A General Constructive Form of Higman's Lemma

Stefano Berardi
Dipartimento di Informatica, Università di Torino, Italy

Gabriele Buriola
Dipartimento di Informatica, Università di Verona, Italy

Peter Schuster
Dipartimento di Informatica, Università di Verona, Italy

Abstract

In logic and computer science one often needs to constructivize a theorem $\forall f \exists g.P(f,g)$, stating that for every infinite sequence f there is an infinite sequence g such that $P(f,g)$. Here P is a computable predicate but g is not necessarily computable from f. In this paper we propose the following constructive version of $\forall f \exists g.P(f,g)$: for every f there is a "long enough" finite prefix g_0 of g such that $P(f,g_0)$, where "long enough" is expressed by membership to a bar which is a free parameter of the constructive version.

Our approach with bars generalises the approaches to Higman's lemma undertaken by Coquand–Fridlender, Murthy–Russell and Schwichtenberg–Seisenberger–Wiesnet. As a first test for our bar technique, we sketch a constructive theory of well-quasi orders. This includes yet another constructive version of Higman's lemma: that every infinite sequence of words has an infinite ascending subsequence. As compared with the previous constructive versions of Higman's lemma, our constructive proofs are closer to the original classical proofs.

2012 ACM Subject Classification Theory of computation → Constructive mathematics; Theory of computation → Proof theory; Mathematics of computing → Discrete mathematics

Keywords and phrases intuitionistic logic, constructive mathematics, formal proof, inductive predicate, bar induction, well quasi-order, Higman's lemma

Digital Object Identifier 10.4230/LIPIcs.CSL.2024.16

Acknowledgements The present study was started while the third author worked within the project "Reducing complexity in algebra, logic, combinatorics – REDCOM" belonging to the programme "Ricerca Scientifica di Eccellenza 2018" of the Fondazione Cariverona. The second and the third author are members of the "Gruppo Nazionale per le Strutture Algebriche, Geometriche e le loro Applicazioni" (GNSAGA) of the Istituto Nazionale di Alta Matematica (INdAM). The authors wish to thank Daniel Fridlender for his interest and suggestions, and to express their gratitude to Thierry Coquand, whose original idea set the basis for this work. Last but not least, the anonymous reviewers' meticulous reading and constructive critique have been extremely helpful.

1 Introduction: Higman's Lemma in Constructive Mathematics

By "Higman's lemma for sequences" we understand the following statement: "If Σ is a finite alphabet, then for every infinite sequence $\sigma = a_0, a_1, a_2, \ldots$ of words over Σ, there exists an infinite subsequence$\tau = a_{i_0}, a_{i_1}, a_{i_2} \ldots$ of σ such that $a_{i_0} \sqsubseteq a_{i_1} \sqsubseteq a_{i_2}, \ldots$, where $\sqsubseteq$ denotes the embedding order on words" (see later for more details). Given our focus on finite alphabet, Σ will always denote such a set. Although Higman's lemma for sequences has a well-known classical proof, i.e. with classical logic, one can easily check that it has no constructive proofs; the selection of the weakly increasing, **w.i.** for short, subsequence cannot be made recursively in general.

By simply taking the first two elements of any infinite w.i. subsequence, Higman's lemma for sequences entails that every infinite sequence $a_0, a_1, a_2, \ldots$ of words over a finite alphabet has a w.i. subsequence of length 2, i.e. there are $i_0 < i_1$ for which $a_{i_0} \sqsubseteq a_{i_1}$. This is

32nd EACSL Annual Conference on Computer Science Logic (CSL 2024).
Editors: Aniello Murano and Alexandra Silva; Article No. 16; pp. 16:1–16:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

one of the consequences of Higman's lemma for sequences that are provable already with intuitionistic logic: Murthy and Russell [15] applied combinatorial techniques over finite sequences; Richman and Stolzenberg [22] proved it for decidable well quasi-orders using hereditary (inductive) well-foundedness; Coquand and Fridlender [7] resorted to inductive definitions for the binary case and Seisenberger [24] extended the same approach to the general case; Schwichtenberg, Seisenberger and Wiesnet [23] extracted the computational content of Higman's lemma; and Powell [19] applied Gödel's Dialectica interpretation.

There are more examples along the same lines. We can ask for a w.i. subsequence of length k for any $k \in \mathbb{N}$ whatsoever, for which we take the first k elements of any w.i. infinite subsequence. But we do not have to stop here; for example, we can construct a non-empty w.i. subsequence of length $\texttt{len}(a_{i_0})+1$. In this case we take the first element of the w.i. infinite subsequence, compute its length $n = \texttt{len}(a_{i_0})$ as a word, and then take n more elements from the w.i. infinite subsequence. More generally, for every functional F from infinite sequences to $\mathbb{N}$ there is an infinite sequence σ having an infinite subsequence τ the first $F(\tau)$ elements of which are in weakly increasing order. We will deduce all these statements from an even stronger theorem, the Higman lemma for bars, which we will state and prove with intuitionistic logic.

The constructive history of Higman's lemma

The theory of well quasi-orders has found applications in many different fields, and so has Higman's lemma, one of this theory's milestones. Given the concrete character of Higman's lemma especially in the case of a finite alphabet, and its applicability in computer science, the search for a constructive and more perspicuous proof has started very early: not only to make possible program extraction from proofs, but also for a better understanding both of the original non-constructive proof of Higman's lemma and the short and elegant but still non-constructive proof by Nash-William [16]. To position the results of this paper in the literature, we now briefly survey the existing constructive approaches to Higman's lemma and related results such as Kruskal's theorem. For an historical survey of well quasi-order broadly understood we refer to [13].

The presumably first constructive proof of Higman's lemma was obtained by Murthy and Russell [15] using a smart manipulation of finite strings. Richman and Stolzenberg [22] then proved Higman's lemma by induction on subsets. Coquand and Fridlender [7] instead used structural induction over inductive definitions; their results were extended by Seisenberger [24]. Fridlender [9] gave a type-theoretic version of Higman's lemma, and Veldman [28] an inductive intuitionistic proof. Worthy of mention is Berger's constructive proof [3] of the equivalence between Dickson's lemma and Higman's lemma for a two-element alphabet.

The connection between Higman's lemma and programs has been addressed several times. Schwichtenberg, Seisenberger, and Wiesnet [23] analyzed the computational content of Higman's lemma. Powell has successfully applied Gödel's Dialectica interpretation to well quasi-orders [20] and Higman's lemma [19]. Concerning computer-assisted theorem proving, Berghofer [4], has formalized a constructive proof of Higman's lemma in Isabelle, starting from the article by Coquand and Fridlender; more recently, Sternagel [26] used open induction to obtain a proof in Isabelle/HOL.

Finally, also Kruskal's theorem [12], the natural extension of Higman's lemma from strings to finite trees, has been put under constructive scrutiny by Veldman [28] and Seisenberger [25], whereas Goubault-Larrecq [10] gave a topological constructive version of Kruskal's theorem.

A Constructive Form of Higman's Lemma – Now for Bars

Classically, a bar B for lists is a set of finite lists such that every infinite chain of one-step extensions meets B, i.e. has an element in B. Within intuitionistic logic we need and have a more perspicuous definition of bar (see later). Higman's lemma for bars now says: "for every bar B, every infinite sequence σ of words on a finite alphabet Σ has an infinite subsequence τ with a weakly increasing prefix τ_0 in B". Here we interpret $\tau_0 \in B$ as that τ_0 is "long enough for our purposes", with "our purposes" expressed by the choice of B.

E.g. if B is the set of lists of length 2 or more, then Higman's lemma for bars entails that every infinite sequence σ of words on Σ has an infinite subsequence τ with a weakly increasing prefix τ_0 in B, i.e. of two or more elements. This is the first of the desired consequences of Higman's lemma, all of which can be deduced from Higman's lemma for bars.

We will prove Higman's lemma for bars with intuitionistic logic. In fact, we prove a stronger version in which the requirement "σ is infinite" is replaced by one about the bar B. During our proof, we are able to constructively interpret several non-constructive classical theorems of the following form: for every sequence f there is an infinite sequence g such that $P(f, g)$. Higman's lemma for sequences is of course a typical case.

For instance, we rephrase the notion of wqo (well quasi-order), the main ingredient of the classical proof of Higman's lemma, by quantification on bars; and call "wqo(bar)" this novel notion of wqo. We have short proofs with intuitionistic logic that the concept wqo(bar) is closed by unions (provided that the union is a preorder), by products and by right-invertible morphisms; these are all properties of wqo typically occurring in a classical proof of Higman's lemma. With the notion of wqo(bar) at hand, we consider possible to develop a constructive version of the theory of wqo close to the classical one.

The structure of the article is as follows. In Section §2 we briefly recall the words and sequences terminology; Section §3 is devoted to prove several properties of bars used in the rest of the paper; in Section §4 we state Higman's lemma for bars and prove the desired corollaries; in Section §5, after some closure properties of the concept wqo(bar), we prove Higman's lemma for bars.

2 Lists, Words and Sequences

We start by recalling some well-known terminology about lists, sublists and labels (§2.1), as well as notions related to alphabets and words (§2.2). This part is straighforward, and the reader may want to jump to the subsection about the anticone of a word (§2.3).

