CHAPTER VIII

The Lengths of Proofs

Pavel Pudlák
Mathematical Institute, Academy of Sciences of the Czech Republic
115 67 Prague 1, The Czech Republic

Contents

HANDBOOK OF PROOF THEORY
Edited by S. R. Buss

1. Introduction

In this chapter we shall consider the problem of determining the minimal complexity of a proof of a theorem in a given proof system. We shall deal with propositional logic and first order logic. There are several measures of complexity of a proof and there are many different proof systems. Let us give some reasons for this research, before we discuss particular instances of the problem.

1.1. Our subject could be called *the quantitative study of the proofs*. In contrast with the classical proof theory we want to know not only whether a theorem has a proof but also whether the proof is feasible, i.e., can be actually written down or checked by a computer. An ideal justification for such research would be a proof that a particular theorem for which we have only long proofs (such as the four color theorem), or a conjecture for which we do not have any proof (such as $\mathcal{P} \neq \mathcal{NP}$), does not have a short proof in some reasonable theory (such as **ZF**). Presently this seems to be a very distant goal; we are only able to prove lower bounds on the lengths of proofs for artificial statements, or for natural statements, but in very weak proof systems. The situation here is similar to the situation in the study of (weak) fragments of arithmetic and complexity theory. In fragments of arithmetic we can prove unprovability of Π_1^0 sentences only for sentences obtained by diagonalization, and in complexity theory we can separate complexity classes also only when diagonalization is possible. These three areas are very much connected and it is not possible to advance very much in one of them without making progress in the others.

Nevertheless there are already now some practical consequences of this research. For instance in first order logic we know quite precisely how much cut-elimination increases the size of proofs. In propositional logic we have simple tautologies which have only exponentially long resolution proofs. This is very important information for designers of automated theorem provers.

Another reason for studying the lengths of proofs is that information about the size of proofs is very important in the study of weak fragments of arithmetic, namely when metamathematics of fragments is considered. For instance, in bounded arithmetic the exponentiation function is not provably total. Therefore the cut-elimination theorem is not provable in bounded arithmetic (in fact first order cut-elimination requires more than elementary increase in the size of proofs). Consequently we have (at least) two different concepts of consistency in bounded arithmetic: the usual one and cut-free consistency.

Furthermore there is a relation between provability in bounded arithmetic and the lengths of proofs in certain proof systems for propositional logic. This seems to be the most promising way of proving concrete independence results for bounded arithmetic.

Finally this area is important because of its tight relation to complexity theory. Actually, research into the lengths of proofs should be considered as a part of complexity theory. There are two kinds of connections with computational complexity.

On the one hand there are explicit connections such as the fact that a proof system for propositional logic is a nondeterministic algorithm for the ($co\mathcal{NP}$ complete) set of tautologies. On the other hand there are intuitive connections which are not supported by theorems. For example the relation between Frege systems and extension Frege systems (see below for definitions) for propositional logic is very much like the relation between the complexity measures of boolean functions based on formula size and circuit size, respectively. It is an open problem whether Frege systems are as powerful as extension Frege systems and also it is an open problem whether formulas are as powerful as circuits; but we are not able to prove any of two implications between these apparently related problems. Some people think that the difficult problems in complexity theory such as $\mathcal{P} = \mathcal{NP}$? are essentially logical (not combinatorial) problems. If it is so, then proof theory, and in particular the lengths of proofs, should play an important role in their solution.

1.2. Now we shall briefly outline the contents of this chapter. Section 2 introduces some basic concepts. In section 3 we describe a technique of constructing short formulas for inductively defined concepts. This technique has various applications. Section 4 contains results about dependence of different measures of complexity of proofs and a remark on the popular Kreisel Conjecture. In section 5 we shall consider the cut-elimination theorem from the point of view of the lengths of proofs; namely, we shall show a lower bound on the increase of the length. In section 6 we shall prove a version of the second incompleteness theorem for finite consistencies. This enables us to prove some concrete lower bounds and speed-up. In section 7 we survey speed-up theorems, namely results about shortening of proofs when a stronger theory is used instead of a weaker one and related results. Section 8 is a survey of the most important propositional proof systems. In section 9 we give a nontrivial example of a lower bound on the lengths of propositional proofs in the resolution system. In section 10 we present important relations between the lengths of proofs in propositional logic and provability in fragments of arithmetic. The final section 11 surveys especially those results which have not been treated in the main text.

2. Types of proofs and measures of complexity

In this section we introduce notation and some basic concepts used in both propositional logic and first order logic.

2.1. One can consider many different formalizations and it is difficult to find a classification schema which would cover all. There is however one basic property which all formalizations of the concept of a proof must satisfy: it must be computable in polynomial time whether a given sequence is a proof of a given formula. Here we assume, as usual, that proofs and formulas are encoded as strings in a finite alphabet and we identify feasible computations with polynomial time computations. This trivial observation gives us important link to computational complexity. The proof systems in such a general setting are just nondeterministic decision procedures

for the set of tautologies or the set of theorems of a theory in question. More specifically, an upper bound on the size of proofs for a particular proof system gives a nondeterministic decision procedure with the bound on the running time and, conversely, a lower bound on the nondeterministic time complexity is a lower bound for any proof system.

In particular, let $TAUT$ be the set of propositional tautologies in some fixed complete basis of connectives. A *propositional proof system* is a binary relation $P(x, y)$ which is *computable in polynomial time* and

$$\varphi \in TAUT \equiv \exists y\, P(\varphi, y).$$

Since the set of propositional tautologies is $\mathcal{NP}$-complete, we get the following immediate corollary.

2.1.1. Theorem. (Cook and Reckhow [1979]) *There exists a proof system for propositional logic in which all tautologies have proofs of polynomial length if and only if $\mathcal{NP} = co\mathcal{NP}$.* □

This general concept of a proof system can be generalized even further. Firstly, we can allow randomized computations; secondly, we can assume that the proof is not given to us, but we can access parts of the proof via an oracle. Usually such an *interactive proof system* is presented as a two player game, where we are the *Verifier* and the oracle is the *Prover.* It turns out that the Verifier can check with high probability that a proof for a given formula exists without learning almost anything about the proof. The most striking example is the so-called PCP theorem by Arora et al. [1992]. Roughly speaking, they showed, that there there are interactive proof systems, where the Verifier needs to check only a constant number of randomly selected bits of the sequence in order to check with high probability that the proof is correct.

Note, however, that these results concern only the question *how can be proofs checked* but do not give new information about *the lengths of proofs.*

2.2. We turn now to more structured proofs, which are typical for logic, while the above concepts rather belong to complexity theory. Such proof systems are usually defined using a finite list of *deduction rules.* The basic element of a proof, called *a proof step,* or *a proof line*, is a formula, a set of formulas, a sequence of formulas or a sequent (pair of sequences of formulas). A *proof* is either a *sequence* or a *tree* of proof steps such that each step is an axiom or follows from previous ones by a deduction rule. The complete information about the intended way of proving a given theorem should also contain the information for each step of which rule is applied and to which previous steps it is applied. However in most cases this does not influence the complexity of the proofs essentially.

It is important to realize that when proof lines and deduction rules are determined, there are two possible forms of proofs: *the tree form* and *the sequence form.* In the tree form, a proof line may be a premise of an application of a rule only once, while

in the sequence form it can be used again and again. The trivial transformation from the sequence form to tree form results in exponential increase of size.

The most important measure of complexity of proofs is the *size* of a proof. We take a finite alphabet and a natural encoding of proofs as sequences (words) in a finite alphabet. Then the size of a proof is the length of its code.

