LOWER BOUNDS ON PROOF LENGTH IN AXIOMATIC THEORIES*

Vladimir Orevkov
Steklov Institute of Mathematics
St. Petersburg
orevkov@pdmi.ras.ru

Abstract The aim of this talk is to obtain lower bounds on proof length in axiomatic theories of algebraically closed and real-closed fields and in arithmetic without multiplication. We will also obtain upper bounds on the complexity of correct proofs in the theory of feasible numbers with induction scheme. The advantage of these bounds (over other author's bounds) is that they do not depend on the complexity of formulas in the proof.

1. Lower bounds on proof length in axiomatic theory of fields

We'll consider algebraically closed fields with characteristic 0 and real-closed ones. Let's fix an axiomatic Hilbert-type theory T (classical or intuitionistic) in language with $=, 0, 1, +$, and $\cdot$. Non-logical axioms of T are a finite list of closed formulas and open formulas

$$\exists x(x^n + t_{n-1} \cdot x^{n-1} + \cdots + t_0 = 0), \tag{1}$$

$$0 \neq \underbrace{1 + 1 + \cdots + 1}_{p \text{ times}}, \tag{2}$$

where $n > 0$, the variable x does not occur in terms $t_0, \ldots, t_{n-1}$; p is any prime number.

In case of real-closed fields the number n in (1) is odd and axioms (2) should be omitted.

The language of Presburger's arithmetic (or arithmetic without multiplication) contains constants 0 and 1, functional symbol $+$, predicates $=, <$. Non-logical axioms of this theory are a finite list of closed formulas and open for-

*Supported by INTAS (grant No. 96-0760) and RFBR (grants No. 94-01-01030 and 96-01-01612).

P. Gärdenfors, J. Woleński and K. Kijania-Placek (eds.), In the Scope of Logic, Methodology and Philosophy of Science, Vol. I, 17-26.

mulas

$$\exists yz(t = \underbrace{y+y+\cdots+y}_{n \text{ times}} + z \& z < \underbrace{1+1+\cdots+1}_{n \text{ times}}), \tag{3}$$

where $n > 0$, variables y and z do not occur in term t.

Let D be a proof in T. The *length* of D is the number of different formulas in D. The length of D will be denoted by $l[D]$. The expression

$$T \vdash_k A$$

means there exists a proof D of A in T in which

$$l[D] \leqslant k.$$

Theorems 1–3 following below are examples of generalization of proofs.

Theorem 1 *For any formula $A(x)$, any natural number m and any sufficiently large natural number n, if for any number i $(0 \leqslant i \leqslant n)$*

$$\mathbf{T} \vdash_{\frac{\log_2 n}{3\log_2\log_2 n}} A(\mathbf{0}_{m+i}),$$

then

$$\mathbf{T} \vdash \forall x A(x),$$

where **T** *is the theory of algebraically closed fields with characteristic* 0 *or the theory of real-closed ones or arithmetic without multiplication. Here the expression $\mathbf{0}_n$ denotes the term*

$$\underbrace{0+0+\cdots+0}_{n \text{ times}}.$$

Below we will use the following notation:

$$\mathbf{1}_n \rightleftharpoons \begin{cases} \underbrace{1+1+\cdots+1}_{n \text{ times}}, & \text{if } n \geqslant 1, \\ 0, & \text{if } n = 0. \end{cases}$$

Theorem 2 *For any formula $A(x)$, any natural number m and any sufficiently large natural number n, if for any number i $(0 \leqslant i \leqslant n)$*

$$\mathbf{T} \vdash_{\frac{\log_2 n}{3\log_2\log_2 n}} A(\mathbf{1}_{m+i}),$$

then there exist natural numbers $l_1, l_2, \ldots, l_k$ such that

$$\mathbf{T} \vdash \forall x \left(\left(\bigwedge_{i=1}^{k} (0 \neq \mathbf{1}_{l_i} + x) \right) \supset A(x) \right),$$

where **T** *is the theory of algebraically closed fields with characteristic 0.*

