# A Locale for Minimal Bad Sequences

Christian Sternagel*

JAIST, Japan
c-sterna@jaist.ac.jp

**Abstract**

We present a locale that abstracts over the necessary ingredients for constructing a minimal bad sequence, as required in classical proofs of Higman's lemma and Kruskal's tree theorem.

## 1 Introduction

The so called *minimal bad sequence argument* was first used by Nash-Williams in [8]; where he first proves a variant of *Higman's lemma* [4] for finite sets, and then – again using a minimal bad sequence argument – *Kruskal's tree theorem*. This proof is usually considered to be simple and elegant. To a certain extend we agree, but then again, formalizing a proof (using a proof assistant) typically requires us to be more rigorous than on paper. During our Isabelle/HOL [9] formalization of Higman's lemma and Kruskal's tree theorem [11] we found that Nash-Williams' reasoning for constructing a minimal bad sequence is far from comprehensive. That is, assuming that there exists a minimal bad sequence, both proofs can be formalized almost exactly as presented in [8].[1] But to prove the existence of such a minimal bad sequence turns out to be rather involved. (A step that is omitted in any classical paper-proof using the minimal bad sequence argument we could catch sight of.)

To this end, we formalized a locale *mbs* that encapsulates the required ingredients for constructing a minimal bad sequence starting from an arbitrary bad sequence. Interpreting this locale for lists (with list-embedding) and finite trees (with homeomorphic embedding on finite trees) is easy and takes care of the biggest "gaps" in the paper-proofs of Nash-Williams.

The remainder is structured as follows: In Section 2 we give some preliminaries on well-quasi-order (wqo) theory and put our minimal bad sequence construction into context. Afterwards, in Section 3, we present our locale and sketch our construction of a minimal bad sequence. Then, in Section 4, we give two applications. Finally, we conclude in Section 5.

## 2 Preliminaries

In this section we recall well-quasi-orders as well as Higman's lemma and Kruskal's tree theorem. This serves to give a context for our minimal bad sequence construction. Moreover, we give basic definitions that are used throughout our formalization.

In the following we use $\preceq$ for an arbitrary (not necessarily reflexive) binary relation on the elements of an arbitrary set $A$ (when not stated otherwise). In the literature, well-quasi-orders are typically defined as follows:

**Definition 1.** A set $A$ is *well-quasi-ordered* (wqo) by $\preceq$ iff $\preceq$ is reflexive, transitive, and satisfies: for every infinite sequence $f$ over elements of $A$ there are indices $i < j$, s.t., $f\ i \preceq f\ j$.

*Supported by the Austrian Science Fund (FWF): J3202.

[1] Apart from two claims of the form "the absence of a bad sequence of a certain shape implies the absence of any bad sequence," whose proofs are omitted.

Using the terminology of [8] an infinite sequence $f$ for which we have indices $i < j$, s.t., $f\ i \preceq f\ j$ is called *good*. A sequence that is not good is called *bad*. Moreover, a relation $\preceq$ satisfying the last condition of Definition 1 (i.e., all infinite sequences over elements of $A$ are good) is called *almost full*.[2]

Now consider the above concepts as defined in our Isabelle/HOL formalization, where we encode binary relations by functions of type $\alpha \Rightarrow \alpha \Rightarrow bool$ (i.e., binary predicates) and infinite sequences by functions of type $nat \Rightarrow \alpha$.

An infinite sequence $f$ is *good* (w.r.t. a binary predicate $\preceq$), written $good_{\preceq}(f)$, iff $\exists\, i\ j.\ i < j \wedge f\ i \preceq f\ j$. It is *bad* (w.r.t. $\preceq$), written $bad_{\preceq}(f)$, iff it is not good. A binary relation $\preceq$ is *reflexive on* a set $A$, written $refl_A(\preceq)$, iff $\forall\, a{\in}A.\ a \preceq a$. It is *transitive on* $A$, written $trans_A(\preceq)$, iff $\forall\, a{\in}A.\ \forall\, b{\in}A.\ \forall\, c{\in}A.\ a \preceq b \wedge b \preceq c \longrightarrow a \preceq c$. It is *almost full on* $A$, written $af_A(\preceq)$, iff $\forall f.\ (\forall\, i.\ f\ i \in A) \longrightarrow good_{\preceq}(f)$. The relation $\preceq$ is a *wqo on* $A$, written $wqo_A(\preceq)$, iff $trans_A(\preceq) \wedge af_A(\preceq)$. It *is well-founded on* $A$, written $wf_A(\preceq)$, iff $\neg\ (\exists f.\ \forall\, i.\ f\ i \in A \wedge f\ (i+1) \preceq f\ i)$ (i.e., there are no infinite descending sequences over elements of $A$). It is easy to see that every almost full relation is also reflexive.

