersets, multisets, and perhaps trees.

References

1. Abdulla, P.A., Čerāns, K., Jonsson, B., Tsay, Y.K.: Algorithmic analysis of programs with well quasi-ordered domains. Inform. and Comput. 160, 109–127 (2000)
2. Buchholz, W., Cichoń, E.A., Weiermann, A.: A uniform approach to fundamental sequences and hierarchies. Math. Logic Quart. 40, 273–286 (1994)
3. Chambart, P., Schnoebelen, P.: Post embedding problem is not primitive recursive, with applications to channel systems. In: Arvind, V., Prasad, S. (eds.) FSTTCS 2007. LNCS, vol. 4855, pp. 265–276. Springer, Heidelberg (2007)
4. Chambart, P., Schnoebelen, P.: The ordinal recursive complexity of lossy channel systems. In: Proc. LICS 2008, pp. 205–216. IEEE, Los Alamitos (2008)
5. Chambart, P., Schnoebelen, P.: Pumping and counting on the Regular Post Embedding Problem. In: Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G. (eds.) ICALP 2010. LNCS, vol. 6199, pp. 64–75. Springer, Heidelberg (2010)
6. Cichoń, E.A.: Ordinal complexity measures. In: Conference on Proofs and Computations in Honour of Stan Wainer on the Occasion of his 65th Birthday (2009)
7. Cichoń, E.A., Tahhan Bittar, E.: Ordinal recursive bounds for Higman's Theorem. Theor. Comput. Sci. 201, 63–84 (1998)
8. Clote, P.: On the finite containment problem for Petri nets. Theor. Comput. Sci. 43, 99–105 (1986)
9. Figueira, D., Figueira, S., Schmitz, S., Schnoebelen, P.: Ackermannian and primitive-recursive bounds with Dickson's Lemma. In: Proc. LICS 2011. IEEE, Los Alamitos (to appear, 2011); arXiv:1007.2989 (cs.LO), `http://arxiv.org/abs/1007.2989`
10. Finkel, A., Schnoebelen, P.: Well-structured transition systems everywhere! Theor. Comput. Sci. 256, 63–92 (2001)
11. Henzinger, T.A., Majumdar, R., Raskin, J.F.: A classification of symbolic transition systems. ACM Trans. Comput. Logic 6, 1–32 (2005)
12. de Jongh, D.H.J., Parikh, R.: Well-partial orderings and hierarchies. Indag. Math. 39, 195–207 (1977)
13. Kruskal, J.B.: The theory of well-quasi-ordering: A frequently discovered concept. J. Comb. Theory A 13, 297–305 (1972)
14. Löb, M., Wainer, S.: Hierarchies of number theoretic functions, I. Arch. Math. Logic 13, 39–51 (1970)
15. McAloon, K.: Petri nets and large finite sets. Theor. Comput. Sci. 32, 173–183 (1984)
16. Moser, G., Weiermann, A.: Relating derivation lengths with the slow-growing hierarchy directly. In: Nieuwenhuis, R. (ed.) RTA 2003. LNCS, vol. 2706, pp. 296–310. Springer, Heidelberg (2003)
17. Schnoebelen, P.: Lossy counter machines decidability cheat sheet. In: Kučera, A., Potapov, I. (eds.) RP 2010. LNCS, vol. 6227, pp. 51–75. Springer, Heidelberg (2010)
18. Touzet, H.: Propriétés combinatoires pour la terminaison de systèmes des réécriture. Thèse de doctorat, Université de Nancy 1, France (September 1997)
19. Touzet, H.: A characterisation of multiply recursive functions with Higman's Lemma. Inform. and Comput. 178, 534–544 (2002)
20. Weiermann, A.: Complexity bounds for some finite forms of Kruskal's Theorem. J. Symb. Comput. 18, 463–488 (1994)