Theorem 3 *For any formula* $A(x)$, *any natural number* m *and any sufficiently large natural number* n, *if for any number* i $(0 \leqslant i \leqslant n)$

$$\mathbf{T} \vdash_{\frac{\log_2 n}{3\log_2 \log_2 n}} A(\mathbf{1}_{m+i}),$$

then there exists a natural number k *such that* $k \leqslant m + n$ *and*

$$\mathbf{T} \vdash \forall x((x > \mathbf{1}_k) \supset A(x)),$$

where **T** *is arithmetic without multiplication.*

In the proofs of theorems 1–3 we use some lemmas and estimates from Orevkov (1993) and the following section.

Theorems 1–3 can be used to obtain lower bounds on proof length.

The expression

$$\mathbf{T} \nvdash_k A$$

means the negation of the assertion $\mathbf{T} \vdash_k A$.

Theorem 4 *For infinitely many natural numbers* n, *the following conditions hold*

$$\mathbf{T} \nvdash_{\frac{\log_2 n}{3\log_2 \log_2 n}} 0 = \mathbf{0}_n,$$

$$\mathbf{T} \nvdash_{\frac{\log_2 n}{3\log_2 \log_2 n}} (\mathbf{1}_n + \mathbf{1}_n) \neq 1,$$

$$\mathbf{T} \nvdash_{\frac{\log_2 n}{3\log_2 \log_2 n}} (\mathbf{1}_n \cdot \mathbf{1}_n) \neq 1 + 1,$$

where **T** *is the theory of algebraically closed fields with characteristic 0 or the theory of real-closed ones or arithmetic without multiplication.*

We can also obtain upper bounds on proof length.

Theorem 5 *There is a natural number* c *such that for any* n

$$\mathbf{T} \vdash_{c\log_2 n} 0 = \mathbf{0}_n,$$

$$\mathbf{T} \vdash_{c\log_2 n} (\mathbf{1}_n + \mathbf{1}_n) \neq 1,$$

$$\mathbf{T} \vdash_{c\log_2 n} (\mathbf{1}_n \cdot \mathbf{1}_n) \neq 1 + 1,$$

where **T** *is the theory of algebraically closed fields with characteristic 0 or the theory of real-closed ones or arithmetic without multiplication.*

FEW NON MINIMAL TYPES ON NON STRUCTURE*

SH603

Saharon Shelah
Institute of Mathematics, The Hebrew University
Jerusalem, Israel
Rutgers University, Department of Mathematics
New Brunswick, NJ USA

Abstract We deal with abstract elementary classes $\mathfrak{K}$ which has amalgamation in λ. Our main result is that if ($2^\lambda < 2^{\lambda^+}$ and) the minimal types on members of K_λ are not dense (among non algebraic (complete) types over models in K_λ, extending our given model), then the number of models in K of cardinality λ^+ or λ^{++} is maximal. For this we deal with some claims in pcf. This improves a result in Shelah (2001), but the amount of relying is small, mostly of a "black box" character.

Keywords: model theory, abstract elementary classes, classification theory, categoricity, nonstructure theory, pcf theory

Annotated Content

0. Introduction [We explain our aim and define our framework.]

1. Non minimal types and nonstructure [We define unique amalgamation, UQ, and try to use it for building many models in λ^+ when $2^\lambda < 2^{\lambda^+}$ (so the weak diamond holds). If this approach fails we still get the many models in λ^{++} by the "easy" criterion of Shelah (2001, §3) but it works only if the weak diamond ideal on λ^+ is not λ^{++}-saturated.]

2. Remarks on pcf [We prove some pcf observations needed here.]

*I thank Alice Leonhardt for the beautiful typing
Split from Sh600; work done Spring of 1995
Latest version - 2000/Apr/6

P. Gärdenfors, J. Woleński and K. Kijania-Placek (eds.), In the Scope of Logic, Methodology and Philosophy of Science, Vol. I, 29-53.

3. Finishing the many models [We prove the result of Section 1 without the extra assumption on the saturation of the weak diamond ideal.]