**Lemma 1.** $af_A(\preceq) \Longrightarrow refl_A(\preceq)$

*Proof.* Take an arbitrary but fixed $x \in A$. We have to show $x \preceq x$. To this end, consider the infinite sequence $f\ i = x$, which is an infinite sequence over elements of $A$ and thus, since $\preceq$ is almost full, we obtain indices $i < j$ with $f\ i \preceq f\ j$. Since $f$ equals $x$ at every position, we obtain the required $x \preceq x$. □

Thus, any almost full relation that is transitive is a wqo and the other way round. Since, in our formalization, we strive for minimality, and furthermore transitivity is not required in the proofs of Higman's lemma and Kruskal's tree theorem (and typically easily added afterwards), we concentrate on almost full relations.

Before we continue, note that wqos are interesting (at least) due to the following fact: The strict part $\prec$ of a wqo $\preceq$ is well-founded on $A$. Where $x \prec y = (x \preceq y \wedge y \npreceq x)$.

**Lemma 2.** $wqo_A(\preceq) \Longrightarrow wf_A(\prec)$

*Proof.* Assume to the contrary that $\prec$ is not well-founded on $A$. Then there is an infinite descending sequence $f$ over elements of $A$. Hence, for all $i < j$ we have $f\ j \prec f\ i$, since $\prec$ is transitive. Moreover, since $\prec$ is irreflexive, $f\ i \npreceq f\ j$ for all $i < j$. Thus $f$ is bad, contradicting the fact that $\preceq$ is almost full. □

Now, consider Higman's lemma and Kruskal's tree theorem as stated in our formalization.

**Lemma 3** (Higman's Lemma)**.** $wqo_A(\preceq) \Longrightarrow wqo_{A^*}(\preceq_{\mathsf{emb}})$

**Theorem 1** (Kruskal's Tree Theorem)**.** $wqo_A(\preceq) \Longrightarrow wqo_{\mathcal{T}(A)}(\preceq_{\mathsf{emb}})$

In the above two statements, $A^*$ denotes the set of finite lists built over elements of $A$ and $\mathcal{T}(A)$ denotes the set of finite trees built over elements of $A$. The binary relation $\preceq_{\mathsf{emb}}$, denotes homeomorphic embedding on finite lists and finite trees, respectively (see Section 4 for concrete definitions). (Note that the interesting parts of the above proofs correspond to $af_A(\preceq) \Longrightarrow af_{A^*}(\preceq_{\mathsf{emb}})$ and $af_A(\preceq) \Longrightarrow af_{\mathcal{T}(A)}(\preceq_{\mathsf{emb}})$, respectively.)

In both proofs (as presented by Nash-Williams) the existence of a minimal bad sequence is essential. However, the only thing Nash-Williams has to say about the construction of a

[2] The notion *almost full* was first introduced in [13] and very recently revived in [14].

minimal bad sequence is roughly (where we use $A^\circ$ to denote the set of "objects" built over elements of $A$; which might refer to the set of finite subsets, the set of finite lists, the set of finite trees, ... in a concrete case):

> *Select an $x_1 \in A^\circ$ such that $x_1$ is the first term of a bad sequence of members of $A^\circ$ and $|x_1|$ is as small as possible. Then select an $x_2$ such that $x_1$, $x_2$ (in that order) are the first two terms of a bad sequence of members of $A^\circ$ and $|x_2|$ is as small as possible [...]. Assuming the Axiom of Choice, this process yields a [minimal] bad sequence [...]*

Interestingly, most non-formalized proofs of Higman's lemma and Kruskal's tree theorem in the literature (that the author is aware of) are similarly vague about the actual construction of a minimal bad sequence. The point is that a crucial proof is missing in the above recipe, namely that it is actually possible to select elements as described.