2.1 Lists and Operations on Lists

Let $\mathbb{N}$ be the set of natural numbers and I a given set. We call a *list* l on I any map l such that $l : [0, n[\to I$ for some $n \in \mathbb{N}$ or $l : \mathbb{N} \to I$. We set $\mathtt{dom}(l) = [0, n[$, $\mathtt{range}(l) = l([0, n[)$ in the first case and $\mathtt{dom}(l) = \mathbb{N}$, $\mathtt{range}(l) = l(\mathbb{N})$ in the second case; moreover, we call $\mathtt{dom}(l)$ the set of indexes of l and $\mathtt{range}(l)$ the set of elements of l, abbreviating $i \in \mathtt{range}(l)$ by $i \in l$. The length of l, denoted $\mathtt{len}(l)$, is $n \in \mathbb{N}$ in the first case and is ∞ (infinite) in the second case; in the first case we say that the list l is finite and in the second case that l is infinite. We call each $x \in \mathtt{range}(l)$ an element of l and write $\mathtt{Fin}(I)$ for the set of finite lists on I, $\mathtt{Inf}(I)$ for the set of infinite lists, and $\mathtt{List}(I) = \mathtt{Fin}(I) \cup \mathtt{Inf}(I)$ for the set of all lists on I.

We write a finite list $l \in \mathtt{Fin}(I)$ of length $n = \mathtt{len}(l)$ as $\langle l(0), \dots, l(n-1)\rangle$, denoting by $\mathtt{Nil} = \langle\rangle$ the empty list, the unique list of length 0. For $l, m \in \mathtt{List}(I)$, we write $l \sqsubseteq m$, or "l is a sublist of m" for: there is a finite increasing list $f : [0, \mathtt{len}(l)[\to [0, \mathtt{len}(m)[$ of natural numbers such that $l(i) = m(f(i))$ for all $i \in [0, \texttt{len}(a)[$; we call such an f an *embedding* of l in m. For instance, if I is the English alphabet, if $l = \langle w, o, r, d\rangle$, representing the word "word", and $m = \langle w, o, r, l, d\rangle$, representing the word "world", then $l \sqsubseteq m$. An embedding of l in m is $f : [0, 4[\to [0, 5[$ defined by $f(0) = 0, f(1) = 1$, $f(2) = 2$ and $f(3) = 4$. $\texttt{range}(f)$ does not include 3, the index of the symbol "l" in "world". Another example: $l : \mathbb{N} \to \mathbb{N}$ defined by $a(i) = 2i$ is the list of all even numbers, $m : \mathbb{N} \to \mathbb{N}$ defined by $m(i) = i$ is the list of all natural numbers, and $f(i) = 2i$ is an embedding from l to m. Roughly speaking, we have $l \sqsubseteq m$ if and only if we can obtain l by skipping zero or more elements from m, without changing the order of the elements of b.

If I, J are sets, then $I \times J = \{\langle x, y\rangle \mid x \in I, y \in J\}$ denotes their Cartesian product. A *binary relation* R on I, J is given by a subset of $I \times J$ and we write R^{-1} for the inverse binary relation $\{\langle y, x\rangle \in J \times I \mid \langle x, y\rangle \in R\}$, using the notation $R(x, y)$ or xRy for $\langle x, y\rangle \in R$. If X is a set, then the *R-upward cone* of X, denoted $R(X)$, is the set of $y \in J$ such that $\exists x \in X.R(x, y)$; often abbreviating "upward cone" by "cone". If $x \in I$, we write $R(x)$ for $R(\{x\})$, and we call $R(x)$ the R-cone of x in J; moreover, we call $J \setminus R(x)$ the *anticone* of x in J: it is the cone of x with respect to the complement in $I \times J$ of the relation R.

We write $\sqsubseteq_{I,J}$ for the binary relation $\{\langle l, m\rangle \in \texttt{Fin}(I) \times \texttt{Fin}(J) \mid l \sqsubseteq m\}$ defined by the sublist predicate restricted to $\texttt{Fin}(I) \times \texttt{Fin}(J)$ and we write $\sqsubseteq_I$ for $\sqsubseteq_{I,I}$ and $\sqsupseteq_{I,J}$ for the inverse binary relation $\{\langle l, m\rangle \in \texttt{Fin}(I) \times \texttt{Fin}(J) \mid l \sqsupseteq m\}$

For $i \in \mathbb{N}$, we define the restriction $l\lceil i \in \texttt{Fin}(I)$ of l to $[0, i[$ by $(l\lceil i)(j) = l(j)$ for all $j < \texttt{len}(l), j < i$ and $\texttt{len}(l\lceil i) = \min(\texttt{len}(l), i)$. When l is a restriction of some (possibly infinite) list m, then we say that l is a *prefix* of m and we write $l \leqslant m$.

We define the concatenation $m = l\star l'$ of two lists l, l' with l finite by: $m(i) = l(i)$ for all $i < \texttt{len}(l)$ and $m(\texttt{len}(l) + j) = l'(j)$ for all $j < \texttt{len}(l')$. By definition we have $\langle l(0), \ldots, l(n-1)\rangle\star\langle l'(0), \ldots, l'(m-1), \ldots\rangle = \langle l(0), \ldots, l(n-1), l'(0), \ldots, l'(n'-1), \ldots\rangle$. The length of m is $\texttt{len}(l) + \texttt{len}(l')$ if l' is finite and ∞ if l' is infinite. If $m = l\star l'$ for some $l, l' \in \texttt{Fin}(I)$, then we say that l' is a *suffix* of m.

Given $l, m \in \texttt{Fin}(I)$, we write $l <_1 m$ if $m = l\star\langle i\rangle$ for some $i \in I$, i.e. m is obtained by adding the element i to the end of l. We write $<$ for the transitive closure of the relation $<_1$. Remark that the prefix relation $\leqslant$ is the reflexive closure of $<$.

2.2 Alphabet, Words and Sequences

A finite alphabet Σ is any finite set Σ in bijection with $[0, n[$ for some n and through some map f. Equality on Σ is provably decidable with intuitionistic logic, because $i = j$ in Σ if and only if $f(i) = f(j)$ in $\mathbb{N}$. We call the elements of Σ "symbols" of the alphabet, and we denote them with the letters a, b, c and their variants, $a', a_1, \ldots$ the basic example is $\Sigma = \{0, 1\}$. A *word* on Σ is given by a finite list on Σ and we write $\Sigma^* = \texttt{Fin}(\Sigma)$ for the set of words on Σ. We use `nil` for the empty word in Σ^*, this is just another name for $\texttt{Nil} = \langle\rangle$, and we denote words with the letters v, w, z and their variants, $v', v_1, \ldots$; moreover, with a slight and harmless abuse of notation, we use the expression $c \in w$ and $c \notin w$ to denote respectively that c is, or is not, one of the letters of w. A "symbol" could be anything, therefore we could use "finite set" for "alphabet" and "finite list on a finite set" for word, but it is customary to use "alphabet" and "word" in the context of Higman's lemma, because the intended application of the Lemma are the words of an alphabet. If $v, w \in \Sigma^*$, when $v \sqsubseteq w$ we say that v is a subword of w and w a superword of v.

We introduce abbreviations used only for words. If $c_1, \ldots, c_n \in \Sigma$, we abbreviate the word $w = \langle c_1, \ldots, c_n\rangle$ with $w = c_1 \ldots c_n$, written without spaces inside. If $v, w \in \Sigma^*$, we abbreviate $v\star w$ with the juxtaposition vw. If $c \in \Sigma$ and $w \in \Sigma^*$, we abbreviate $\langle c\rangle\star w$ by cw, and $w\star\langle c\rangle$ by wc.

We call a "sequence of words" on Σ, just a "sequence" for short, any list on Σ^*. A sequence is finite if it is a finite list and it is infinite if it is an infinite list. Again, we could use "list of words" as well, but it is customary to say "sequence". Within this terminology, $\texttt{Fin}(\Sigma^*)$ and $\texttt{Inf}(\Sigma^*)$ are the set of finite and infinite sequences on Σ^*. Finally, we adopt the following notation rule: finite sequences are denoted by Latin letters, whereas infinite sequences by Greek letters.

2.3 Anticone and Slice of a Word

In this subsection we characterize the words which are superwords of a given word and those which are not. Let us fix $v \in \Sigma^*$. We recall that $\not\sqsubseteq_\Sigma(v)$ denotes the anticone of v, which is the set of all $w \in \Sigma^*$ such that $v \not\sqsubseteq w$. The first step in our proof of Higman's lemma is to characterize the words in the anticone of v. To this aim, we need one preliminary step. We introduce a smaller set of words $\texttt{Slice}_\Sigma(v) \subseteq \Sigma^*$, dubbed the *slice* of v, consisting of all words $w \in \Sigma^*$ for which v is minimal among the words not embeddable in w.

▶ **Definition 1** (Slice of v)**.** *For each word $v \in \Sigma^*$ we define $\texttt{Slice}_\Sigma(v)$ as the set of words in Σ^* which are superwords of all $v' < v$, but are not superwords of v.*

We characterize the words in $\texttt{Slice}_\Sigma(v)$. We have $\texttt{Slice}_\Sigma(\texttt{nil}) = \emptyset$, because all words are superlists of $\texttt{nil}$. Assume that $v = c_0 \dots c_{k-1}$ is not empty, that is, that $k \geqslant 1$. Then by definition unfolding we have:

$$\texttt{Slice}_\Sigma(v) = \{w \in \Sigma^* \mid (c_0 \dots c_{k-2} \sqsubseteq w) \wedge (c_0 \dots c_{k-1} \not\sqsubseteq w)\}$$

To say otherwise, $\texttt{Slice}_\Sigma(v) = \sqsubseteq_\Sigma(c_0 \dots c_{k-2}) \cap \not\sqsubseteq_\Sigma(c_0 \dots c_{k-1})$, which is the set of words in Σ^* which are superwords of $c_0 \dots c_{k-2}$ but not of $c_0 \dots c_{k-1}$. We provide a detailed description of words in $\texttt{Slice}_\Sigma(v)$. Let us abbreviate $\Sigma_i = \Sigma \setminus \{c_i\}$: then Σ_i^* is the set of $w \in \Sigma^*$ such that $c_i \notin w$. We will prove that the words in $\texttt{Slice}_\Sigma(v)$ are exactly all words of the form $w = w_0 c_0 w_1 c_1 \dots c_{k-2} w_{k-1}$, such that $c_i \notin w_i$, that is, such that $w_i \in \Sigma_i^*$, for all $i < k$. Such a decomposition will be unique and, for all $i < k$, it will define a map α_i such that $w_i = \alpha_i(w)$. We first prove that we have a slightly different decomposition for the words of the cone of v, then we prove the required decomposition for the words of $\texttt{Slice}_\Sigma(v)$.