4. A minor debt [There was one point in Shelah (2001) where we use $\lambda > \aleph_0$, though our aim there was to generalize theorem known for $\lambda = \aleph_0$. We eliminate this use.]

0. Introductions

In Shelah (2001) there was an important point where we used as assumption $I(\lambda^{+3}, K) = 0$. This was fine for the purpose there, but is unsuitable in other frameworks, like Shelah (200x): we want to analyze what occurs in higher cardinals, so our main aim here is to eliminate its use and add to our knowledge on non-structure.

The point was "the minimal triples in K^3_λ are dense" (Shelah 2001, 3.17t). For this we assume we have a counterexample, and try to build many nonisomorphic models. Hence we get cases of amalgamation which are necessarily unique. Those "unique amalgamations" are normally too strong (even for first order superstable theories), but here they help us to prove positive theorems, controlling omitting types. So we try to build many models in λ^+ by omitting "types" over models of size λ, in a specific way where unique amalgamation holds. If this argument fails, we prove $\boldsymbol{C}^1_{\mathfrak{K},\lambda}$ has weak λ^+-coding (see Shelah 2001, §3) and by it get $2^{\lambda^{++}}$ non-isomorphic models except when the weak diamond ideal on λ^+ is λ^{++}-saturated; this is done in Section 1. In Section 3 we work harder and by partition to cases relying on pcf theory we succeed to get the full result. We work also to get large IE (many models no one $\leqslant_{\mathfrak{K}}$-embedding to another). The pcf lemmas (which are pure infinite combinatorics) are dealt with in Section 2.

There was also another point left in Shelah (2001, 4.2t), for the case $\lambda = \aleph_0$ only, this is filled in Section 4.

* * *

Definition 0.1 We say $\mathfrak{K} = (K, \leqslant_{\mathfrak{K}})$ is an abstract elementary class, aec or a.e.c. in short, if ($\tau = \tau_K$ is a fixed vocabulary, K a class of τ-models (and *Ax* 0 holds and)) *Ax I* − *VI* hold where:

Ax 0: The holding of $M \in K, N \leqslant_{\mathfrak{K}} M$ depends on N, M only up to isomorphism i.e. [$M \in K, M \cong N \Rightarrow N \in K$], and [if $N \leqslant_{\mathfrak{K}} M$ and f is an isomorphism from M onto the τ-model M' mapping N onto N' then $N' \leqslant_{\mathfrak{K}} M'$].

Ax I: If $M \leqslant_{\mathfrak{K}} N$ then $M \subseteq N$ (i.e. M is a submodel of N).

Ax II: $M_0 \leqslant_{\mathfrak{K}} M_1 \leqslant_{\mathfrak{K}} M_2$ implies $M_0 \leqslant_{\mathfrak{K}} M_2$ and $M \leqslant_{\mathfrak{K}} M$ for $M \in K$.

Ax III: If λ is a regular cardinal, $M_i (i < \lambda)$ is $\leqslant_{\mathfrak{K}}$-increasing (i.e. $i < j < \lambda$ implies $M_i \leqslant_{\mathfrak{K}} M_j$) and continuous (i.e. for limit ordinal $\delta < \lambda$ we have $M_\delta = \bigcup_{i<\delta} M_i$) then $M_0 \leqslant_{\mathfrak{K}} \bigcup_{i<\lambda} M_i$.

Ax IV: If λ is a regular cardinal, $M_i (i < \lambda)$ is $\leqslant_{\mathfrak{K}}$-increasing continuous, $M_i \leqslant_{\mathfrak{K}} N$ then $\bigcup_{i<\lambda} M_i \leqslant_{\mathfrak{K}} N$.

Ax V: If $M_0 \subseteq M_1$ and $M_\ell \leqslant_{\mathfrak{K}} N$ for $\ell = 0, 1$, then $M_0 \leqslant_{\mathfrak{K}} M_1$.