The next one is the *number of proof lines*. Trivially, the number of proof lines is at most the size, however, a proof may contain very large formulas, thus there is an essential difference between the two measures.

Quite often it is important to bound the maximal complexity of formulas in the proof. Usually we consider the quantifier complexity or the number of logical symbols. Thus we get other measures.

Comparing the above measures with the complexity measures in computational complexity we see that the size corresponds clearly to time. At first glance it may seem that the maximal size of a formula (or proof line) should correspond to space, but this is not correct. In order to present a proof in a lecture, or to check it on a computer we cannot show a single formula (proof line) at a time, we have to keep the formulas (lemmas) on the blackboard until they are used for the last time as premises. The minimal size of a blackboard on which the proof can be presented is the right concept corresponding to space. Note that a suitable choice of the concept of a proof line and rules leads to *linear proofs*, where each rule has at most one premise (Craig [1957a]). In such proofs the maximal size of a proof line is the measure corresponding to space.

In first order logic we consider also the proofs in a *theory* T. This means that we can use axioms of T in proofs. Talking about theories is not quite precise here; different axiomatizations give clearly different concepts of proofs and hence the smallest size proofs of a given formula may be different. Therefore we shall use preferably the term *axiomatization*.

2.3. We shall use the following notation. The size of a formula φ resp. a proof d will be denoted by $|\varphi|$ resp. $|d|$. Let A be a proof system or a proof system plus an axiomatization of a theory. Then we write $d : A \vdash \varphi$, if d is a proof of φ in A; $A \vdash \varphi$, if $\varphi$is provable in A; and $A \vdash^n \varphi$, if φ has a proof of size $\leq n$ in A. Note that the same notation is often used for the number of proof steps. We shall distinguish it by writing $A \vdash^n_{\text{steps}} \varphi$. Often it is more convenient to use the alternative notation:

$$\|\varphi\|_A = \begin{cases} \text{minimal } n \text{ such that } A \vdash^n \varphi \text{ if } A \vdash \varphi \\ \infty \text{ otherwise.} \end{cases}$$

This enables us to write inequalities such as

$$\|\psi\|_A \leq \|\varphi\|_A + \|\varphi \to \psi\|_A + |\psi| + O(1),$$

which holds in the presence of modus ponens in A.

2.4. Suppose that we consider a particular logical calculus. In propositional logic, this simply means that we fix a set of connectives; in first order logic, this means that we fix a language and, possibly consider some theory. Then we can compare the power of different proof systems with respect to the complexity of proofs. If we consider the *size* of proofs, then it is quite natural to disregard polynomial differences in proofs. In particular we define $P_1 \preceq P_2$, if there exists a polynomial $p(x)$ such that for each tautology (resp. theorem) φ, if $d_1 : P_1 \vdash \varphi$, then for some $d_2, |d_2| \leq p(|d_1|), d_2 : P_2 \vdash \varphi$, (using the norm notation: $\|\varphi\|_{P_1} \leq p(\|\varphi\|_{P_2})$).

Usually, if $P_1 \preceq P_2$, then there exists a polynomial time algorithm to construct d_2 from d_1; in such a situation we say that P_2 *polynomially simulates* P_1, (see Cook and Reckhow [1979]). We say that P_1 is *polynomially equivalent* to P_2, if P_1 and P_2 polynomially simulate each other.

A well-known theorem of Craig states that a theory has a recursive axiomatization, if it is recursively enumerable. It is an easy exercise to prove the following modification of the theorem.

2.4.1. Proposition. *Let P_1 be an arbitrary proof system for a calculus with the connective of implication. Then there exists a polynomially equivalent calculus P_2 based on a polynomial time decidable set of axioms and the single rule of* modus ponens. □

Consequently one has to consider stronger assumptions in order to restrict the class of proof systems. The usual approach is to work with the *schematic theories* of Parikh [1973], where we have a *finite* set of rules and *axiom schemas*.

2.5. We shall conclude this section by presenting the most often used proof systems for first order logic; we consider those used in mathematical logic, there are several others used in artificial intelligence, see Chang and Lee [1973] and Eder [1992].

2.5.1. Gentzen [1935] attributes the following system to Hilbert and Glivenko:

Rules

(6.1) $$\frac{A, A \to B}{B}$$

(6.2) $$\frac{A \to \Phi(x)}{A \to \forall y \Phi(y)},$$ where x does not occur in A,

(6.3) $$\frac{\Phi(x) \to A}{\exists x \Phi(x) \to A},$$ where x does not occur in A.

Axiom Schemas

(1.1) $A \to A$
(1.2) $A \to (B \to A)$
(1.3) $(A \to (A \to B)) \to (A \to B)$
(1.4) $(A \to (B \to C)) \to (B \to (A \to C))$
(1.5) $(A \to B) \to ((B \to C) \to (A \to C))$
(2.1) $(A \wedge B) \to A$
(2.2) $(A \wedge B) \to B$
(2.3) $(A \to B) \to ((A \to C) \to (A \to (B \wedge C)$
(3.1) $A \to (A \vee B)$
(3.2) $B \to (A \vee B)$
(3.3) $(A \to C) \to ((B \to C) \to ((A \vee B) \to C))$
(4.1) $(A \to B) \to ((A \to \neg B) \to \neg A)$
(4.2) $\neg A \to (A \to B)$
(5.1) $\forall x \Phi(x) \to \Phi(t)$
(5.1) $\Phi(t) \to \exists x \Phi(x)$

(t stands for a term, x, y are variables).

We shall refer to this calculus as the *Hilbert style*calculus.Note that in a system such as above we can either say that we have *axiom schemas* or that we have *axioms and allow the substitution rule to be applied only to axioms.* We shall consider the power of various proof systems for propositional logic in section 8. The propositional part of the Hilbert style system is a special case of calculi called Frege systems. Contrary to the history, the general substitution rule is not permitted in Frege systems.

There are more compact Hilbert style systems, e.g. the one considered by Hilbert and Ackermann [1928], use only the connectives $\vee$ and $\neg$. As we shall see, the propositional parts simulate each other and (unless we use some strange quantifier rules) this can be extended to the whole systems.

Let us note that there are natural proof systems for first order logic which have only modus ponens as a rule and the quantifier rules are replaced by a finite number of simple axiom schemas, see e.g. Grzegorczyk [1974].

2.5.2. Another important system has been introduced by Gentzen [1935]. The basic elements of the proof are *sequents* which are sequences $\varphi_1, \ldots, \varphi_n \Leftrightarrow \psi_1, \ldots, \psi_m$. Here $\Leftrightarrow$ is a syntactical symbol, a different symbol $\to$ is used for implication. The interpretation of such a sequent is $\varphi_1 \wedge \ldots \wedge \varphi_n \to \psi_1 \vee \ldots \vee \psi_m$ (with $\to$ standing now for implication). The system has a single axiom scheme $A \Leftrightarrow A$, where A is a formula, and several rules which have one or two sequents as assumptions and one sequent as a conclusion. A proof is a tree of sequents where leaves are instance of the axiom and every other sequent follows from its predecessors by a rule. The tree structure is very convenient for analyzing proofs, but one can also consider sequences

of sequents as a proof. The most important rule as the *cut rule*

$$\frac{\alpha_1, \dots, \alpha_k \Leftrightarrow \beta_1, \dots, \beta_l, \varphi \qquad \varphi, \gamma_1, \dots, \gamma_m \Leftrightarrow \delta_1, \dots, \delta_n}{\alpha_1, \dots, \alpha_k, \gamma_1, \dots, \gamma_m \Leftrightarrow \beta_1, \dots, \beta_l, \delta_1, \dots, \delta_n}$$

Observe that for $k = l = m = 0, n = 1$ we get essentially modus ponens. The whole system is described in Chapter I. Gentzen presented transformations of proofs from the Hilbert style calculus to his sequent calculus and vice versa. In Eder [1992] it is shown that this in fact gives polynomial simulations of the systems if

1. in both we take tree-proofs or
2. in both we take sequence-proofs.