▶ **Lemma 2** (Characterization of cone and of slice)**.** *Let $v = c_0 \dots c_{k-1}$, $w \in \Sigma^*$.*

1. Cone*. If v is embedded in w through f, then there is a unique decomposition $w = w_0 c_0 w_1 c_1 \dots w_{k-1} c_{k-1} w_k$, such that $c_i \notin w_i$, for all $i < k$. We have no requirement for w_k. Furthermore, if*

   $$g(i) = \texttt{len}(w_0 c_0 w_1 c_1 \dots c_{i-1} w_i)$$

   for all $i < k$, then g is the minimum embedding of v in w in the point-wise ordering: $g(i) \leqslant f(i)$ for all $i < k$.
2. Slice*. If $k \geqslant 1$, then $\texttt{Slice}_\Sigma(v)$ is the set of all words w such that $w = w_0 c_0 \dots w_{k-2} c_{k-2} w_{k-1}$ and $c_i \notin w_i$ for all $i < k$. The decomposition of w if it exists it is unique.*

From the uniqueness of the decomposition of $w \in \texttt{Slice}_\Sigma(v)$ we define the maps $\alpha_i(w)$ for $i < \texttt{len}(v)$.

▶ **Definition 3** (The maps α_i)**.** *Assume that $v = c_0 \dots c_{k-1}$, $k \geqslant 1$ and $i \in \mathbb{N}$, $i < k$. Let us abbreviate $\Sigma_i = \Sigma \setminus \{c_i\}$. Assume that $w = w_0 c_0 \dots w_{k-2} c_{k-2} w_{k-1}$ and $c_i \notin w_i$ for all $i < k$. We define $\alpha_i : \texttt{Slice}_\Sigma(v) \to \Sigma_i^*$ by $\alpha_i(w) = w_i$.*

If X and Y are sets with binary relations R and S, respectively, then by a *morphism* $f : (X, R) \to (Y, S)$ we understand a map $f : X \to Y$ such that if xRx', then $f(x)Sf(x')$.

The "product" α of the α_i in Def. 3 defines a bijection, whose inverse is a morphism for $\sqsubseteq$; the map α plays a crucial role in the proof of Higman's lemma.

▶ **Lemma 4** (Product map and Slices). *The product map* $\alpha = \alpha_1 \times \ldots \times \alpha_k : \mathtt{Slice}_\Sigma(v) \to \Sigma_0^* \times \ldots \times \Sigma_{k-1}^*$*, defined as* $\alpha(w) = (\alpha_0(w), \ldots, \alpha_{k-1}(w))$*, is a bijection. Its inverse* α^{-1} *is a morphism from* $(\Sigma_0^* \times \ldots \times \Sigma_{k-1}^*, \sqsubseteq \times \ldots \times \sqsubseteq)$ *to* $(\mathtt{Slice}_\Sigma(v), \sqsubseteq)$.

Now we can characterize the anticone $\not\sqsupseteq_\Sigma(v)$ as a finite union of slices $\mathtt{Slice}_\Sigma(v')$.

▶ **Lemma 5** (Anticone). $\not\sqsupseteq_\Sigma(v)$ *is the union of all* $\mathtt{Slice}_\Sigma(v')$ *for* $v' \leqslant v$.

These are all the properties we need about words, for what concerns bars we refer to the next section.

3 Bars: Definition and Properties

In this section we define bars and their related notions, proving with intuitionistic logic the properties required in the rest of the paper. The strongest property says that the Cartesian product of barred sets is barred by the union of the inverse image of the two projections. It is worth noticing that if we consider the empty bar, then from each result in this section about bars (except for "monotonicity", which only makes sense for bars) we obtain some well-known result about well-founded sets.

3.1 Quasi-orders, Labels, Well-founded Relations and Bars

A *quasi-order* $(P, \leqslant)$ is a set P with a transitive and reflexive relation $\leqslant$; a quasi-order $(P, \leqslant)$ is a *partial order* if $\leqslant$ is antisymmetric. A sequence $(p_k)_k$, finite or infinite, over $(P, \leqslant)$ is *weakly increasing*, for short w.i., if, for every indices $i \leqslant j$, we have $p_i \leqslant p_j$.

A *labelling* of I on P is a map $\varphi : I \to P$. A length n list $l = \langle l(0), \ldots, l(n-1)\rangle \in \mathtt{Fin}(I)$ can be turned into a list $\varphi l = \langle \varphi l(0), \ldots, \varphi l(n-1)\rangle \in \mathtt{Fin}(P)$ on P, by composing with the labelling φ of I. When $I = P$, we also consider the identical label $\varphi = \mathtt{id}$, in which the list of labels of a list is the list itself. We write $\mathtt{Incr}(\leqslant, \varphi, I)$ for the set of finite lists $l \in \mathtt{Fin}(I)$ such that φl is a weakly increasing list in P with respect to $\leqslant$.

We say that $B \subseteq \mathtt{Fin}(I)$ is $<_1$*-closed*, or closed by one-step extension, if for all $l \in B$, $la <_1 m$ we have $m \in B$. Being closed by one-step extension is the same than being closed by $\leqslant$ (by extension).

We define now the notions of a (hereditarily) well-founded set (see for instance [14,17,18]) and of a barred set, both given with respect to a given binary relation R. Our definitions are classically equivalent to the definition "all R-decreasing sequences intersect the bar" but in intuitionistic logic they allow to derive more results. Our bars generalize Troelstra's definition of bar ([27], page 77, Def. 1.9.20).

We notice that the word *inductive* is often used as a synonimous of *hereditary*.

▶ **Definition 6** (Well-founded and Barred Sets). *Let P, X, B be sets and R be a binary relation.*

1. *P is X, R-*hereditary *whenever, for all $x \in X$, if for all $x' \in X$ with $x'Rx$ we have $x' \in P$, then $x \in P$.*
2. *X is R-*well-founded *if for all P X, R-hereditary such that $P \subseteq X$ we have $P = X$.*
3. *B* bars *X, R if for all P X, R-hereditary such that $B \cap X \subseteq P \subseteq X$ we have $P = X$.*
4. *B* bars *x in X, R if for all P X, R-hereditary such that $B \cap X \subseteq P \subseteq X$ we have $x \in P$.*

Some comments on these definitions are in order. We already stressed that "X, R-hereditary" is exactly "X, R-inductive". This is the word chosen for instance in [2]. Next, we remark that B bars X, R if and only if B bars x in X, R for every $x \in X$.

In general, the subset consisting of the $x \in X$ such that B bars x in X, R is defined as the intersection of all X, R-hereditary $P \subseteq X$ such that $B \cap X \subseteq P$; and one can easily check that this intersection itself is X, R-hereditary. Hence "B bars x in X, R" coincides with the predicate $B \cap X \mid x$ from [7,8]: that is, the least X, R-hereditary predicate on X which contains $B \cap X$. B is often called the *inductively defined predicate* from X, R.

So "B bars x in X, R" can be interpreted as "x is accessible from B in X, R": for $B = \emptyset$ this is nothing but the *accessibility predicate* from [5]. Accordingly, X is R-well-founded if and only if X, R is barred by $B = \emptyset$, or barred by any B such that $B \cap X = \emptyset$.

In Troelstra ([27], page 77, Def. 1.9.20) the definition of bar is given with $X =$ the set of all lists of natural numbers and $R =$ the one-step extension; it is also assumed that B is either decidable or closed by extensions. But the main difference is that the definition of bar is given as in classical mathematics, B is a bar if all infinite lists of natural numbers have some prefix in B. Instead, we defined B as the intersection of all X, R-hereditary properties, since we find this version more suitable for constructive proofs; this is the typical definition in the context of generalized inductive definitions [1,21].

In the case we do not mention it, by R we mean $>_1$, the reverse of the one-step extension relation. In this case we say that B bars l in X, respectively that B bars X, meaning that B bars l in $X, >_1$, respectively that B bars $X, >_1$.

A subset B of X is said to be *R-downward-closed* if, for all $x \in B$, if yRx, then $y \in B$. We have a puzzling point to stress here, if $R =>_1$, then R-downward-closed in fact means that for all $x \in B$ if $y >_1 x$, then $y \in B$. That is, "R-downward-closed" in this case means "closed by one-step *extensions*". The reason is that in the literature, set of lists are often used to represent trees, and in the case of trees, it is customary to consider "smaller" a one-step-extension of a node of a tree, i.e. downward trees. We will still use the word "downward-closed" in this case, because it is a well-established terminology for inductive reasoning, but we will point out that "downward-closed" in this case means "closed by one-step extensions".

A last warning. In our definition, bars for set of lists *do not have to be closed by extensions*. For instance, the set B of all finite lists on I having odd length is a bar for the set of all lists on I and $>_1$, because each list is either odd and barred by B, or has all one-step extensions odd and barred B, and in this case is barred because being barred is an hereditary predicate. Yet, each one-step extension of a list in B is some even length list, which is not in B. Closure of a bar for a set of lists by list extension is an useful feature in some proofs, nevertheless it is not strictly required in most cases.

3.2 Basic Properties of Bars

In this subsection, we derive some basic properties for bars, requiring little more than definition unfolding.