In the next section we make the notion *minimal bad sequence* more concrete (i.e., answer the question: "Minimal in what sense?") but at the same time abstract over the basic ingredients ("What are the properties that have to be satisfied for a minimal bad sequence to exist?").

# 3 Constructing Minimal Bad Sequences

We encapsulate the construction of a minimal bad sequence over elements from a given set (which we call *objects*) inside a locale taking the following arguments:

- a function $A^\circ$ that returns the set of objects that are built over elements of $A$,
- a relation $\preceq_\circ$ that is used to check whether an infinite sequence of objects is good (where $\preceq$ is a relation on elements of $A$),
- and a relation $\lhd$ used for checking minimality (whose reflexive closure is denoted by $\trianglelefteq$).

The required properties are:

- *right-compatibility* of $\lhd$ with $\preceq_\circ$: $[\![x \preceq_\circ y;\ y \lhd z]\!] \Longrightarrow x \preceq_\circ z$
- *well-foundedness* of $\lhd$ on elements of $A^\circ$: $wf_{A^\circ}(\lhd)$
- *transitivity* of $\lhd$: $[\![x \lhd y;\ y \lhd z]\!] \Longrightarrow x \lhd z$
- $\lhd$ *reflects* the property of being in $A^\circ$: $[\![x \lhd y;\ y \in A^\circ]\!] \Longrightarrow x \in A^\circ$

In the following, we will need a way of piecing together infinite sequences. Given two infinite sequences $f$ and $g$, we can splice them at position $n$, s.t., in the resulting sequence all elements at positions smaller than $n$ are taken from $f$ and all others are taken from $g$. This operation is written $f\langle n\rangle g$ and defined by $f\langle n\rangle g \equiv \lambda j.$ *if* $n \leq j$ *then* $g\ j$ *else* $f\ j$.

Furthermore, we say that an infinite sequence $f$ is *minimal at* a position $n$, written $min_n(f)$, if all "subsequences" of $f$ that coincide on the first $n - 1$ elements and have a smaller (w.r.t. $\lhd$) $n$-th element are good (w.r.t. $\preceq_\circ$). The sense in which we use "subsequence" here, is made clear by the following definition:

$$min_n(f) \equiv \forall\, g.\ (\forall\, i{<}n.\ g\ i = f\ i) \wedge g\ n \lhd f\ n \wedge (\forall\, i{\geq}n.\ \exists\, j{\geq}n.\ g\ i \trianglelefteq f\ j) \longrightarrow good_{\preceq_\circ}(g)$$

which makes sure that objects in $g$ only contain elements that where already present in some object of $f$.

Now the key lemma in the construction of a minimal bad sequence is the following:

**Lemma 4.**

$$\begin{aligned}&[\![f\ (n+1) \in A^\circ;\ min_n(f);\ bad_{\preceq_\circ}(f)]\!]\\ &\Longrightarrow \exists\, g.\ (\forall\, i{\leq}n.\ g\ i = f\ i) \wedge\\ &\qquad g\ (n+1) \trianglelefteq f\ (n+1) \wedge\\ &\qquad (\forall\, i{\geq}n+1.\ \exists\, j{\geq}n+1.\ g\ i \trianglelefteq f\ j) \wedge\\ &\qquad bad_{\preceq_\circ}(f\langle n+1\rangle g) \wedge min_{n+1}(f\langle n+1\rangle g)\end{aligned}$$

*Proof.* Let $P(f)$ abbreviate the conclusion of the key lemma. We use the well-founded induction principle induced by the well-foundedness of $\lhd$ on the term $f\ (n+1)$. As a result, we have to show $P(g)$ for some arbitrary but fixed sequence $g$. We proceed by a case analysis on whether $g$ is already minimal at position $n+1$ or not. For details, check [11]. □

By Lemma 4 we obtain a bad sequence that is minimal at $n+1$ from a bad sequence that is minimal at $n$. This allows us to inductively define a (globally) minimal bad sequence. The only missing part is that there actually is a bad sequence that is minimal at $0$, which is shown by the following lemma:

**Lemma 5.** $[\![f\ 0 \in A^\circ;\ bad_{\preceq_\circ}(f)]\!] \Longrightarrow \exists\, g.\ (\forall\, i.\ \exists\, j.\ g\ i \trianglelefteq f\ j) \wedge min_0(g) \wedge bad_{\preceq_\circ}(g)$

*Proof.* We use the same techniques as in the proof of Lemma 4, but the second part of the case analysis is considerably simpler. □

Finally we are in a position to show the existence of a minimal bad sequence over objects constructed from elements of $A$.