In section 4 we shall show that there is also polynomial simulation of sequence-proofs by tree-proofs in the Hilbert style calculus. Thus the most commonly used systems are polynomially equivalent.

The systems above are prototypes of what is called *a schematic theory*. This concept is a natural extension of the concept of the Frege system used in propositional logic. In first order logic, however, it is not easy to define precisely such a concept especially because restrictions on occurrences of variables in quantifier rules (or axioms) are needed. For possible definitions of schematic theories see Vaught [1967], Parikh [1973], Krajíček [1989a], Farmer [1984,1988] and Buss [1994].

Hilbert's *ε-calculus* is based on a different language. Instead of quantifiers it uses ε-terms $\varepsilon_{\varphi(x)}$ whose meaning is an element which satisfies the formula $\varphi(x)$ if there is any, otherwise $\varepsilon_{\varphi(x)}$ is an arbitrary element. This system is described in the famous book of Hilbert and Bernays [1934,1939]. Other popular systems are Beth's system of *semantic tableaux*, described in Beth [1959] and Smullyan [1968], and Prawitz's *natural deduction* system, described in Prawitz [1970] and Girard [1989]. Brief descriptions of these systems can be found in Chapter I of this volume.

For most systems polynomial simulations have been found and it seems very likely that mutual polynomial simulations with other systems can be found. Thus I do not expect that interesting results on the length of proof can be obtained here. Nevertheless various systems may be useful in some other situations. E.g., as Matthias Baaz pointed out, the ε-calculus is useful in situations where we study the structure of the terms in the proof; the prominent example is Kreisel's Conjecture (see section 4).

2.6. The main theorem of Gentzen [1935] asserts that the sequent calculus *without the cut rule* is still complete. This is a very strong statement, since the cut rule is the only rule where some structure present in the assumptions is missing in the conclusion (if we disregard the terms). Some of the numerous application of the *cut-elimination theorem* and its proof can be found in Chapters I and II of this handbook. Of course we have to pay something for it and the price is high: the increase of the size cannot be bounded by an elementary recursive function, i.e., cannot be bounded by a constant number of iterations of the exponential function. We shall prove such a lower bound in section 5.

2.6.1. There are several theorems which are in a sense equivalent to the cut-elimination theorem: Herbrand's theorem, Hilbert's ε-theorem (see Hilbert and Bernays [1934,1939]), semantic tableaux. Each of them can be used to define a concept of a proof and the resulting measures are closely related. Namely, the known transformations give mutual simulations in time bounded by iterated exponential functions, see 5.1. Thus we have two main classes of proof systems for first order logic: (1) the unrestricted ones, and (2) cut-free (and the equivalent ones).

Considering the undecidability of first order logic, which means that we cannot bound the size of a proof of a formula by *any computable function*, it is quite surprising that the spectrum of natural complexity measures consists essentially of two elements. Can this empirical evidence be supported by a mathematical theorem?

3. Some short formulas and short proofs

In this section we discuss basic concepts used in the study of the length of proofs in first order logic and prove some bounds on the length of proofs. The upper bounds have two applications: firstly they enable us to show big differences in lengths between different types of proofs, the so-called speed-up; secondly, they are needed for reductions of the lower bounds on the length of proofs of one set of formulas to another one.

3.1. We shall use the Hilbert style proof system described in the previous section with the following axioms of equality:

$$x = x$$
$$x = y \to y = x,$$
$$x = y \wedge y = z \to x = z,$$
$$x_1 = y_1 \wedge \ldots \wedge x_n = y_n \to (R(x_1, \ldots, x_n) \to R(y_1, \ldots, y_n))$$

for each predicate symbol R, and

$$x_1 = y_1 \wedge \ldots \wedge x_n = y_n \to F(x_1, \ldots, x_n) = F(y_1, \ldots, y_n)$$

for each function symbol F.

3.2. The first question that we consider is the length of formulas defined by iterating a certain construction (some examples will be considered below). Let $\Phi(R, a_1, \ldots, a_k, b_1, \ldots, b_l)$ be a formula with a specified k-ary predicate symbol and where $a_1, \ldots, a_k, b_1, \ldots, b_l$ are all free variables of Φ. Let us abbreviate the strings of variables by $\bar{a}$ and $\bar{b}$. Now suppose a formula $\varphi_0(\bar{a}, \bar{b})$ is given and we need a sequence of formulas $\varphi_1, \varphi_2, \ldots$ such that

$$\varphi_{n+1}(\bar{a}, \bar{b}) \equiv \Phi(\varphi_n, \bar{a}, \bar{b}) \tag{1}$$

is provable in first order logic. Here $\equiv$ denotes the biconditional.

In order to understand better what is going on, let us write Φ as

$$\Phi(R(\bar{x}_1), \ldots, R(\bar{x}_t), \bar{a}, \bar{b}), \tag{2}$$

where $R(\bar{x}_i)$ denote particular occurrences of R in Φ and $\bar{x}_i$ is a string of k bound, not necessarily distinct, variables of Φ. Thus (1) is better represented by:

$$\varphi_{n+1}(\bar{a}, \bar{b}) \equiv \Phi(\varphi_n(\bar{x}_1, \bar{b}), \ldots, \varphi_n(\bar{x}_t, \bar{b}), \bar{a}, \bar{b}). \tag{3}$$

The variables $\bar{b}$ do not change, they are "parameters", thus we shall omit them from now on.

A trivial solution is to take $\varphi_{n+1}(\bar{a})$ to be *equal* to $\Phi(\varphi_n, \bar{a})$. However often we need φ_n to be of polynomial size and, in fact, we need a polynomial (in n) size proof of (3). If $t > 1$, which is usually the case, then mere substitutions lead to exponentially large formulas. The solution is to replace $\Phi(R, \bar{a})$ by an equivalent formula, in which R occurs only once. This is always possible if $\equiv$ as a connective is present in our language.

3.2.1. Theorem. (Ferrante and Rackoff [1979]) *Suppose $\equiv$ is present in the language. Then, for every formula $\Phi(R, \bar{a})$, there exists an equivalent formula $\Psi(R, \bar{a})$, in which R occurs only once.* □

We shall not prove this theorem here, because we want to construct polynomial size formulas not using biconditional. Several people observed that the assumption about biconditional is essential for Theorem 3.2.1, (of course the negation of $\equiv$ is sufficient too). If we consider, say, all binary connectives without biconditional and its negation, then one can define *positive* and *negative* occurrences of R and it is not possible to replace one by the other. Therefore the theorem fails to hold in this case.

Let us consider the construction of formulas satisfying the inductive condition (1) using Theorem 3.2.1. If we disregard the size of variables, i.e., we assign a unit cost to each variable, we clearly get formulas $\varphi_n(\bar{a})$ of *linear* size by iterating $\varphi_{n+1}(\bar{a}) =_{def} \Phi(\varphi_n(\bar{a}), \bar{a})$.

In order to obtain a polynomial size proof of (1) we prove, for every n,

$$\Psi(\varphi_n, \bar{a}) \equiv \Phi(\varphi_n, \bar{a}). \tag{4}$$

The proof of this formula is obtained from the proof d of $\Psi(R, \bar{a}) \equiv \Phi(R, \bar{a})$ by substituting φ_n for each occurrence of R in d. Hence the proof is also of linear size in n.