An *R-descending chain* in X is a finite or infinite list $x_0 R^{-1} x_1 R^{-1} x_2 R^{-1} \ldots$ of elements of X. For instance, a $<$-descending chain in $\mathbb{N}$ is any (necessarily finite) list $x_0 > x_1 > x_2 > \ldots$ of natural numbers. We will prove that if B bars X, R, then every infinite R-descending chain in X intersects B. Using classical Logic, and some choice, the two properties are equivalent; but within intuitionistic logic, we only have the implication from the former to the latter.[1]

[1] We sketch a folk-lore proof. There is a model of Intuitionistic Logic in which all chain are recursive, while some order $<$ on some X has all infinite recursive $<$-descending chain finite and some non-recursive

▶ **Proposition 7** (Infinite R-descending chains). *Let X, B be sets and R be a binary relation.*
1. *X is X, R-hereditary.*
2. *The intersection $\cap\mathcal{F}$ of any inhabited family $\mathcal{F}$ of X, R-hereditary sets is X, R-hereditary.*
3. *the predicate "B bars x in X, R" on $x \in X$ is between $B \cap X$ and X and it is itself X, R-hereditary.*
4. *If B bars X, R, then every infinite R-descending chain in X intersects B in an infinite set of indexes.*

If B bars X, R, then we can prove that a property $P \subseteq X$ holds for all $x \in X$ by bar-induction on B, X, R. Bar-induction is the following principle. Assume that $P \subseteq X$ and: *(i. base case)* for all $x \in B \cap X$ we have $x \in P$; *(ii. inductive case)* if for all yRx, $y \in X$ we have $y \in P$, then $x \in P$. Then we conclude that $P = X$. As an example, Proposition 7.4 is proved by bar-induction on B, X, R.

We give an interpretation of a proof by bar-induction of some property P on X. We have to think of $B \cap X$ as the set of elements for which we can prove the property P directly. The one-step extension yRx of a sequence x are all elements "smaller" than x and in the inductive step of bar-induction, we have proved that if all elements "smaller" than an element x are in P, then x is in P. Eventually, if B bars X, R, then we conclude that $P = X$.

A tool for proving that B bars X, R is the notion of "simulation". We say that x' is an R-predecessor of x if $x'Rx$. Roughly speaking, $V \subseteq X \times Y$ is a simulation between X, R and Y, S if whenever two elements are related by V, then any R-predecessor of the first element is V-related with some S-predecessor of the second element.

▶ **Definition 8.** *We say that $V \subseteq X \times Y$* simulates *X, R in Y, S if for all $x, x' \in X$, $y \in Y$, if $x'Rx$ and xVy, then there is some $y' \in Y$, $y'Sy$ such that $x'Vy'$.*

We will prove that a simulation V, when V is everywhere defined, moves bars backwards from Y to X. By this we mean: if B bars Y, S, then $V^{-1}(B)$ bars X, R. In particular, simulation moves well-foundedness backwards: if we take $B = \emptyset$, we obtain that if Y is S-well-founded then X is R-well-founded. We will prove the same result for *morphisms*; that is, if $f : X \to Y$ maps pairs related by R into pairs related by S, then f^{-1} maps bars for Y, S into bars for X, R.

▶ **Lemma 9** (Simulation Lemma). *Let X, Y, B, C be sets and R, S be binary relations.*
1. (simulation) *Assume that $V \subseteq X \times Y$ simulates X, R in Y, S, that V is everywhere defined, i.e., for every $x \in X$ there exists $y \in Y$ such that xVy, and that C bars Y, S; then $B = V^{-1}(C)$ bars X.*
2. (morphism) *Assume that $f : X, R \to Y, S$ is a morphism and C bars Y; then $f^{-1}(C)$ bars X.*

Now we prove that, if we extend a bar and we reduce the barred set and the relation, then the fact of being a bar is preserved. Choosing the empty bar, we obtain a well-known result for well-founded relations, namely well-foundedness is preserved by moving to a subrelation. To say otherwise: if X is R-well-founded, with $Y \subseteq X$ and $S \subseteq R$, then Y is S-well-founded.

▶ **Lemma 10** (Monotonicity and Antimonotonicity). *Let X, Y, B, C be sets, and R, S binary relations.*
1. (monotonicity) *If B bars X, R and $B \cap X \subseteq C \cap X$, then C bars X, R.*
2. (antimonotonicity) *If B bars X, R and $Y \subseteq X$, $S \subseteq R$, then B bars Y, S.*

---

infinite $<$-descending chain infinite, with set of elements C. In this model all infinite $<$-descending chain in X intersects $\emptyset$, because no infinite $<$-descending chain exists. Yet, the set $P = X \setminus C$ is $X, <$-hereditary while $P \neq X$. Thus, it is not true that $\emptyset$ bars X, R.

For every family of sets Y_x indexed by $x \in X$ we write $\Sigma_{x\in X} Y_x$ for the set of pairs (x, y) such that $x \in X$ and $y \in Y_x$.

Now let R be a binary relation, and $S = \{S_x\}_{x\in X}$ an indexed family of binary relations on Y. We can think of S as a ternary relation such that $S(x, y', y) \Leftrightarrow S_x(y', y)$ for all $x \in X$ and $y', y \in Y$. The *lexicographic product* $R \times S$ is the relation comparing (x', y') with (x, y) according to xRx', or, if $x = x'$, according to yS_xy'. Formally:

$$(x', y')(R \times S)(x, y) \quad \Leftrightarrow \quad x'Rx \vee (x' = x \wedge y'S_xy).$$

$R \times S$ is a partial order if R and all S_x are partial orders, in this case $R \times S$ is called the lexicographic order on pairs.

Assume that the dependency on $x \in X$ is trivial, that is, for some Z, T and for all $x \in X$ we have $Y_x = Z$, $S_x = T$. In this case we write $R \times T$ for $R \times S$. By definition unfolding, $R \times T$ is a relation on $\Sigma_{x\in X} Y_x = X \times Z$ defined by $(x', y')(R \times T)(x, y) \Leftrightarrow x'Rx \vee (x' = x \wedge y'Ty)$.

With the next lemma we define a bar D for $\Sigma_{x\in X} Y_x$, $R \times S$. When the dependency on $x \in X$ is trivial, D is a bar for $X \times Z$, $R \times T$. Our result generalises [14, Chapter I, Theorem 6.3], which is, in our terminology, the special case when D is the empty bar.

▶ **Lemma 11** (Lexicographic Product). *Let X, Y, B and C_x for $x \in X$ be sets, R a binary relation and S a ternary relation. Suppose that B bars X, R, and that C_x bars Y_x, S_x for all $x \in X$. Let D be the set of all pairs $(x, y) \in \Sigma_{x\in X} Y_x$ such that $x \in B$ or $y \in C_x$.*

1. *D bars $\Sigma_{x\in X} Y_x$ with $R \times S$, the lexicographic product of R, S.*
2. *If for some Z, T and for all $x \in X$ we have $Y_x = Z$, $S_x = T$, then D bars $X \times Z$, $R \times T$.*

4 Higman's Lemma for Bars

In this section, we state Higman's lemma for bars, which is a constructive version of Higman's lemma for subsequences, and we argue why this version is stronger with intuitionistic logic than the versions proposed until now.

Let $(P, \leqslant)$ be a partial order. For a given labelling $\varphi : I \to P$, we recall that we write $\mathtt{Fin}(I)$ for the set of finite lists in I and $\mathtt{Incr}(\varphi, I)$ for the subset $\mathtt{Incr}(\leqslant, \varphi, I)$ of $\mathtt{Fin}(I)$ consisting of the finite lists ℓ in I such that $\varphi\ell$ is a weakly increasing list on P for the order $\leqslant$. We can now introduce the constructive version wqo(bar) of the notion of wqo.[2] A quasi-order $(P, \leqslant)$ is wqo(bar) if for every set $X \subseteq \mathtt{Fin}(I)$, a bar B for the subset of X consisting of all w.i. lists (those in $\mathtt{Incr}(\varphi, I)$) is a bar for the whole of X, provided that X is closed by sublists and B by superlists.

▶ **Definition 12** (Well quasi-order with bars). *A quasi-order $(P, \leqslant)$ is called* wqo(bar) *if*

$$B \text{ bars } X \cap \mathtt{Incr}(\varphi, I) \implies B \text{ bars } X \tag{1}$$

for all labellings $\varphi : I \to P$ of I by P, for every subset $X \subseteq \mathtt{Fin}(I)$ closed by I-sublists and for every subset $B \subseteq \mathtt{Fin}(I)$ closed by I-superlists.

As before, "B bars ..." is meant for the converse $>_1$ of the one-step extension order $<_1$ on $\mathtt{Fin}(I)$.

Classically, (1) means that every infinite $<_1$-increasing chain $\sigma : \mathbb{N} \to X$ meets B if this is the case already for any such σ for which in addition we have $\varphi\sigma(0) \sqsubseteq \varphi\sigma(1) \sqsubseteq \dots$ Within classical logic, condition (1) is equivalent to the more commonly used notion of wqo(set): for every infinite list $\sigma : \mathbb{N} \to \Sigma^*$ there is a an infinite $\sqsubseteq$-weakly increasing sublist $\tau : \mathbb{N} \to \Sigma^*$.

[2] For a constructive comparison of the customary concepts of wqo we refer to [6].

We focus on partial orders $P = \Sigma^*$, given by the set of words for a finite alphabet Σ, with the subword order $\sqsubseteq$ as $\leqslant$, and we prove that:

▶ **Theorem 13** (Higman's lemma for bars). *If Σ is a finite alphabet, then Σ^* is a wqo(bar).*

We postpone the proof of Theorem 13 to §5. In the rest of this section we derive with intuitionistic logic some corollaries of Theorem 13, in order to show the interest from an constructive viewpoint of stating the result in this form.