**Theorem 2.**

$$[\![\forall\, i.\ f\ i \in A^\circ;\ bad_{\preceq_\circ}(f)]\!] \Longrightarrow \exists\, g.\ bad_{\preceq_\circ}(g) \wedge (\forall\, n.\ min_n(g)) \wedge (\forall\, i.\ g\ i \in A^\circ)$$

*Proof.* By Lemma 4 (which holds for every $f$ and $n$) and the Axiom of Choice, we obtain a choice function $\nu$, s.t., $\nu(f,n)$ yields the corresponding witness. Moreover, by Lemma 5 we obtain a bad sequence $g$ that is minimal at $0$. This allows us to define the auxiliary sequence of sequences $m'$ recursively by:

$$\begin{aligned}m'(0) &= \nu(g,0)\\ m'(n+1) &= m'(n)\langle n+1\rangle\nu(m'(n),n)\end{aligned}$$

The actual minimal bad sequence is then $m(i) = m'(i)(i)$. And the proof is (mainly) shown by induction over $i$ (after considerably strengthening the induction hypothesis). Again, we refer to [11] for the gory details. □

# 4 Applications

Our current applications for the *mbs* locale are Lemma 3 and Theorem 1 (where the former is used in the proof of the latter). Thus, we have to interpret the locale once in the context of lists and once in the context of finite trees. For the latter we use the datatype

**datatype** $\alpha$ *tree* = *Node* $\alpha$ ($\alpha$ *tree list*)

representing finite (non-empty) trees (isomorphic to ground terms of first-order term rewriting as used in [12], where we want to apply our formalization of wqo theory eventually).

The three parameters for the list case are *lists*::$\alpha$ *set* $\Rightarrow$ $\alpha$ *list set* (the set of lists over elements from a given set), $\preceq_{\mathsf{emb}}$::$\alpha$ *list* $\Rightarrow$ $\alpha$ *list* $\Rightarrow$ *bool* (homeomorphic embedding on lists) and $<$ (the suffix relation on lists), where we use the following definitions:

**Definition 2.** The *homeomorphic embedding* on lists, w.r.t. an arbitrary relation $\preceq$ on list elements, is defined inductively by

$[] \preceq_{\mathsf{emb}} ys$
$xs \preceq_{\mathsf{emb}} ys \Longrightarrow xs \preceq_{\mathsf{emb}} y \# ys$
$[\![x \preceq y;\ xs \preceq_{\mathsf{emb}} ys]\!] \Longrightarrow x \# xs \preceq_{\mathsf{emb}} y \# ys$

which essentially says that we are allowed to drop elements or replace elements by smaller ones (w.r.t., $\preceq$) when going from right to left. The *suffix relation* on lists is given by

$xs < ys \equiv \exists us.\ ys = us \mathbin{@} xs \wedge us \neq []$

In order to obtain a $<$-minimal $\preceq_{\mathsf{emb}}$-bad sequence we have to prove the properties:

$[\![xs \preceq_{\mathsf{emb}} ys;\ ys < zs]\!] \Longrightarrow xs \preceq_{\mathsf{emb}} zs$
$wf_{A^*}(<)$
$[\![xs < ys;\ ys < zs]\!] \Longrightarrow xs < zs$
$[\![xs < ys;\ ys \in A^*]\!] \Longrightarrow xs \in A^*$

all of which are easy.

For the tree case, the parameters are $\mathcal{T}(A)$::$\alpha$ *tree set* (the set of trees over elements from a given set), $\preceq_{\mathsf{emb}}$::$\alpha$ *tree* $\Rightarrow$ $\alpha$ *tree* $\Rightarrow$ *bool* (homeomorphic embedding on trees) and $\lhd$ (the subtree relation; similar to the subterm relation on terms), where we use the following definitions:

**Definition 3.** The *set of trees* over elements from $A$ is given by

$[\![x \in A;\ ts \in \mathcal{T}_{list}(A)]\!] \Longrightarrow Node\ x\ ts \in \mathcal{T}(A)$
$[] \in \mathcal{T}_{list}(A)$
$[\![t \in \mathcal{T}(A);\ ts \in \mathcal{T}_{list}(A)]\!] \Longrightarrow t \# ts \in \mathcal{T}_{list}(A)$