In a more precise computation of proof size, we have to take into account the size of variables. After the reduction to one occurrence the inductive condition is

$$\varphi_{n+1}(\bar{a}) = \Psi(\varphi_n(\bar{x}), \bar{a}), \tag{5}$$

where $\bar{x}$ is a string of variables bound in Ψ. Clearly, we cannot use the same string $\bar{x}$ for all n (except in trivial cases) because of the possible clashes. If we use different

strings $\bar{x}$ for each n, we get formulas of size of the order $n \cdot \log n$, since the n-th variable can be coded by a word of length $O(\log n)$. Alternatively we can use just two strings: one for odd n's and one for even n's. The resulting formulas are of linear size but a little unnatural, since one variable occurs in the scope of several quantifiers bounding it. Though unnatural it is usually permitted.

3.2.2. Now we prove the existence of polynomial size formulas defined by iteration in the case when $\equiv$ is not present in the language.

3.2.3. Theorem. (Solovay [unpublished]) *Suppose* $\neg$ *and at least one of the connectives* $\rightarrow$, $\vee$, $\wedge$ *are present in the language. Let* $\varphi_0(\bar{a})$ *and* $\Phi(R, \bar{a})$ *be given. Then it is possible to construct formulas* $\varphi_1(\bar{a}), \varphi_2(\bar{a}), \ldots$ *such that*

$$\varphi_{n+1}(\bar{a}) \equiv \Phi(\varphi_n, \bar{a}) \tag{6}$$

have polynomial size proofs.

Proof. We shall use only the fact that $p \rightarrow q$ has an equivalent formula in the language where q occurs once. We use $p \equiv q$ in (6) and below as an abbreviation of an equivalent formula in the language, e.g. $(p \rightarrow q) \wedge (q \rightarrow p)$, if both $\rightarrow$ and $\wedge$ are present.

The idea of the proof is the same as for the case with $\equiv$ plus an additional trick. The trick is to first define *the graph of the truth value function* for φ_n's. If $f_n(\bar{a})$ is the truth value function, then both $\varphi_n(\bar{a})$ and $\neg\varphi_n(\bar{a})$ can be expressed as positive statements $f_n(\bar{a}) = 1$ and $f_n(\bar{a}) = 0$ respectively.

In order to get simpler formulas we shall use inessential assumptions that a constant 0 is in the language and $\exists x \exists y(x \neq y)$ is a logical axiom. Consider the formula

$$\Phi(R, \bar{a}) \equiv y = 0.$$

Take a prenex normal form of it

$$\overline{Q}\ \Theta(R(\bar{x}_1), \ldots, R(\bar{x}_t), \bar{a}, y),$$

where $\overline{Q}$ denotes the quantifier prefix which bounds, among others, the variables $\bar{x}_1, \ldots, \bar{x}_t$, and all occurrences of R in Θ are displayed. Now we define formulas for the graphs of $f_n(\bar{a})$'s. In order to simplify the formulas, truth will be represented by 0 and falsehood by anything different from 0. Define $\Psi_0(\bar{a}, y)$ to be the formula

$$\varphi_0(\bar{a}) \equiv y = 0 \tag{7}$$

and define $\Psi_{n+1}(\bar{a}, y)$ to be the formula

$$\begin{aligned}\overline{Q}\forall y_1 \cdots \forall y_z(\forall \bar{z} \forall u(((\bar{z} = \bar{x}_1 \wedge u = y_1) \vee \cdots \vee (\bar{z} = \bar{x}_t \wedge u = y_t)) \rightarrow \Psi_n(\bar{z}, u)) \\ \rightarrow \Theta(y_1 = 0, \ldots, y_t = 0, \bar{a}, y)),\end{aligned} \tag{8}$$

where $\overline{Q}$ is as above, $\bar{z} = \bar{x}_i$ is an abbreviation for $z_1 = x_{i1} \wedge \ldots \wedge z_k = x_{ik}$ and $y_i = 0$ are substituted for $R(\bar{x}_i)$ in Θ. Note that the meaning of the antecedent in

the definition is that y_i codes the truth value of $\varphi_n(\bar{x}_i)$; below we give a formal proof of this.

Since Ψ_n occurs only once in the recurrence relation, we get Ψ_n of polynomial size ($O(n \log n)$ if we use different variables and linear if we "recycle" variables).

Define formulas

$$\begin{aligned} \varphi_n(\bar{a}) &=_{df} \Psi_n(\bar{a}, 0), \\ \alpha_{n+1}(\bar{a}, y) &=_{df} \Psi_{n+1}(\bar{a}, y) \equiv (\Phi(\varphi_n, \bar{a}) \equiv y = 0), \\ \beta_n(\bar{a}, y) &=_{df} \Psi_n(\bar{a}, y) \equiv (\varphi_n(\bar{a}) \equiv y = 0). \end{aligned}$$

3.2.4. Lemma. *Let $\bar{x}$ be the string of all free variables of formulas μ and ν; let $, (\mu)$ and $, (\nu)$ be obtained by substituting μ and ν in $, (R)$ for R. Then*

$$\forall \bar{x}(\mu(\bar{x}) \equiv \nu(\bar{x})) \to , (\mu) \equiv , (\nu)$$

has a polynomial size proof in the size of μ, ν and $, .$

The idea of the proof is to use induction on the depth of $, .$ □

3.2.5. Lemma.
(i) $\exists y \Psi_0(\bar{a}, y)$ *is provable.*

The following formulas have polynomial size proofs.
(ii) $\forall y \alpha_{n+1}(\bar{a}, y) \to \exists y \Psi_{n+1}(\bar{a}, y)$;
(iii) $\forall y \alpha_{n+1}(\bar{a}, y) \to \varphi_{n+1}(\bar{a}) \equiv \Phi(\varphi_n, \bar{a})$;
(iv) $\forall y \alpha_{n+1}(\bar{a}, y) \to \forall y \beta_{n+1}(\bar{a}, y)$;
(v) $\forall \ldots \alpha_n(\bar{a}, y) \to \forall \ldots \alpha_{n+1}(\bar{a}, y)$; *where $\forall \ldots$ denotes the universal closure.*

Proof. (i) Use (7): if $\varphi_0(\bar{a})$, then take $y = 0$, if $\neg\varphi(\bar{a})$, take an arbitrary $y \neq 0$.

(ii) Similar as in (i): to find y such that $\Psi_{n+1}(\bar{a}, y)$ holds distinguish the cases $\Phi(\varphi_n, \bar{a})$ and $\neg\Phi(\varphi_n, \bar{a})$. In the first case take $y = 0$, in the second any $y \neq 0$. The formulas involved are of polynomial size, the number of steps is constant, thus the whole proof is polynomial.

(iii) Assume $\forall y \alpha_{n+1}(\bar{a}, y)$; in particular we have $\alpha_{n+1}(\bar{a}, 0)$ which is

$$\Psi_{n+1}(\bar{a}, 0) \equiv (\Phi(\varphi_n, \bar{a}) \equiv 0 = 0).$$

Using the definition of $\varphi_{n+1}(\bar{a})$ this reduces to the statement

$$\varphi_{n+1}(\bar{a}) \equiv \Phi(\varphi_n, \bar{a}).$$

(iv) Assume $\forall y \alpha_{n+1}(\bar{a}, y)$. By (iii) we can substitute $\varphi_{n+1}(\bar{a})$ for $\Phi(\varphi_n, \bar{a})$ in $\forall y\ \alpha_{n+1}(\bar{a}, y)$, which we assume. Thus we get $\forall y\ \beta_n(\bar{a}, y)$. As we do not have the substitution rule, we must use Lemma 3.2.4 to estimate the length of the proof.