Ax VI: $LS(\mathfrak{K})$ exists[1], where $LS(\mathfrak{K})$ is the minimal cardinal λ such that: if $A \subseteq N$ and $|A| \leqslant \lambda$ then for some $M \leqslant_{\mathfrak{K}} N$ we have $A \subseteq |M| \leqslant \lambda$ and we demand for simplicity $|\tau| \leqslant \lambda$.

Notation 0.2 1) $K_\lambda = \{M \in K : \|M\| = \lambda\}$ and $K_{<\lambda} = \bigcup_{\mu<\lambda} K_\mu$.
See more in Shelah (2001, §0).

Definition 0.3

1) For $\mu \geqslant LS(\mathfrak{K})$ and $M \in K_\mu$ we define $\mathscr{S}(M)$ as $\{\mathrm{tp}(a, M, N) : M \leqslant_{\mathfrak{K}} N \in K_\mu \text{ and } a \in N\}$
where $\mathrm{tp}(a, M, N) = (M, N, a)/E_M$ where E_M is the transitive closure of E_M^{at}, and the two-place relation E_M^{at} is defined by:

$$(M, N_1, a_1) E_M^{at} (M, N_2, a_2) \text{ iff there is } N \in K_\mu \text{ and } \leqslant_{\mathfrak{K}}\text{-embeddings}$$
$$f_\ell : N_\ell \to N \text{ for } \ell = 1, 2 \text{ such that:}$$
$$f_1 \restriction M = \mathrm{id}_M = f_2 \restriction M \text{ and } f_1(a_1) = f_2(a_2).$$

(of course $M \leqslant_{\mathfrak{K}} N_1, M \leqslant_{\mathfrak{K}} N_2$ and $a_1 \in N_1, a_2 \in N_2$)

2) We say "a realizes p in N" if $a \in N, p \in \mathscr{S}(M)$ and for some $N' \in K_\mu$ we have $M \leqslant_{\mathfrak{K}} N' \leqslant_{\mathfrak{K}} N$ and $a \in N'$ and $p = \mathrm{tp}(a, M, N')$; so $M, N' \in K_\mu$ but possibly $N \notin K_\mu$.

3) We say "a_2 strongly realizes $(M, N^1, a^1)/E_M^{at}$ in N" if for some N^2, a^2 we have $M \leqslant_{\mathfrak{K}} N^2 \leqslant_{\mathfrak{K}} N$ and $a_2 \in N^2$ and $(M, N^1, a^1) E_M^{at} (M, N^2, a^2)$.
(Note: if M_0 is an amalgamation base, see below, then the difference between realize and strongly realize disappears).

4) We say $M_0 \in \mathfrak{K}_\lambda$ is an amalgamation base if: for every $M_1, M_2 \in \mathfrak{K}_\lambda$ and $\leqslant_{\mathfrak{K}}$-embeddings $f_\ell : M_0 \to M_\ell$ (for $\ell = 1, 2$) there is $M_3 \in \mathfrak{K}_\lambda$ and $\leqslant_{\mathfrak{K}}$-embeddings $g_\ell : M_\ell \to M_3$ (for $\ell = 1, 2$) such that $g_1 \circ f_1 = g_2 \circ f_2$.

5) We say $\mathfrak{K}$ is stable in λ if $LS(\mathfrak{K}) \leqslant \lambda$ and $M \in K_\lambda \Rightarrow |\mathscr{S}(M)| \leqslant \lambda$.

6) We say N is λ-universal over M if for every $M', M \leqslant_{\mathfrak{K}} M' \in K_\lambda$, there is a $\leqslant_{\mathfrak{K}}$-embedding of M' into N over M. If we omit λ we mean $\|N\|$.

MINIMISATION VS. RECURSION ON THE PARTIAL CONTINUOUS FUNCTIONALS

Ulrich Berger
Department of Computer Science
University of Wales Swansea

Abstract We study the relationship between partial continuous higher type functionals defined by minimisation on the one hand and recursion on the other hand. We prove that already at type level two minimisation is weaker than recursion.