*Homomorphic embedding* on trees, w.r.t. an arbitrary relation $\preceq$ on tree elements, is defined inductively by

$t \in set\ ts \Longrightarrow t \preceq_{\mathsf{emb}} Node\ f\ ts$
$[\![f \preceq g;\ ss \preceq^{=}_{\mathsf{emb}} ts]\!] \Longrightarrow Node\ f\ ss \preceq_{\mathsf{emb}} Node\ g\ ts$
$[\![s \preceq_{\mathsf{emb}} t;\ t \preceq_{\mathsf{emb}} u]\!] \Longrightarrow s \preceq_{\mathsf{emb}} u$
$s \preceq_{\mathsf{emb}} t \Longrightarrow Node\ f\ (ss \mathbin{@} s \# ts) \preceq_{\mathsf{emb}} Node\ f\ (ss \mathbin{@} t \# ts)$

The *subtree relation* is defined inductively by

$t \in set\ ts \Longrightarrow t \lhd Node\ x\ ts$

$[\![s \lhd t;\ t \in set\ ts]\!] \Longrightarrow s \lhd Node\ x\ ts$

In order to obtain a $\lhd$-minimal $\preceq_{\mathsf{emb}}$-bad sequence we have to prove the properties:

$[\![s \preceq_{\mathsf{emb}} t;\ t \lhd u]\!] \Longrightarrow s \preceq_{\mathsf{emb}} u$
$wf_{\mathcal{T}(A)}(\lhd)$
$[\![s \lhd t;\ t \lhd u]\!] \Longrightarrow s \lhd u$
$[\![s \lhd t;\ t \in \mathcal{T}(A)]\!] \Longrightarrow s \in \mathcal{T}(A)$

all of which are relatively easy.

Our bigger concern was (and rightly so) whether our definition of homeomorphic embedding on trees really corresponds to what is usually used in the literature. To this end, we used the definition of homeomorphic embedding that is used in term rewriting (for first-order terms) as our specification.

**Definition 4.** Let $\mathcal{E}\mathsf{mb}(\preceq)$ be the (infinite) term rewrite system consisting of the following rules:

$$\begin{aligned} f(ts) &\to t && \text{if } t \in set\ ts \\ f(ts) &\to g(ss) && \text{if } g \preceq f \text{ and } ss =_{\mathsf{emb}} ts \end{aligned}$$

We were able (after adapting our initial inductive definition several times) to prove[3] that

$$s \preceq_{\mathsf{emb}} t \longleftrightarrow t \to^{+}_{\mathcal{E}\mathsf{mb}(\preceq)} s$$

which reassures us that our definition is correct. (Note that for this proof we had to use the datatype **datatype** $(\alpha, \beta)$ *term* $=$ *Var* $\beta$ | *Fun* $\alpha$ $((\alpha, \beta)$ *term list*$)$ instead of $\alpha$ *tree*. However, it is easy to see that those two datatypes are isomorphic when disregarding variables.)

Definition 4 is an adaption of the (potentially finite) term rewrite system from [6, Definition 3.4], where we reuse the definition of homeomorphic embedding on lists and avoid variables in order to simplify the above proof.

# 5 Conclusions and Related Work

There have been formalizations of Higman's lemma in other proof assistants [7, 2, 3, 10, 5] and also in Isabelle/HOL [1] (but restricted to a two-letter alphabet of list elements). Those existing formalizations usually strive for constructive proofs, whereas our approach is purely classical. However, to the best of our knowledge our work [11] constitutes the first formalization of Kruskal's tree theorem ever.

Furthermore, [15] could already make use of our formalization of Higman's lemma for formalizing the following: *For an arbitrary language $L$, the set of substrings/superstrings of words in $L$ is regular.*

A final remark, during the whole formalization process one of the key points was to go away from proving facts (about binary predicates) on whole types and instead make the carrier explicit. To this end, predicates like $refl_A(\preceq)$, $trans_A(\preceq)$, $wf_A(\preceq)$, ... have been most helpful and we plead to include them in the standard Isabelle/HOL distribution and introduce their "implicit" cousins (working on whole types) as abbreviations.

[3] See theory `Embedding_Trs` in the IsaFoR repository http://cl-informatik.uibk.ac.at/software/ceta.