(v) Assume $\forall \ldots \alpha_n(\bar{a}, y)$. By the definition of Ψ_0 and (iv) we have also $\forall \ldots \beta_n(\bar{a}, y)$. From definition (8) we immediately get

$$\begin{aligned} \Psi_{n+1}(\bar{a}, y) \equiv\ & \overline{Q} \forall y_1 \ldots \forall y_t (\Psi_n(\bar{x}_1, y_1) \wedge \ldots \wedge \Psi_n(\bar{x}_t, \bar{y}_t) \to \\ & \to \Theta(y_1 = 0, \ldots, y_t = 0, \bar{a}, y)), \end{aligned}$$

using a polynomial size proof. By $\forall \ldots \beta_n(\bar{a}, y)$ we can substitute $\varphi_n(\bar{x}_i)$ for $y_i = 0$.

Thus we get

$$\Psi_{n+1}(\bar{a}, y) \equiv \overline{Q}\forall y_t(\Psi_n(\bar{x}_1, y_1) \wedge \ldots \wedge \Psi_n(\bar{x}_t, y_t) \rightarrow$$

$$\rightarrow \Theta(\varphi_n(\bar{x}_1), \ldots, \varphi_n(\bar{x}_t), \bar{a}, y)).$$

Pushing the universal quantifies inside, we get

$$\Psi_{n+1}(\bar{a}, y) \equiv \overline{Q}(\exists y_1 \Psi_n(\bar{x}_1, y_1) \wedge \ldots \wedge \exists y_t \Psi_n(\bar{x}_t, y_t) \rightarrow$$

$$\rightarrow \Theta(\varphi_n(\bar{x}_1), \ldots, \varphi_n(\bar{x}_t)\bar{a}, y)).$$

Now, by (ii), $\exists y_i \Psi_n(\bar{x}_i, y_i)$ have polynomial size proofs, thus we get

$$\Psi_{n+1}(\bar{a}, y) \equiv \overline{Q}\ \Theta(\varphi_n(\bar{x}_1), \ldots, \varphi_n(\bar{x}_t), \bar{a}, y).$$

By definition of Θ, this is equivalent to

$$\Psi_{n+1}(\bar{a}, y) \equiv (\Phi(\varphi_n, \bar{a}) \equiv y = 0).$$

The calculation that the proofs are of polynomial size use Lemma 3.2.4 and the same ideas as we have already used before. For instance, the last equivalence is obtained by taking the constant size proof of

$$(\Phi(R, \bar{a}) \equiv y = 0) \equiv \overline{Q}\ \Theta(R(\bar{x}_1), \ldots, R(\bar{x}_t)\bar{a}, y)$$

and substituting φ_n for each occurrence of R in the proof.

□

To finish the proof of the theorem we first prove $\forall \ldots \alpha_1(\bar{a}, y)$. The proof is identical with (v) above, except that we get $\forall \ldots \beta_0(\bar{a}, y)$ directly from the defining equation (7). Now we combine the polynomial size proofs of $\forall \ldots \alpha_1(\bar{a}, y) \rightarrow \forall \ldots \alpha_2(\bar{a}, y), \ldots, \forall \ldots \alpha_n(\bar{a}, y) \rightarrow \forall \ldots \alpha_{n+1}(\bar{a}, y)$ to obtain a polynomial size proof of $\forall \ldots \alpha_{n+1}(\bar{a}, y)$. Then, by (iii), we get a polynomial size proof of

$$\varphi_{n+1}(\bar{a}) \equiv \Phi(\varphi_n, \bar{a}).$$

□

3.2.6. Suppose we allow repeated use of the same variables, hence φ_n's are of linear size. Then one can easily check that the sentences in Lemma 3.2.5 have linear size proofs hence (6) has quadratic size proofs.

3.3. We consider two applications of Theorem 3.2.3. The first application is to construct a partial truth definition.

We shall consider T, a sufficiently strong fragment of arithmetic or set theory; namely, we need to be able to formalize syntax in T. A natural assumption is that the theory T is *sequential* which means that T contains Robinson arithmetic **Q** (see Chapter II) and a formula formalizing the relation *"x is the i-th element of y"*; we only require that there exists an empty sequence and each sequence can be prolonged by adding an arbitrary element. E.g. in the Gödel-Bernays set theory **GB** we can define the i-th element of the sequence coded by a class X by

$$(X)_i =_{def} \{x \ ; \ (x,i) \in X\}.$$

Let us stress that it is important to code all elements, it would not suffice to code, say, only sets in **GB**.

Let $\ulcorner \alpha \urcorner$ denote the Gödel number (the code) of a formula α. By a well-known theorem of Tarski [1936], there is no formula $\varphi(x)$ such that

$$T \vdash \varphi(\ulcorner \alpha \urcorner) \equiv \alpha$$

for all sentences α (it is a simple application of the diagonalization lemma). However it is possible to construct such a formula for some classes of sentences α, in particular for α with bounded quantifier complexity. We shall need the following particular case. We would like to define *satisfaction* for formulas $\alpha(\bar{x})$ of bounded size and a string $\bar{x}$ of elements. Let $(x)_i$ denote some coding function in T, i.e., $(x)_i$ is the i-th element of the sequence x (we may assume that every element is a code of some sequence). We want to construct formulas $\varphi_n(x,y), n = 1, 2, \ldots$, such that for every $\alpha(y_1, \ldots, y_n)$ of depth $\leq n$,

$$T \vdash \varphi_n(\ulcorner \alpha \urcorner, x) \equiv \alpha((x)_1, \ldots, (x)_k), \tag{9}$$

using a polynomial size proof, (depth 0 are atomic formulas etc.). In fact we need more: we want to have polynomial size proofs of *Tarski's conditions* for φ_n. Tarski's conditions are conditions which define satisfaction by induction on the depth of formulas. For each connective and each quantifier there is one condition. E.g., for implication Tarski's condition is

$$\varphi_n(\ulcorner \beta \to \gamma \urcorner, x) \equiv (\varphi_n(\ulcorner \beta \urcorner, x) \to \varphi_n(\ulcorner \gamma \urcorner, x)).$$

It is assumed that satisfaction for open formulas is easily definable. This is true in our case, since we assume that T is sufficiently strong. Let $R(x,y)$ be a new binary predicate. Let φ_0 be a formula defining satisfaction for open formulas and let $\Phi(R,x,y)$ be a formula expressing the following:

1. if x is atomic then $\varphi_0(x,y)$,
2. if x is $\neg u$ then $\neg R(u,y)$
3. if x is $u \to v$ then $R(u,y) \to R(v,y)$,

4. if x is $\forall z_i\, u$ and $R(u, y')$ for every sequence y' identical with y on all coordinates $j \neq i$, then $R(x, y)$,

etc. for the other connectives and for quantifiers.