Our corollaries are about functionals. We add a bottom element $\perp$ to $\mathbb{N}$, then we consider partial and total continuous functional $F\colon \mathtt{Inf}(\Sigma^*) \to \mathbb{N} \cup \{\perp\}$ on infinite sequences of words. We take the canonical topology on $\mathtt{Inf}(\Sigma^*) \to \mathbb{N} \cup \{\perp\}$.[3] F maps infinite sequences of words in $\mathbb{N} \cup \{\perp\}$. F continuous means that F, when convergent, uses only a finite part of its input. Informally, a partial functional F explores larger and larger finite prefixes of an infinite sequence of words, until F finds a prefix long enough to compute some $n \in \mathbb{N}$. Formally, we define F as a map on *finite* lists, which can return the bottom element $\perp$, and if it returns $n \in \mathbb{N}$ on a finite list l then returns the same n on all extensions of l. If σ is infinite, then $F(\sigma) = n$ if and only if $F(l) = n$ for some finite prefix l of σ. Classically, F is called "total" if F returns some $n \in \mathbb{N}$ on all infinite lists. In order to make possible proofs with intuitionistic logic, we define totality through a bar instead.

▶ **Definition 14.**
1. *The strict order $\prec$ on $\mathbb{N} \cup \{\perp\}$ is defined by $\perp \prec n$ for all $n \in \mathbb{N}$ and no comparison between two natural numbers. $\preceq$ is the associated weak order.*
2. *A* partial continuous functional *is a map $F\colon \mathtt{Fin}(\Sigma^*) \to \mathbb{N} \cup \{\perp\}$ which is monotone with respect to the prefix order $\leqslant$ and $\preceq$.*
3. *A partial continuous functional F is* (bar-)total *if $F^{-1}(\mathbb{N})$ bars $\mathtt{Fin}(\Sigma^*)$.*
4. *If F is a total continuous functional, then its* canonical extension *to all $\sigma \in \mathtt{Inf}(\Sigma)$ is given by $F(\sigma) = n$ if for some finite prefix l of σ, we have $F(l) = n$.*[4]

▶ **Proposition 15.** *If F is bar-total and $\sigma \in \mathtt{Inf}(\Sigma)$, then $F(\sigma)$ exists, it is in $\mathbb{N}$ and it is unique.*

Proof. . From $F^{-1}(\mathbb{N})$ bar of $\mathtt{Fin}(\Sigma^*)$ and Lemma 7.4, every infinite list σ has some finite prefix l in the bar $F^{-1}(\mathbb{N})$, therefore $F(\sigma) = F(l) = n \in \mathbb{N}$ for some $n \in \mathbb{N}$. The value n is unique: if $F(\sigma) = F(l') = n' \in \mathbb{N}$ for another finite prefix of σ, then either $l \leqslant l'$ or $l' \leqslant l$, therefore $F(l) \preceq F(l')$ or $F(l') \preceq F(l)$, that is, $n \preceq n'$ or $n' \preceq n$. In both cases we conclude $n = n'$. ◀

Thus, if F is bar-total, then F is "total" with the usual classical definition: F returns some $n \in \mathbb{N}$ on all infinite lists. Classically, the reverse implication holds, but with intuitionistic logic bar-total is a stronger property.[5] From now on, by "total" we will always mean bar-total.

---

[3] For any $l \in \mathtt{Inf}(\Sigma^*)$, we define $O_l = \{m \in \mathtt{Inf}(\Sigma^*) \mid l \leqslant m\}$; we then take on $\mathbb{N}$ the discrete topology, on $\mathtt{Inf}(\Sigma^*)$ the topology generated by the sets O_l with $l \in \mathtt{Inf}(\Sigma^*)$ and the function topology on $\mathtt{Inf}(\Sigma^*) \to \mathbb{N} \cup \{\perp\}$.

[4] The idea is that we can approximate an element of $\mathtt{Inf}(\Sigma)$ considering all its initial segments which are elements of $\mathtt{Fin}(\Sigma^*)$.

[5] We claim that there is some recursive functional F which is defined on all recursive sequences, but returning $\perp$ on some non-recursive sequence. The proof uses the folk-lore result there is some recursive tree, whose recursive branches are all finite, but having some infinite non-recursive branch.

Let us fix a total functional F and a finite alphabet Σ. Higman's lemma for subsequences implies that for every infinite list σ over Σ^*, there is an infinite sublist $\tau \sqsubseteq \sigma$ whose first $F(\tau)$ elements are in w.i. order. Classically, it is enough to take any infinite w.i. sub-list τ of σ and then a finite prefix l of $F(\tau)$ elements. We call "F-long" the prefix of τ with $F(\tau)$-elements.

Informally speaking, this result means that we can provide infinite sublists τ having a w.i. prefix of any given length, with the length $F(\tau)$ we require described by some bar-total continuous functional F applied to the very sublist τ we are defining. We can provide a proof with intuitionistic logic of this result as an immediate corollary of Higman's lemma for bars.

▶ **Corollary 16** (sublists with an F-long w.i. prefix). *Let Σ be a finite alphabet and $F : \mathtt{Fin}(\Sigma^*) \to \mathbb{N} \cup \{\bot\}$ a bar-total continuous functional. Then every infinite sequence of words $\sigma \in \mathtt{Inf}(\Sigma^*)$ has an infinite subsequence τ with the first $F(\tau)$ elements in w.i. order, i.e. such that τ has an F-long w.i. prefix.*

Proof. Let $\varphi = \mathrm{id}_I$ where $I = \Sigma^*$. Set $X_0 = \mathtt{Incr}(\varphi, I)$ and $X = \mathtt{Fin}(I)$. Let $B_0 = \{\rho \in X \,|\, F(\rho) \in \mathbb{N}\}$. By the hypotheses on F, this B_0 bars X, and is upwards closed in X for the prefix order $\leqslant$. By the antimonotonicity of bars (Lemma 10.2), B_0 also bars $X_0 \subseteq X$.

Let $B_1 = \{\rho \in B_0 \cap X_0 \mid \mathtt{len}(\rho) \geqslant F(\rho)\}$. *Claim:* B_1 bars X_0. To prove this, set $P = \{\rho \in X_0 \mid B_1 \text{ bars } \rho\}$. Then the Claim means $P = X_0$, which we show by bar induction with the bar B_0 for X_0. Since P is hereditary (Proposition 7), which is the induction step, we only need to verify the base case $B_0 \cap X_0 \subseteq P$. To this end we show $\rho \in P$ for all $\rho \in B_0 \cap X_0$ by induction on $f(\rho) = \max(0, F(\rho) - \mathtt{len}(\rho))$.

Case $f(\rho) = 0$: Then $F(\rho) \leqslant \mathtt{len}(\rho)$ and thus $\rho \in B_1 \subseteq P$.

Case $f(\rho) = n + 1$: For every $\rho' \in X_0$ with $\rho <_1 \rho'$ we have $\mathtt{len}(\rho') = \mathtt{len}(\rho) + 1$, and $F(\rho) = F(\rho')$ by continuity, so $f(\rho') = n$. In addition, $\rho' \in B_0$ (because $\rho \in B_0$ and B_0 is upwards closed for $\leqslant\, \supseteq\, <_1$); whence $\rho' \in P$ by induction. As P is hereditary, $\rho \in P$ follows.

This ends the proof of the Claim.

Now let $B = \{\rho \in X \mid \exists \eta \sqsubseteq \rho\, (\eta \in B_1)\}$. Then B is upwards closed for $\sqsubseteq$, i.e. closed by superlists; trivially, X is closed by sublists; and B bars $X_0 = X \cap \mathtt{Incr}(\varphi, I)$. The latter holds by the monotonicity of bars (Lemma 10.1); in fact B_1 bars X_0 by the Claim, and $B_1 \subseteq B$. In all, Higman's lemma for bars (Theorem 13) applies, and yields that B bars X.

Now let $\sigma \in \mathtt{Inf}(I)$. Since B bars X, the infinite list σ has a finite prefix $\sigma_0 \in B$. By definition of B, there is $\tau_0 \sqsubseteq \sigma_0$ such that $\tau_0 \in B_1$, which is to say that $\tau_0 \in X_0 = \mathtt{Incr}(\varphi, I)$, $F(\tau_0) \in \mathbb{N}$ and $\mathtt{len}(\tau_0) \geqslant F(\tau_0)$. We extend τ_0 to an infinite sublist τ of σ. From $F(\tau_0) \in \mathbb{N}$ we get $F(\tau) = F(\tau_0) \leqslant \mathtt{len}(\tau_0)$. Hence the first $F(\tau)$ entries of τ form a prefix of τ_0 and thus are in w.i. order. ◀

▶ **Example 17.** Let $\sigma \in \mathtt{Inf}(\Sigma^*)$ be an infinite sequence of words over a finite alphabet Σ.

1. For all $k \in \mathbb{N}$ there is some w.i. length k subsequence of σ.
2. There are w.i. subsequences τ_1, τ_2, τ_3 of σ which have length $\mathtt{len}(\tau_1(0))+1$, $\mathtt{len}(\tau_2(0))^2+1$ and $2^{\mathtt{len}(\tau_3(0))}$.