1. Introduction

There are two well-known ways of extending the schemata for the primitive recursive functions such that all partial recursive functions are obtained. One is recursion the other is minimisation. In Kleene (1959a) both are extended to higher types: recursion via the schemata (S1-S9) and minimisation via μ-recursion. Kleene showed that at type two (S1-S9) and μ-recursion coincide, but already at type three the latter is weaker than the former. In Bergstra (1976) it is proved that also on the total continuous functionals (Kleene 1959b, Kreisel 1959) the schemata (S1-S9) are weaker than recursive continuity (having a recursive associate). These results were based on an interpretation of computations on *total* (continuous) functionals, whereas it is now common to interpret them on the *partial* continuous functionals. Normann showed that this makes a difference by proving that under the new interpretation (S1-S9) is as powerful as recursive continuity when restricted to the total continuous functional (Normann 2000). To be precise Normann works with the functional language PCF (Plotkin 1977), which however, as essentially shown in Platek (1966), is equivalent to (S1-S9) on the partial continuous functionals.

In this paper it is shown that the relationship between minimisation and recursion is affected by the choice of the underlying domain, too. We prove that already at type level two minimisation is weaker than recursion. From our result we deduce that at those types minimisation is not only denotationally but also operationally weaker than PCF.

P. Gärdenfors, J. Woleński and K. Kijania-Placek (eds.), In the Scope of Logic, Methodology and Philosophy of Science, Vol. I, 57-64.

After giving in section 2 the basic definitions we present in 3 as a starter a new short proof that every computable monotone functions of type level one is μ-recursive in the parallel OR, a result first proved in Trakhtenbrot (1976). Section 4 contains the main result already discussed above, and in section 5 we conclude by discussing some open problems arising at higher types.

2. Partial continuous functionals

The hierarchy of *partial continuous functionals* is a family of effective Scott-Ershov-domains (Scott 1982, Griffor, Lindström and Stoltenberg-Hansen 1993) defined by

$$D_\iota := \mathbf{N}_\perp, \quad D_o := \mathbf{B}_\perp, \quad D_{\rho\to\sigma} := D_\rho \to D_\sigma,$$

where $\mathbf{N}_\perp := \mathbf{N} \cup \{\perp\}$ and $\mathbf{B}_\perp := \mathbf{B} \cup \{\perp\}$ ($\mathbf{B} = \{\mathrm{tt}, \mathrm{ff}\}$) are the flat domains of partial integers and boolean values respectively. $D \to E$ denotes the domain of continuous functions from D to E, i.e. the exponential in the cartesian closed category of Scott-Ershov domains and continuous functions. We let $\mathcal{D}$ be the union of the D_ρ. By a *functional* (of type ρ) we mean an element of $\mathcal{D}$ (D_ρ). When writing $f\colon \rho$ we mean that f is a functional of type ρ. A functional of type ρ is called *computable* if it is a computable element of the effective Scott-Ershov domain D_ρ, i.e. its compact approximations are recursively enumerable.

We let τ range over the base types ι and o. A type $\rho_1 \to (\rho_2 \to \ldots \to (\rho_n \to \sigma)\ldots)$ will often be written $\rho_1 \to \rho_2 \to \ldots \to \rho_n \to \sigma$ or simply $\vec{\rho} \to \sigma$. An iterated application $f(x_1)\ldots(x_n)$ will also be written $fx_1\ldots x_n$ or $f(x_1,\ldots,x_n)$. For $\rho = \vec{\rho} \to \tau$ we define $\perp_\rho\colon \rho$ by $\perp_\rho(\vec{x}) := \perp$. The level of a type ρ is defined as usual by $\mathrm{lev}(\iota) = \mathrm{lev}(o) := 0$, $\mathrm{lev}(\rho \to \sigma) := \max(\mathrm{lev}(\rho)+1, \mathrm{lev}(\sigma))$. The level of a functional $f \in D_\rho$ is the level of ρ. A functional $g \in \mathcal{D}$ is *explicitly definable* from a set of functionals $\mathcal{F} \subseteq \mathcal{D}$ if g can be defined from elements of $\mathcal{F}$ by application and λ-abstraction.