By Theorem 3.2.3 we have polynomial size formulas $\varphi_n(x, y)$ and polynomial size proofs of

$$\varphi_{n+1}(x, y) \equiv \Phi(\varphi_n, x, y). \tag{10}$$

What we need is a little different; namely, we need polynomial size proofs in T of

$$dpt_n(x) \to \varphi_n(x, y) \equiv \Phi(\varphi_n, x, y), \tag{11}$$

where $dpt_n(x)$ is a formula saying that x is a formula of depth $\leq n$. To prove this, it suffices to prove, using polynomial size proofs in T,

$$dpt_n(x) \to \varphi_{n+1}(x, y) \equiv \varphi_n(x, y). \tag{12}$$

To prove (12) we observe that for $n = 0$ it follows from the definition of Φ and for $n > 0$

$$\begin{aligned} &\forall x, y(dpt_n(x) \to \varphi_{n+1}(x, y) \equiv \varphi_n(x, y)) \to \\ &\to \forall x, y(dpt_{n+1}(x) \to \varphi_{n+2}(x, y) \equiv \varphi_{n+1}(x, y)) \end{aligned} \tag{13}$$

have polynomial size proofs. Let us prove the implication. Assume the antecedent and $dpt_{n+1}(x)$. We distinguish the cases: x is atomic, x is a negation, x is an implication etc. If x is atomic, then, by the definition of Φ and (10), both $\varphi_{n+2}(x, y)$ and $\varphi_{n+1}(x, y)$ are equivalent to $\varphi_0(x, y)$. If x is the negation of x' then we have (by the definition of Φ)

$$\varphi_{n+2}(x, y) \equiv \neg\varphi_{n+1}(x', y)$$

and

$$\varphi_{n+1}(x, y) \equiv \neg\varphi_n(x'y).$$

By our assumption we have

$$\varphi_{n+1}(x', y) \equiv \varphi_n(x', y),$$

since we have also $dpt_n(x')$. Thus $\varphi_{n+2}(x, y) \equiv \varphi_{n+1}(x, y)$. The other cases are proved in the same way. In order to see that the resulting proof has polynomial size, observe that it does not use the structure of formulas $\varphi_n, \varphi_{n+1}, \varphi_{n+2}$. Thus these proofs can be constructed from a *finite* proof schema by substituting the formulas $\varphi_n, \varphi_{n+1}, \varphi_{n+2}$. The resulting proof has size linear in the size of $\varphi_n, \varphi_{n+1}, \varphi_{n+2}$. This finishes the proof of (11).

Now we can show easily that (9) have also polynomial size proofs provided α is of depth $\leq n$ (and of polynomial size, i.e., does not use variables with long codes). This is done by induction on the depth of α; we leave out the details.

The following theorem summarizes what we have proved above.

3.3.1. Theorem. (Pudlák [1986]) *Let T be a sequential theory. There exists a sequence of formulas $\varphi_n(x,y)$ (of polynomial size) and such that there are polynomial size proofs in T of Tarski's condition for $\varphi_n(x,y)$ where x is of depth $\leq n$ and polynomial size proofs of*

$$\varphi_n(\lceil\alpha\rceil, x) \equiv \alpha((x)_1, \ldots, (x)_n)$$

for α of depth $\leq n$. □

3.4. Now we consider another application of Theorem 3.2.3. Let T be a fragment of arithmetic, let S be a function symbol for the successor. Now T can be much weaker; we shall specify the condition that we need later. Let $\varphi(x)$ be a formula. We say that $\varphi(x)$ is a *cut* in T of T proves

$$\varphi(0), \tag{14}$$

$$\forall x(\varphi(x) \to \varphi(S(x))), \tag{15}$$

$$\forall x, y(x \leq y \wedge \varphi(y) \to \varphi(x)). \tag{16}$$

If $\varphi(x)$ satisfies only (14) and (15), then $\varphi(x)$ is called *inductive*. Let $\varphi(x)$ be inductive and assume that T proves $x + 0 = 0$, $x + S(y) = S(x + y)$ and the associative law for $+$.

Define $\psi(x)$ by

$$\psi(x) \equiv_{df} \forall z(\varphi(z) \to \varphi(z + x)). \tag{17}$$

Then one can easily show that $\psi(x)$ is also inductive in T and

$$T \vdash \psi(x) \wedge \psi(y) \to \psi(x + y); \tag{18}$$

(this construction is due to Solovay, unpublished). If (18) is satisfied, we say that ψ is closed under addition. Assuming a little bit more about T and that $\varphi(x)$ is a cut, we get that $\psi(x)$ is also a cut. Suppose that T contains exponentiation 2^x along with axioms

$$2^0 = S(0),\ 2^{S(x)} = 2^x + 2^x. \tag{19}$$

Then we can continue by first taking

$$\varphi_1(x) \equiv_{df} \psi(2^x). \tag{20}$$

We get that $\varphi_1(x)$ is inductive (resp., is a cut) and

$$T \vdash \forall x\ \varphi_1(x) \to \varphi(2^x), \tag{21}$$

since $T \vdash \psi(x) \to \varphi(x)$. Then we can repeat the construction, since $\varphi_1(x)$ is inductive, and we obtain some $\varphi_2(x)$ with

$$T \vdash \forall x\ \varphi_2(x) \to \varphi(2^{2^x})$$

etc. Observe that the above construction is *schematic*, we could have assumed that φ is just a second order variables and derive (21) from (14) - (16). More precisely it means the following: let $R(x)$ be a unary predicate, let $Ind(R(x))$ (resp. $Cut(R(x))$ denote the conjunction of (14) and (15) (and (16) resp.). Then there exists a formula $\Psi(R,x)$ and a finite fragment of arithmetic T_0 such that

$$T_0 \vdash Ind(R(x)) \to Ind(\Psi(R,x)) \wedge \forall y(\Psi(R,y) \to R(2^y)).$$

Applying Theorem 3.2.3 to $\Phi(R,x)$ defined by

$$\Phi(R,x) \equiv_{df} \bigwedge T_0 \to \Psi(R,x)$$

we obtain the following theorem.

3.4.1. Theorem. *Let T be a sufficiently strong fragment of arithmetic; suppose $\varphi_0(x)$ is inductive (resp. is a cut) in T. Then there exists a sequence of formulas $\varphi_1(x), \varphi_2(x), \ldots$, such that for each n*

$$Ind(\varphi_{n+1}(x)) \wedge \forall x(\varphi_{n+1}(x) \to (\varphi_n(x) \wedge \varphi_n(2^x)))$$

(resp. the formula with Cut instead of Ind) has a proof in T of size polynomial in n. □

3.5. Since cuts are quite important in the study of theories containing some part of arithmetic, we shall mention a few basic facts about them, though they are not needed in this chapter. More can be found e.g. in Hájek and Pudlák [1993].

In order to obtain a cut closed under multiplication and contained in $\varphi(x)$ one can apply the trick of (17) to $\psi(x)$. It is possible to go on and get cuts closed under more rapidly growing functions, but not for 2^x (unless $\varphi(x)$ has some special properties). There is another way to get such cuts, using which we can better see what these functions are. Let 2^x_n denote n-times iterated exponential function. Let $\omega_n(x)$ be nondecreasing functions such that

$$2^{S(x)}_n = \omega_n(2^x_n); \tag{22}$$

we assume that these properties are provable in T.

Let $\psi_n(x)$ be defined by

$$\psi_n(x) \equiv_{df} \exists y(\varphi_n(y) \wedge x \leq 2^y_n). \tag{23}$$

By construction

$$T \vdash \varphi_n(y) \to \varphi(2^y_n), \tag{24}$$

hence $\psi_n(x)$ is contained in $\varphi(x)$. Also it is easy to check that $\varphi_n(x)$ is a cut in T. To see that $\psi_n(x)$ is closed under $\omega_n(x)$ just observe that

$$T \vdash x \leq 2^y_n \to \omega_n(x) \leq \omega_n(2^y_n) = 2^{S(y)}_n.$$

For instance take $\omega_2(x) = x^2$, then we obtain $\psi_2(x)$ closed under x^2, hence closed under multiplication.

4. More on the structure of proofs

In this section we shall prove two basic results. First we prove that for the usual calculi for predicate logic proofs as sequences can be replaced by tree-proofs with only a polynomial increase. Thus the size measures based on proofs as sequence and proofs as trees are polynomially related. (We shall sketch a different proof of the same result for some propositional proof systems in section 8.) The second result says that the depth of a proof can be bounded by a square root of its size, provided the proved sentence has negligible size. The proof is based on the theory of unification of terms. We shall survey a few other results which use unification, in particular Kreisel's Conjecture on generalizations of proofs in arithmetic.