Proof. Apply Corollary 16 to the functionals defined by $F_0(\rho) = k$, $F_1(\rho) = \mathtt{len}(\rho(0)) + 1$, $F_2(\rho) = \mathtt{len}(\rho(0))^2 + 1$ and $F_3(\rho) = 2^{\mathtt{len}(\rho(0))}$ where $\rho \in \mathtt{Fin}(\Sigma^*)$, which are bar-total continuous. In fact, $F_0^{-1}(\mathbb{N}) = \mathtt{Fin}(\Sigma^*)$ and $F_\nu^{-1}(\mathbb{N}) = \mathtt{Fin}(\Sigma^*) \setminus \{\mathtt{nil}\}$ for $\nu \in \{1, 2, 3\}$; whence $F_\nu^{-1}(\mathbb{N})$ bars $\mathtt{Fin}(\Sigma^*)$ in all cases. ◀

The particular case $k = 2$ of Example 17 means that there are $i < j$ for which $\sigma(i) \sqsubseteq \sigma(j)$. This is Higman's lemma in its usual form.

5 A Constructive Proof of Higman's Lemma for Bars

In this section we first prove some basic properties of wqo(bar): closure under finite product, finite union and right-invertible morphism. All these properties are classically true for the classically equivalent notion of wqo, see for example the original article by Higman [11]. Subsequently, we prove Higman's lemma for bars by induction on the finite alphabet Σ. We assume that all Δ^* are wqo(bar), for all Δ smaller than Σ, in order to prove that Σ^* is a wqo(bar). The crucial step will be proving that the anticone of every $v \in \Sigma^*$ is a wqo(bar).

5.1 Essential Properties of Wqo (bar)

We start by giving two immediate examples, of a quasi-order which is wqo(bar) and a quasi-order which is no wqo(bar). For every set I, $(I, =)$ is a quasi-order. Assume that Σ is a finite set, we can prove with intuitionistic logic that $(\Sigma, =)$ is a wqo(bar); whereas $(\mathbb{N}, =)$ is not.

▶ **Proposition 18** (wqo(bar)). *Assume that Σ is a finite set. Then*
1. $(\Sigma, =)$ *is wqo(bar).*
2. $(\mathbb{N}, =)$ *is not wqo(bar).*

Proof.
1. We assume that $X \subseteq \mathtt{Fin}(I)$ is closed by I-sublist, that B is closed by I-superlists, and that B bars $X \cap \mathtt{Incr}(=, \varphi, I)$, in order to prove that B bars X. We argue by induction on Σ. Assume that $\Sigma = \emptyset, \{x\}$. Then all labelling (if any) are constantly equal to x, therefore are weakly increasing. We deduce that $X \cap \mathtt{Incr}(=, \varphi, I) = X$ and we conclude that B bars X. Assume that Σ has two or more elements. Then $\Sigma = \Sigma_1 \cup \Sigma_2$ for some $\Sigma_1, \Sigma_2 \subset \Sigma$. Let $I_1 = \varphi^{-1}(\Sigma_1)$ and $I_2 = \varphi^{-1}(\Sigma_2)$. Then $I = I_1 \cup I_2$, and by antimonotonicity B bars $X \cap \mathtt{Incr}(=, \varphi, I_1)$ and B bars $X \cap \mathtt{Incr}(=, \varphi, I_2)$. By $X \subseteq \mathtt{Fin}(I)$ closed by I-sublist, B is closed by I-superlists and Lemma 19 we conclude that B bars X.
2. $\mathbb{N}, =$ is a partial order. In order to prove that it is not a wqo(bar), we will provide some $X \subseteq \mathtt{Fin}(I)$ closed by I-sublist, some B is closed by I-superlists, such that B bars $X \cap \mathtt{Incr}(=, \varphi, I)$ and B does not bars X. We choose $X =$ the set of non-repeating lists of length 1 words. X is closed by I-sublists and all its length $\geqslant 2$ sublists are not increasing, because if $i \neq j$, then $\langle i \rangle \not\sqsubseteq \langle j \rangle$. Then $X \cap \mathtt{Incr}(=, \varphi, I)$ consists of all lists with 1 word of length 1. These lists are not comparable by $>_1$, therefore this set is trivially well-founded by $>_1$, and it is barred by $B = \emptyset$. However, B does not bar X, because the infinite list $\sigma = \langle 0 \rangle, \langle 1 \rangle, \langle 2 \rangle, \ldots$ in $\mathbb{N}$ does not intersect $\emptyset$. ◀

For comparison, if we use the notion of wqo(set), then point 1 above say that all infinite lists on a finite set Σ have an infinite constant sublist, while point 2 says there is an infinite list on $\mathbb{N}$ with no infinite constant sublist. Point 1 requires classical logic (this is why we avoid using the notion of wqo(set)). Point 2 follows by taking the infinite list $0, 1, 2, 3, \ldots$.

In order to derive more basic properties of wqo(bar), we have first to find a constructive counterpart of the following classical property. In classical logic, given an infinite list σ in $\mathtt{List}(I_1 \cup I_2)$, if σ_1 is the sublist obtained by restricting σ to the elements in I_1, and σ_2 is the sublist obtained by restricting σ to the elements in I_2, then either σ_1 is infinite or σ_2 is infinite. We propose to call this property the *Riffling Property for infinite lists*, because if I_1, I_2 are disjoint, then σ can obtained from σ_1, σ_2 as when we riffle two decks of card in order to obtain a single deck of cards, while preserving the order we have in each deck. Riffling is not provable with intuitionistic logic, because we cannot decide whether we have an infinite sublist in $\texttt{Fin}(I_1)$ or in $\texttt{Fin}(I_2)$. In order to constructivise riffling, we prove a kind of contrapositive: if X is a set of lists and we bar with B the infinite I_1-lists in X and the infinite I_2-lists in X, then we bar with B the infinite $I_1 \cup I_2$-lists in X. When we state Riffling, we move from lists in X to sublists in X, and from sublists in the bar B to lists in the same B. Therefore Riffling requires two new assumptions, that B is closed by I-superlists and that X is closed by I-sublist. These are the same assumptions we have in the definition of wqo(bar).

▶ **Lemma 19** (Riffling for Bars). *Assume that the set X is closed by $I_1 \cup I_2$-sublists and the set B is closed by $I_1 \cup I_2$-superlists, then:*

$$B \textit{ bars } X \cap \texttt{Fin}(I_1) \ \wedge \ B \textit{ bars } X \cap \texttt{Fin}(I_2) \implies B \textit{ bars } X \cap \texttt{Fin}(I_1 \cup I_2)$$

From the Riffling Property for Bars we deduce with intuitionistic logic that wqo(bar) are closed under binary compatible union.

▶ **Lemma 20** (Compatible union of wqo's). *If $(P, \leqslant_P)$ and $(Q, \leqslant_Q)$ are wqo(bar), $(P \cup Q, \leqslant)$ is a quasi-order and $\leqslant_P$, $\leqslant_Q \subseteq \leqslant$, then $(P \cup Q, \leqslant)$ is a wqo(bar).*

The next step is to prove with intuitionistic logic that wqo(bar) are closed by componentwise product.

▶ **Lemma 21** (Componentwise product of wqo(bar)). *Assume that $(P, \leqslant_P)$ and $(Q, \leqslant_Q)$ are wqo(bar). Then $(P \times Q, \leqslant_P \times \leqslant_Q)$ with the componentwise order is a wqo(bar).*

The last preliminary step is to prove with intuitionistic logic that wqo(bar)'s are closed by right-invertible morphisms. Again, this property is easily proved for the classical definition of wqo. Assume that we have a morphism $f\colon P \to Q$ with right inverse $g\colon Q \to P$ (i.e., $fg = \texttt{id}_Q$) and $(P, \leqslant_P)$ is a wqo, then every infinite list $\sigma\colon \mathbb{N} \to Q$ is mapped by g into an infinite list $g\sigma\colon \mathbb{N} \to P$, which has an infinite w.i. sublist $\tau\colon \mathbb{N} \to P$, which is mapped by f into an infinite w.i. list $f\tau\colon \mathbb{N} \to Q$. From τ sublist of $g\sigma$ we deduce that $f\tau$ is a sublist of $fg\sigma$. From $fg\sigma = \sigma$ we conclude that $f\tau$ is an infinite w.i. sublist of σ. If we use the notion of wqo (bar), we can provide a proof with intuitionistic logic for the same result.

▶ **Lemma 22** (right-invertible morphism on a wqo(bar)). *Assume that $(P, \leqslant_P)$ is a wqo(bar), $(Q, \leqslant_Q)$ is a quasi-order and $f\colon P \to Q$ is a morphism with right inverse g.*[6] *Then $(Q, \leqslant_Q)$ is a wqo(bar).*

5.2 The Anticone of a Word is a Wqo (bar)

In this subsection, we fix a labelling $\varphi : I \to \Sigma^*$, and we assume that Δ^* is a wqo(bar) for all $\Delta \subset \Sigma$; then we prove that the anticone of every $v \in \Sigma^*$ is a wqo(bar). This is a crucial step in the proof of Higman's lemma for bars.

▶ **Lemma 23** (Slices and Anticones of a Word). *Assume that Σ is a finite alphabet and for all $\Delta \subset \Sigma$ the partial order Δ^* a wqo(bar). Let $v = c_1 \dots c_k \in \Sigma^*$, then:*

1. $\texttt{Slice}_\Sigma(v)$*, the slice of v, is a wqo(bar).*
2. $\not\sqsupseteq_\Sigma(v)$*, the anticone of v, is a wqo(bar).*

[6] Observe that g does not need to be a morphism.

5.3 A Decomposition of Finite Lists of Words over a Finite Language

In this subsection, where $\varphi : I \to P = \Sigma^*$ labels an arbitrary set I with words over a finite alphabet Σ, we introduce the last ingredient needed in the proof of the Higman lemma for bars. We extract from each finite list l with labels $\langle w_0, \ldots, w_{p-1}\rangle$ two disjoint sublists:

1. some φ-w.i.sub-list $\mathtt{Lex}(l, \varphi)$ of l, with labels $w_{i_0} \sqsubseteq \ldots \sqsubseteq w_{i_{n-1}}$. $\mathtt{Lex}(l, \varphi)$ is the sub-list obtained by selecting each time as next element *the first element making the sub-list φ-w.i.*;
2. the suffix $\mathtt{Suff}(l, \varphi)$ of l, with labels $\langle w_m, \ldots, w_{p-1}\rangle$ of l, such that $m = i_{n-1} + 1$, and that $w_{i_{n-1}} \not\sqsubseteq w_m, \ldots, w_{p-1}$. If this is not possible, then $\mathtt{Suff}(l, \varphi)$ is the empty list.