**Acknowledgments.** Thanks to Mizuhito Ogawa for helpful discussion and pointing out that the suffix relation is appropriate for constructing a minimal bad sequence inside the proof of Higman's lemma.

# References

[1] Stefan Berghofer. A constructive proof of Higman's lemma in Isabelle. In Stefano Berardi, Mario Coppo, and Ferruccio Damiani, editors, *Types for Proofs and Programs, TYPES 2003*, volume 3085 of *Lecture Notes in Computer Science*, pages 66–82. Springer, 2004. doi:10.1007/978-3-540-24849-1_5.

[2] Daniel Fridlender. Higman's lemma in type theory. In Eduardo Giménez and Christine Paulin-Mohring, editors, *Types for Proofs and Programs, TYPES 1996*, volume 1512 of *Lecture Notes in Computer Science*, pages 112–133. Springer, 1996. doi:10.1007/BFb0097789.

[3] Hugo Herbelin. A program from an A-translated impredicative proof of Higman's lemma, 1994. http://coq.inria.fr/pylons/contribs/view/HigmanNW/v8.3.

[4] Graham Higman. Ordering by divisibility in abstract algebras. *Proceedings of the London Mathematical Society*, s3-2(1):326–336, 1952. doi:10.1112/plms/s3-2.1.326.

[5] Francisco Jesús Martín-Mateos, José Luis Ruiz-Reina, José Antonio Alonso, and María José Hidalgo. Proof pearl: A formal proof of Higman's lemma in ACL2. *Journal of Automated Reasoning*, 47(3):229–250, 2011. doi:10.1007/s10817-010-9178-x.

[6] Aart Middeldorp and Hans Zantema. Simple termination of rewrite systems. *Theoretical Computer Science*, 175(1):127–158, 1997. doi:10.1016/S0304-3975(96)00172-7.

[7] Chetan R. Murthy. *Extracting Constructive Content from Classical Proofs*. PhD thesis, Cornell University, 1990. http://hdl.handle.net/1813/6991.

[8] Crispin St. John Alvah Nash-Williams. On well-quasi-ordering finite trees. *Proceedings of the Cambridge Philosophical Society*, 59(4):833–835, 1963. doi:10.1017/S0305004100003844.

[9] Tobias Nipkow, Lawrence Charles Paulson, and Makarius Wenzel. *Isabelle/HOL - A Proof Assistant for Higher-Order Logic*, volume 2283 of *Lecture Notes in Computer Science*. Springer, 2002. It is strongly recommended to download its updated version, which is part of the documentation of the latest Isabelle release. doi:10.1007/3-540-45949-9.

[10] Monika Seisenberger. *On the Constructive Content of Proofs*. PhD thesis, LMU Munich, 2003. http://nbn-resolving.de/urn:nbn:de:bvb:19-16190.

[11] Christian Sternagel. Well-Quasi-Orders. In Gerwin Klein, Tobias Nipkow, and Lawrence Charles Paulson, editors, *The Archive of Formal Proofs*. http://afp.sf.net/devel-entries/Well_Quasi_Orders.shtml, 2012. Formal proof development.

[12] René Thiemann and Christian Sternagel. Certification of termination proofs using CeTA. In Stefan Berghofer, Tobias Nipkow, Christian Urban, and Makarius Wenzel, editors, *2009*, volume 5674 of *Lecture Notes in Computer Science*, pages 452–468. Springer, 2009. doi:10.1007/978-3-642-03359-9_31.

[13] Wim Veldman and Marc Benzem. Ramsey's theorem and the pigeonhole principle in intuitionistic mathematics. *Journal of the London Mathematical Society*, s2-47(2):193–211, 1993. doi:10.1112/jlms/s2-47.2.193.

[14] Dimitrios Vytiniotis, Thierry Coquand, and David Wahlstedt. Stop when you are almost-full – adventures in constructive termination. In *Interactive Theorem Proving, ITP 2012*. to appear.

[15] Chunhan Wu, Xingyuan Zhang, and Christian Urban. The Myhill-Nerode theorem based on regular expressions. In Gerwin Klein, Tobias Nipkow, and Lawrence Charles Paulson, editors, *The Archive of Formal Proofs*. http://afp.sf.net/entries/Myhill-Nerode.shtml, 2011. Formal proof development.