4.1. Theorem. (Krajíček [1994a]) *Let* $\|\varphi\|^{\text{sequence}}$ *resp.* $\|\varphi\|^{\text{tree}}$ *be the size of the smallest sequence-proof resp. tree-proof of a provable sentence* φ *in the Hilbert style calculus. Then there exists a polynomial* $p(x)$ *such that*

$$\|\varphi\|^{\text{tree}} \leq p(\|\varphi\|^{\text{sequence}})$$

for every provable sentence φ.

We consider here only the Hilbert style calculus, but the result can be extended to the Gentzen sequent calculus, as there are polynomial simulations for both versions - tree and sequence, cf. Eder [1992].

This result is quite surprising, since there is an obvious similarity between sequence-proofs and circuits on the one hand, and tree-proofs and formulas on the other hand. In circuits the output of a gate can be connected with several other gates, thus we can use the boolean function computed at this gate several times. While a formula is represented by a tree, thus each node has at most one successor. Similarly in a sequence-proof we can use a formula several times as a premise of a rule, while in a tree proof it is allowed only once. It is generally accepted, through still a difficult open problem, that circuits are exponentially more powerful for computations of boolean formulas than formulas. Still the corresponding statement for proofs is false as we shall see below.

Proof. We shall first prove the theorem for the propositional calculus. The idea is quite simple. Let $(\varphi_1, \ldots, \varphi_n)$ be a proof. We shall replace this sequence by $\varphi_1, \varphi_1 \wedge \varphi_2, \ldots, (\ldots(\varphi_1 \wedge \varphi_2) \wedge \ldots) \wedge \varphi_n$. In this sequence each formula follows from the previous one. This sequence is, however, not a proof, thus we have to insert some proof trees in it such that a leaf of a tree is $(\cdots(\varphi_1 \wedge \varphi_2) \wedge \cdots) \wedge \varphi_i$ and the root is $(\cdots(\varphi_1 \wedge \varphi_2) \wedge \cdots) \wedge \varphi_{i+1}$.

In order to simplify notation we agree to omit parenthesis in expressions like $(\cdots((\varphi_1 \wedge \varphi_2) \wedge \varphi_3)\cdots) \wedge \varphi_n$. Furthermore let us say that a class of sentences has *polynomial size tree proofs*, abbreviated by *pst-proofs*, if there is a polynomial upper bound on the size of tree proofs in terms of the size of a formula. We shall use this also for proofs from assumptions.

The proof now reduces to the two statements in the following lemma.

4.1.1. Lemma.
(a) $\alpha \to \alpha \wedge \beta$ has a pst-proof provided β is an instance of an axiom;
(b) $\alpha_1 \wedge \cdots \wedge \alpha_n \wedge \gamma$ has a pst-proof from $\alpha_1 \wedge \cdots \wedge \alpha_n$ provided $\alpha_i \to \gamma$ is α_j for some $1 \leq i, j \leq n$.

Proof. (a) The proof of (a) is trivial, but since we shall use the same argument several times below we shall spell it out at least once. Suppose β is an instance of an axiom $\psi(p_1, \ldots, p_k)$, thus β is $\psi(\beta_1, \ldots, \beta_k)$ for some $\beta_1, \ldots, \beta_k$. Consider the following formula $p_{k+1} \to p_{k+1} \wedge \psi(p_1, \ldots, p_k)$. It is a tautology, thus it has a tree-proof d_0. Let arbitrary α and $\beta_1, \ldots, \beta_n$ be given. Then we obtain a tree-proof of $\alpha \to \alpha \wedge \psi(\beta_1, \ldots, \beta_n)$ simply by substituting $\beta_1, \ldots, \beta_k, \alpha$ for $p_1, \ldots, p_{k+1}$ in d_0. Thus the size of this tree-proof will be bounded by $c \cdot (|\alpha| + |\beta_1| + \cdots + |\beta_k|) \leq c \cdot (|\alpha| + |\beta|) = O(|\alpha \to \alpha \wedge \beta|)$, where the constant c is determined by d_0.

(b) To prove the second statement we derive another lemma. (We shall not need it in its full strength.)

4.1.2. Lemma. *Let π be a permutation on $\{1, \ldots, n\}$, $\alpha_1, \ldots, \alpha_n$ formulas. Then*

$$\alpha_1 \wedge \alpha_2 \wedge \cdots \wedge \alpha_n \to \alpha_{\pi(1)} \wedge \alpha_{\pi(2)} \wedge \cdots \wedge \alpha_{\pi(n)} \tag{25}$$

has a pst-proof.

Proof. First we prove that

$$\delta \wedge \beta \wedge \alpha_1 \wedge \alpha_2 \cdots \wedge \alpha_n \wedge \gamma \to \delta \wedge \gamma \wedge \alpha_1 \wedge \alpha_2 \cdots \wedge \alpha_n \wedge \beta \tag{26}$$

has a *pst*-proof. Clearly

$$(\xi \wedge \gamma \to \zeta \wedge \beta) \to (\xi \wedge \eta \wedge \gamma \to \zeta \wedge \eta \wedge \beta) \tag{27}$$

have a *pst*-proofs for any $\beta, \gamma, \xi, \zeta, \eta$. Thus start with a *pst*-proof of

$$\beta \wedge \gamma \to \gamma \wedge \beta,$$

take the *pst*-proofs of

$$(\xi_i \wedge \gamma \to \zeta_i \wedge \beta) \to (\xi_{i+1} \wedge \beta \to \zeta_{i+1} \wedge \gamma)$$

given by (27) for ξ_j equal to $\delta \wedge \beta \wedge \alpha_1 \wedge \cdots \wedge \alpha_j$ and ζ_j equal to $\delta \wedge \gamma \wedge \alpha_1 \wedge \cdots \wedge \alpha_i$, and then apply modus ponens inferences to get (26).

Notice that (26) shows that also

$$\alpha_1 \wedge \cdots \wedge \alpha_{i-1} \wedge \beta \wedge \alpha_{i+1} \wedge \cdots \wedge \alpha_{n-1} \wedge \gamma \to \alpha_1 \wedge \cdots \wedge \alpha_{i-1} \wedge \gamma \wedge \alpha_{i+1} \wedge \cdots \wedge \alpha_{n-1} \wedge \beta \tag{28}$$

has *pst*-proofs. Since

$$(\xi \to \zeta) \to (\xi \wedge \eta \to \zeta \wedge \eta)$$

has a *pst*-proof, we get from (28) a *pst*-proof of (25) for any transposition π. In order to get it for a general π just recall the well-known fact that each π can be decomposed into a polynomial number of transpositions. □

Using the same argument as in (a) one can show that

$$\xi \wedge (\beta \to \gamma) \wedge \beta \to \xi \wedge (\beta \to \gamma) \wedge \beta \wedge \gamma \tag{29}$$

has a *pst*-proof for any ξ, β, γ. Now we can finish the proof of (b). For a given $\alpha_1 \wedge \cdots \wedge \alpha_n$, first move $\alpha_i \to \gamma$ and α_j to the end of the conjunction using Lemma 4.1.2, then apply (29) to add γ at the end, and finally, again using Lemma 4.1.2 move $\alpha_i \to \gamma$ and α_j back.

□

This finishes the proof of the theorem for the case of propositional logic.