In our terminology, the elements of $\mathtt{Suff}(l, \varphi)$ are in the anticone of $w_{i_{n-1}}$, where $w_{i_{n-1}}$ is the last element of $\mathtt{Lex}(l, \varphi)$. We will prove the Higman lemma for bars by bar induction on such pair of lists. The formal definition of the two sub-lists $\mathtt{Lex}, \mathtt{Suff}$ runs as follows. We have to define first two integer lists $\mathtt{lex}, \mathtt{suff}$, with low case $\mathtt{l}, \mathtt{s}$, consisting of the list of indexes of $\mathtt{Lex}, \mathtt{Suff}$ in φl.

▶ **Definition 24** (Decomposition of a list). *Assume l is any finite list on I, labeled by a map $\varphi : I \to P = \Sigma^*$. Suppose $\varphi l = \langle w_0, \ldots, w_{p-1}\rangle$ is the list of labels of l. By induction on l, we define $\mathtt{lex}(l, \varphi)$, $\mathtt{suff}(l, \varphi)$.*

1. *We define $\mathtt{lex}(\mathtt{Nil}, \varphi) = \mathtt{suff}(\mathtt{Nil}, \varphi) = \mathtt{Nil}$ and $\mathtt{lex}(\langle i\rangle, \varphi) = \langle 0\rangle$, $\mathtt{suff}(\langle i\rangle, \varphi) = \mathtt{Nil}$.*
2. *Suppose $\mathtt{len}(l) = p \geqslant 1$, $\mathtt{lex}(l, \varphi) = \langle i_0 \ldots, i_{n-1}\rangle$ (an integer list) and $x \in I$. We define the clause for $l\star\langle x\rangle$ by cases on the condition: "$w_{i_{n-1}} \sqsubseteq \varphi(x)$".*
   a. Assume $w_{i_{n-1}} \sqsubseteq \varphi(x)$. *Then we set $\mathtt{lex}(l\star\langle x\rangle, \varphi) = \mathtt{lex}(l, \varphi)\star\langle p\rangle$ (we add the index p of x to $\mathtt{lex}$) and $\mathtt{suff}(l\star\langle x\rangle, \varphi) = \mathtt{Nil}$ (we reset $\mathtt{suff}$ to $\mathtt{nil}$).*
   b. Assume $w_{i_{n-1}} \not\sqsubseteq \varphi(x)$. *Then we set $\mathtt{lex}(l\star\langle x\rangle, \varphi) = \mathtt{lex}(l, \varphi)$ ($\mathtt{lex}$ stays the same) and $\mathtt{suff}(l\star\langle x\rangle, \varphi) = \mathtt{suff}(l, \varphi)\star\langle p\rangle$ (we add the index p of x to $\mathtt{suff}$).*

Finally, we define $\mathtt{Lex}, \mathtt{Suff}$, the maps with capital $\mathtt{L}, \mathtt{S}$, by: $\mathtt{Lex}(l, \varphi) = l\mathtt{lex}(l, \varphi)$ and $\mathtt{Suff}(l, \varphi) = l\mathtt{suff}(l, \varphi)$.

A (crucial) example: let $I = \Sigma^*$, $\varphi = \mathtt{id}$ (labelling φl and list l coincide), and $l = \langle w_0, w_1, w_2, w_3, w_4\rangle$, with

$$w_0 = a, \quad w_1 = ab, \quad w_2 = abb, \quad \text{and } w_3 = bb, \; w_4 = bbb$$

According to Def.24 we obtain:

1. for $m = \mathtt{nil}$: $\mathtt{lex}(m, \varphi) = \mathtt{nil}$.
2. for $m = \langle w_0\rangle$: $\mathtt{lex}(m, \varphi) =$ the index 0 of w_0
3. for $m = \langle w_0, w_1\rangle$: $\mathtt{lex}(m, \varphi) =$ the indexes $0, 1$ of w_0, w_1
4. for $m = \langle w_0, w_1, w_2\rangle$: $\mathtt{lex}(m, \varphi) =$ the indexes $0, 1, 2$ of w_0, w_1, w_2

When m increases to $m = \langle w_0, w_1, w_2, w_3\rangle$, the new word w_3 added to m is discarded in $\mathtt{lex}(m, \varphi)$. Indeed, we have $w_2 \not\sqsubseteq w_3, w_4$, therefore if $m = \langle w_0, w_1, w_2, w_3\rangle$ then $\mathtt{lex}(m, \varphi)$ is again equal to the indexes $0, 1, 2$ of w_0, w_1, w_2. The same when $m = \langle w_0, w_1, w_2, w_3, w_4\rangle$: the new word w_4 added to m is again discarded, and we still have $\mathtt{lex}(m, \varphi) =$ the integer list $0, 1, 2$.

The indexes of the discarded words are piled up in $\mathtt{suff}$. The first three values of $\mathtt{suff}(m, \varphi)$ are: $\mathtt{nil}$, $\mathtt{nil}$, $\mathtt{nil}$. From $w_2 \not\sqsubseteq w_3, w_4$, we deduce the following values for $\mathtt{suff}$: $\mathtt{suff}(m, \varphi) =$ the integer list whose only element is 3, and $\mathtt{suff}(m, \varphi) =$ the integer list $3, 4$.

The outputs of $\texttt{Lex}(m,\varphi)$ and $\texttt{Suff}(m,\varphi)$ (with capital $\texttt{L},\texttt{S}$) are the same, except that we take words instead of indexes of words. For the same values of m we obtain for $\texttt{Lex}(m,\varphi)$: $\texttt{nil}$, $\langle w_0\rangle$, $\langle w_0,w_1\rangle$, $\langle w_0,w_1,w_2\rangle$, then again $\langle w_0,w_1,w_2\rangle$ and again $\langle w_0,w_1,w_2\rangle$.

The words w_3,w_4 discarded from $\texttt{Lex}(m,\varphi)$ are piled up in $\texttt{Suff}(m,\varphi)$. Indeed, according to Def.24 we obtain for $\texttt{Suff}(m,\varphi)$: $\texttt{nil}$, $\texttt{nil}$, $\texttt{nil}$, then $\langle w_3\rangle$ and $\langle w_3,w_4\rangle$.

A last example. Suppose we add $w_5 = abb$ to m. In this case $w_2 \sqsubseteq w_5$, then w_5 is added to $\texttt{Lex}(l,\varphi)$ and we obtain $\texttt{Lex}(l\star\langle w_5\rangle,\varphi) = w_0w_1w_2w_5$. Instead, $\texttt{Suff}$ is reset to $\texttt{nil}$: according to Def. 24, we obtain $\texttt{Suff}(l\star\langle w_5\rangle,\varphi) = \texttt{nil}$.

The name we choose for the map $\texttt{lex}$ comes from the fact that $f = \texttt{lex}(l,\varphi)$ is the minimum in the lexicographic ordering of all integer lists such that lf is a φ-w.i. sublist of l. In this paper, however, we do not need a proof of this feature of f and we do not include further details.

The following properties of $f = \texttt{lex}(l,\varphi)$ and $g = \texttt{suff}(l,\varphi)$ are immediate from the definition. First, that $lf \in \texttt{Incr}(\varphi,I)$ for all $l \in \texttt{Fin}(I)$. Second, if $l > \texttt{Nil}$, $g = \texttt{suff}(l,\varphi)$, then lg is equal to the suffix of l after the last element of lf, and that lg is in the anticone of the last element of lf. Both properties can be proved by induction on l.

5.4 Proof of the Main Theorem

▶ **Theorem 25** (Higman's lemma for bars). *If Σ is a finite alphabet, then Σ^* is a wqo(bar).*

Proof. We argue by principal induction on Σ. If $\Sigma = \emptyset$ then $\Sigma^* = \texttt{Nil}$ and Σ^* is a wqo by Lemma 18. Assume Σ has some element. We assume that for all $\Delta \subset \Sigma$ the partial order Δ^* is a wqo(bar), in order to prove that Σ^* is a wqo(bar). We assume that I is a set, $\varphi : I \to \Sigma^*$ any labelling of elements of I by words, $X \subseteq \texttt{Fin}(I)$ is a set of finite I-lists closed by I-sublists, B is a set of finite I-lists closed by I-superlists, and B bars $X \cap \texttt{Incr}(\varphi,I)$; our goal is to prove that B bars X.

Let $\texttt{Lex},\texttt{Suff}$ as in Def. 24, and $\sigma,l \in X$. We define a map $f(\sigma) = \texttt{Lex}(\sigma) \times \texttt{Suff}(\sigma)$ proving that $f\colon X \to Y$ is a morphism, where $Y := \Sigma_{l\in X\cap\texttt{Incr}(\varphi,I)}Y_l$, for a family of sets $\{Y_l \mid l \in X \cap \texttt{Incr}(\varphi,I)\}$ we are going to define. We will prove that Y is barred by some D such that $f^{-1}(D) \subseteq B$; "B bars X" follows from the Simulation Lemma (9) and monotonicity.