In the following we will frequently omit type information as long as it can be recovered from the context.

A functional of type level $\leqslant 1$ will be called a *monotone function*. The following monotone functions will be used frequently (see also Plotkin (1977)). We let i, j, m, n, k range over natural numbers.

$\mathrm{if}_\tau\colon o \to \tau \to \tau \to \tau$	$\mathrm{if}_\tau(\mathrm{tt}, x, y) = x$, $\mathrm{if}_\tau(\mathrm{ff}, x, y) = y$, $\mathrm{if}_\tau(\perp, x, y) = \perp$
$\mathrm{Z}\colon \iota \to o$	$\mathrm{Z}(0) := \mathrm{tt}$, $\mathrm{Z}(n+1) := \mathrm{ff}$, $\mathrm{Z}(\perp) := \perp$
$(+1)\colon \iota \to \iota$	$(+1)(n) := n+1$, $(+1)(\perp) := \perp$
$(-1)\colon \iota \to \iota$	$(-1)(0) := 0$, $(-1)(n+1) := n$, $(-1)(\perp) := \perp$

The set $\{0, \mathrm{if}, \mathrm{Z}, (+1), (-1)\}$ is called the set of *basic functions*.

Recursive definitions in $\mathcal{D}$ are modelled by the *fixed point operators* $Y_\rho\colon (\rho \to \rho) \to \rho$,

$$Y(f) := \text{ the least fixed point of } f = \bigsqcup\{f^n(\bot_\rho) \mid n \in \mathbf{N}\}.$$

A functional is definable in the functional programming language PCF (Plotkin 1977) iff it is explicitly definable from the basic functions and the fixed point operators.

A functional is *μ-recursive* if it is explicitly definable from the basic functions, the *primitive recursor* $R\colon \iota \to (\iota \to \iota \to \iota) \to \iota \to \iota$,

$$R(x,h,0) := x, \quad R(x,h,n+1) := h(n,R(x,h,n)), \quad R(x,h,\bot) := \bot,$$

and the *minimisation functional* $\mu\colon (\iota \to o) \to \iota$,

$$\mu(f) := n \text{ if } f(n) = \mathrm{tt} \text{ and } f(i) = \mathrm{ff} \text{ for } i < n, \quad \mu(f) := \bot \text{ otherwise}$$

A functional g is *μ-recursive in a set of functionals* $\mathcal{F}$ if there are $f_1, \ldots, f_k \in \mathcal{F}$ and a μ-recursive functional h such that $g = h(f_1, \ldots, f_n)$. The notions '*PCF-definable in*' and '*computable in*' are defined similarly.

The following implications are well-known:

$$\mu\text{-recursive} \quad \Rightarrow \quad \text{PCF-definable} \quad \Rightarrow \quad \text{computable}$$

Of course these implications also hold when relativised to a set of functionals.

A *strict function* is a monotone function f such that $f(\vec{x}) = \bot$ whenever $x_i = \bot$ for some i. It follows from ordinary recursion theory that a strict function is computable iff it is μ-recursive.

3. Monotone functions

In Sazonov (1975) it is proved that a monotone function is PCF-computable iff it is an *effectively sequential function* (Vuillemin 1975). In Trakhtenbrot (1976) it is shown that the effectively sequential functions are precisely those definable from computable strict functions and if_τ by composition. Since computable strict functions are μ-recursive we have:

Theorem 1 (Sazonov, Trakhtenbrot) *A monotone function is PCF-definable iff it is μ-recursive.*

The monotone function *parallel or* $OR\colon o \to o \to o$ is defined by

$$OR(\mathrm{tt},x) = OR(x,\mathrm{tt}) = \mathrm{tt}, OR(\mathrm{ff},\mathrm{ff}) = \mathrm{ff}, OR(x,y) = \bot \text{ otherwise.}$$

Note that $OR(\mathrm{tt},\bot) = OR(\bot,\mathrm{tt}) = \mathrm{tt}$, but $OR(\bot,\bot) = \bot$. Hence OR is not sequential and therefore not PCF-definable (although computable, trivially).