If $l = \texttt{Nil}$, we set $Y_{\texttt{Nil}} = \{\texttt{Nil}\}$. If $l \neq \texttt{Nil}$, we define each set Y_l as the set of all I-lists in X φ-labeled by words which are *not* super-words of (which are in the anticone of) the last word of φl. We formally define Y_l as follows. Let v be *the last element of φl*: then we set $Y_l := \texttt{Fin}(\varphi^{-1}(\not\sqsupseteq_\Sigma(v))) \cap X$.

By definition of $\texttt{Lex}$, $\texttt{Suff}$ and the closure of X by I-sublists, we have $\texttt{Lex}(\sigma) \in X \cap \texttt{Incr}(\varphi,I)$ and $\texttt{Suff}(\sigma) \in Y_l$. Moreover, by definition of $\texttt{Lex}$ and $\texttt{Suff}$, whenever we add one element i to l, either we add the same i to $\texttt{Lex}(\sigma,\varphi)$, or $\texttt{Lex}(\sigma,\varphi)$ stays the same and we add i to $\texttt{Suff}(\sigma,\varphi)$. Thus, f is a morphism from $(X,>_1)$ to $\Sigma_{l\in X\cap\texttt{Incr}(\varphi,I)}Y_l$ with relation the lexicographic product $>_1 \times >_1$. By Lemma 23.2 (Slices and Anticones), $\not\sqsupseteq_\Sigma(v)$ is a wqo(bar). B bars $X \cap \texttt{Incr}(\varphi,I)$ by assumption. Then B bars the subset $\texttt{Fin}(\varphi^{-1}(\not\sqsupseteq_\Sigma(v))) \cap X \cap \texttt{Incr}(\varphi,I)$ by antimonotonicity. $\not\sqsupseteq_\Sigma(v)$ is a wqo(bar), therefore B bars $\texttt{Fin}(\varphi^{-1}(\not\sqsupseteq_\Sigma(v))) \cap X$, which is Y_l. Let D be the set of pairs (l,m) such that $l \in B$ or $m \in B$. By Lemma 11 (Lexicographic Product), D bars $Y = \Sigma_{l\in X\cap\texttt{Incr}(\varphi,I)}Y_l, >_1 \times >_1$. By Simulation Lemma (9) we deduce that $f^{-1}(D)$ bars X. In order to prove that B bars X, by monotonicity it is enough to prove that $f^{-1}(D) \subseteq B$.

Assume that $\sigma \in f^{-1}(D)$, then $f(\sigma) = \texttt{Lex}(\sigma) \times \texttt{Suff}(\sigma) \in D$, and by definition of D, we deduce that $\texttt{Lex}(\sigma) \in B$ or $\texttt{Suff}(\sigma) \in B$. From $\texttt{Lex}(\sigma),\texttt{Suff}(\sigma) \sqsubseteq \sigma$ and closure of B by I-superlists, we conclude that $\sigma \in B$, as wished. ◀

Conclusion

Higman's lemma for sequences says that over a finite alphabet every infinite sequence of words has an infinite weakly increasing subsequence, and is inherently nonconstructive. As a constructive alternative we now have put forward what we call Higman's lemma for bars: over a finite alphabet, every bar for the weakly increasing finite lists of words which is closed by super-lists is already a bar for *all* finite lists. In particular, for every total continuous functional, every infinite sequence of such words has a weakly increasing finite sublist of length bounded below by the functional. We also proved the common form of Higman's lemma: the words over a finite alphabet form a well quasi-order, for our notion of well quasi-order. As we work as much as possible in settings more abstract than the one of words over a finite alphabet, we prepare for a constructive theory of well (quasi-)order, and, more in general, for a constructive version of classical theories dealing with Π^1_2-statements.

References

1 P. Aczel. *An introduction to inductive definitions*, volume 90 of *Stud. Logic Found. Math.*, pages 739–782. North-Holland, 1977.

2 S. Berardi and S. Steila. Ramsey's Theorem for Pairs and k Colors as a sub-Classical Principle of Arithmetic. *J. Symbolic Logic*, 82(2):737–753, 2017.

3 J. Berger. Dickson's Lemma and Higman's Lemma are Equivalent. *South American Journal of Logic*, 2(1):35–39, 2016.

4 S. Berghofer. A Constructive Proof of Higman's Lemma in Isabelle. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 3085 of *Lecture Notes in Computer Science*, pages 66–82, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

5 Yves Bertot and Ekaterina Komendantskaya. Inductive and coinductive components of corecursive functions in coq. *Electronic Notes in Theoretical Computer Science*, 203(5):25–47, 2008. Proceedings of the Ninth Workshop on Coalgebraic Methods in Computer Science (CMCS 2008). `doi:10.1016/j.entcs.2008.05.018`.

6 G. Buriola, P. Schuster, and I. Blechschmidt. A Constructive Picture of Noetherian Conditions and Well Quasi-orders. In Gianluca Della Vedova, Besik Dundua, Steffen Lempp, and Florin Manea, editors, *Unity of Logic and Computation*, pages 50–62, Cham, 2023. Springer Nature Switzerland.

7 T. Coquand and D. Fridlender. A proof of Higman's lemma by structural induction. Unpublished Manuscript. `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.47.486`, November 1993.

8 Thierry Coquand and Henrik Persson. Gröbner bases in type theory. In T. Altenkirch, W. Naraschewski, and B. Reus, editors, *Types for Proofs and Programs (Irsee, 1998)*, volume 1657 of *Lecture Notes in Comput. Sci.*, pages 33–46. Springer, Berlin, 1999.

9 D. Fridlender. *Higman's Lemma in Type Theory.* PhD thesis, Chalmers University of Technology, Göteborg, 1997.

10 J. Goubault-Larrecq. A Constructive Proof of the Topological Kruskal Theorem. In Krishnendu Chatterjee and Jirí Sgall, editors, *Mathematical Foundations of Computer Science 2013*, pages 22–41, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

11 G. Higman. Ordering by Divisibility in Abstract Algebras. *Proceedings of the London Mathematical Society*, s3-2(1):326–336, 1952.

12 J.B. Kruskal. Well-quasi-ordering, the tree theorem, and Vazsonyi's conjecture. *Trans. Amer. Math. Soc.*, 95:210–225, 1960.

13 J.B. Kruskal. The theory of well-quasi-ordering: A frequently discovered concept. *J. Comb. Theory A*, 13:297–305, 1972.

14 Ray Mines, Fred Richman, and Wim Ruitenburg. *A Course in Constructive Algebra.* Springer, New York, 1988. Universitext.

15 C.R. Murthy and J.R. Russell. A constructive proof of Higman's lemma. *5th Annual Symposium on Logic in Computer Science, Philadelphia PA*, pages 257–267, 1992.

16 C.St.J.A. Nash-William. On well-quasi-ordering finite trees. *Proc. Cambridge Phil. Soc.*, 59:833–835, 1963.

17 Hervé Perdry. Strongly Noetherian rings and constructive ideal theory. *J. Symb. Comput.*, 37(4):511–535, 2004.

18 Hervé Perdry and Peter Schuster. Noetherian orders. *Math. Structures Comput. Sci.*, 21:111–124, 2011.

19 T. Powell. Applying Gödel's Dialectica interpretation to obtain a constructive proof of Higman's lemma. *Proceedings of Classical Logic and Computation (CLAC'12)*, volume 97 of *EPTCS*:49–62, 2012.

20 T. Powell. Well Quasi-orders and the Functional Interpretation. In Peter M. Schuster, Monika Seisenberger, and Andreas Weiermann, editors, *Well-Quasi Orders in Computation, Logic, Language and Reasoning: A Unifying Concept of Proof Theory, Automata Theory, Formal Languages and Descriptive Set Theory*, pages 221–269, Cham, 2020. Springer International Publishing. doi:10.1007/978-3-030-30229-0_9.

21 Michael Rathjen. Generalized inductive definitions in constructive set theory. In Laura Crosilla and Peter Schuster, editors, *From Sets and Types to Topology and Analysis: Towards Practicable Foundations for Constructive Mathematics*, volume 48 of *Oxford Logic Guides*, chapter 16. Clarendon Press, Oxford, 2005.

22 F. Richman and G. Stolzenberg. Well quasi-ordered sets. *Advances in Mathematics*, 97:145–153, 1993.

23 H. Schwichtenberg, M. Seisenberger, and F. Wiesnet. Higman's Lemma and Its Computational Content. In R. Kahle, T. Strahm, and T. Studer, editors, *Advances in Proof Theory*, pages 353–375, Cham, 2016. Springer International Publishing. doi:10.1007/978-3-319-29198-7_11.

24 M. Seisenberger. An Inductive Version of Nash-Williams' Minimal-Bad-Sequence Argument for Higman's Lemma. *Types for Proofs and Programs*, *LNCS* Vol. 2277, 2001.

25 Monika Seisenberger. Kruskal's tree theorem in a constructive theory of inductive definitions. In P. Schuster, U. Berger, and H. Osswald, editors, *Reuniting the antipodes—constructive and nonstandard views of the continuum (Venice, 1999)*, volume 306 of *Synthese Library*, pages 241–255. Kluwer, Dordrecht, 2001.

26 C. Sternagel. A Mechanized Proof of Higman's Lemma by Open Induction. In Peter M. Schuster, Monika Seisenberger, and Andreas Weiermann, editors, *Well-Quasi Orders in Computation, Logic, Language and Reasoning: A Unifying Concept of Proof Theory, Automata Theory, Formal Languages and Descriptive Set Theory*, pages 339–350, Cham, 2020. Springer International Publishing. doi:10.1007/978-3-030-30229-0_12.

27 A.S. Troelstra. Metamathematica Investigation of Intuitionistic Arithmetic and Analysis. *ILLC pre-publication series*, pages X–93–05, 1993.

28 W. Veldman. An Intuitionistic Proof of Kruskal's Theorem. *Archive for Mathematical Logic*, 43(2):215–264, 2004.