4.1.3. Now we sketch how the above argument should be modified in order to get the result for the predicate calculus. We cannot simply take the conjunction of formulas in the proof, since clashes of variables may occur and it is no longer true that $\varphi_1 \wedge \cdots \wedge \varphi_{i+1}$ follows from $\varphi_1 \wedge \cdots \wedge \varphi_i$. Therefore we work with universal closures of the formulas $\varphi_1, \ldots, \varphi_n$. Let $\forall \ldots \varphi$ denote a universal closure of a formula φ. Instead of modus ponens and the two quantifier rules we need now, for some formulas $\psi, \varphi, \alpha, \beta$,

(1) to derive $\forall \ldots \psi$ from $\forall \ldots (\varphi \to \psi)$ and $\forall \ldots \varphi$;

(2) to derive $\forall \ldots (\alpha \to \forall y \beta(y))$ from $\forall \ldots (\alpha \to \beta(x))$, where x does not occur in α;

(3) to derive $\forall \ldots (\exists y \alpha(y) \to \beta)$ from $\forall \ldots (\alpha(x) \to \beta)$, where x does not occur in β.

This can be done as follows. For each particular case of (1)–(3) prove the corresponding implication, i.e.,

$$\begin{array}{c} \forall \ldots \varphi \to (\forall \ldots (\varphi \to \psi) \to \forall \ldots \psi); \\ \forall \ldots (\alpha \to \beta(x)) \to \forall \ldots (\alpha \to \forall y \beta(y)); \\ \forall \ldots (\alpha(x) \to \beta) \to \forall \ldots (\exists y \alpha(y) \to \beta). \end{array} \tag{30}$$

Insert these subproofs in the sequence $\forall \ldots \varphi_1, \forall \ldots \varphi_2, \ldots \forall \ldots \varphi_n$ obtained from the original proof and then use the proof for the propositional calculus. Thus the proof reduces now to following lemma whose proof we omit.

4.1.4. Lemma. *The sentences (30) have pst-proofs.* □

Let us note that the above formulas (30) are essentially the schemas used to formalize first order logic by a finite number of schemas and the single rule modus ponens. Thus what we actually did above was replacing the quantifier rules by quantifier axiom schemas and applying the result for the propositional logic.

4.2. We shall prove the next result, Theorem 4.2.5, using an estimate on unification. It is also possible to prove it directly, but unification is a very useful tool and it is natural to express the combinatorial statement that we need using it.

Consider terms in a language consisting of variables constants and function symbols. A substitution σ is a mapping from the set of variables into the set of terms. We shall write $t\sigma$ for the result of substitution σ applied to t which means that we replace each variable x in t by $\sigma(x)$. A *unification problem* is a set of pairs of terms $\{(t_1, t_2), (t_3, t_4), \ldots, (t_{2k-1}t_{2k})\}$. A substitution σ is a *unifier* if

$$t_1\sigma = t_2\sigma, \ldots, t_{2k-1}\sigma = t_{2k}\sigma.$$

We think of a unification problem as a system of equations with variables being unknown terms; however variables may occur in a solution (=unifier) too. A unifier Σ is a *most general unifier* if for every unifier σ there exists a substitution δ such that $\Sigma\delta = \sigma$. The following result is easy but has very important applications, see Chapter I and Chang and Lee [1973].

4.2.1. Proposition. *If there exists a unifier then there exists a most general unifier.* □

We shall use this proposition later. Now we only need to observe that the most general unifier gives the smallest possible solution.

As usual, we shall think of terms as rooted trees. We say that the root is in depth 0, its sons in depth 1 etc. The *depth of a term* t, denoted by $d(t)$ is the maximal depth that occurs in it; the *size* of t, denoted by $|t|$, is the number of subterms (i.e., the nodes in the tree). We say that a *subterm* s *of a term* t *is in depth* d if the root of s in depth d in t.

The following lemma is the combinatorial substance of the bound on the depth of formulas which we are going to prove.

4.2.2. Lemma. *Let* Σ *be a most general unifier of a unification problem*

$$\{(t_1, t_2), (t_3, t_4), \ldots, (t_{2k-1}, t_{2k})\}.$$

Let $d = \max_i d(t_i)$, $S = \sum_i |t_i\Sigma|$ *and* $D = \max_i d(t_i\Sigma)$. *Then*

$$D \leq \sqrt{(2 + o(1))(d+1)S}.$$

Proof. Let w be a term $t_{i_0}\Sigma$ with the maximal depth; thus $d(w) = D$. Consider a branch B in w of length D. Let $B_1, \ldots, B_r$, $r = \lfloor \frac{D}{d+1} \rfloor$, be the end segments of B of lengths $d+1, 2(d+1), \ldots, r(d+1)$ respectively.

4.2.3. Claim. *For each subterm* u *of any* $t_i\Sigma$ *with* $d(u) > 0$, *there exists* j *such that* u *occurs in* $t_j\Sigma$ *in depth* $< d(t_j)$.

To prove the Claim, suppose it is false. Then u can occur only in the part of the terms $t_j\Sigma$ which belongs to σ. Thus we can obtain a smaller unifier by replacing all occurrences of u by a variable. □

We shall show that $B_1, \ldots, B_r$ have disjoint occurrences. For each B_i take the term w_i corresponding to the first vertex of B_i and take an occurrence of w_i in the depth $\leq d(t_j)$ in some $t_j\Sigma$. Then the occurrences of D_i's in these occurrences of w_i's must be disjoint. Thus we have

$$\begin{aligned} S &\geq |B_1| + \ldots + |B_r| \\ &> d + 1 + 2(d+1) + \ldots + \left\lfloor \frac{D}{d+1} \right\rfloor (d+1) \\ &= (d+1) \cdot \frac{1}{2} \cdot \left\lfloor \frac{D}{d+1} \right\rfloor \cdot \left(\left\lfloor \frac{D}{d+1} \right\rfloor + 1 \right) \\ &\geq \frac{1}{2} \left\lfloor \frac{D}{d+1} \right\rfloor \cdot D \\ &\geq \frac{D^2}{2(d+1)} \Leftrightarrow \frac{D}{2} = \frac{D^2}{2(d+1)} \cdot (1 \Leftrightarrow o(1)). \end{aligned}$$

□

4.2.4. Suppose that $\Delta = (\varphi_1, \ldots, \varphi_n)$ is a proof of φ, i.e., $\varphi_n = \varphi$. The *skeleton* of Δ is a sequence of the same length where each φ_i is replaced by an axiom schemas or a rule used in Δ at this step; moreover, if a rule was used, then there is also information about the proof lines to which the rule was applied. E.g. a formula obtained by modus ponens from formulas φ_j and φ_k will be replaced by (MP, j, k).

We shall show that for a given formula φ and a skeleton Σ there exists in a sense a most general proof. This proof will be constructed from a most general unifier for a unification problem obtained from Σ. In defining the unification problem assigned to the proof Δ we shall follow Baaz and Pudlák [1993], the idea goes back to Parikh [1973].

Replace all atomic formulas in Δ by a single constant c; let $\Delta' = (\varphi'_1, \ldots, \varphi'_n)$ be the resulting sequence. The language for the terms in the unification problem will consists of the constant c, distinct variables v_β for every subformula β of Δ' and a function symbol for each connective and quantifier, i.e., $f_\to, f_\neg, f_\exists$ etc. We shall write the pairs of the unification problem as equations:

(1) For each propositional axiom schema used in the proof we add an equation which represents it; e.g., if φ'_i is $\alpha \to (\beta \to \alpha)$, then we add equation

$$v_{\varphi'_i} = f_\to(v_\alpha, f_\to(v_\beta, v_\alpha));$$

(2) if φ_i is derived from φ_j and φ_k via modus ponens, where φ_k is $\varphi_j \to \varphi_i$, we add

$$v_{\varphi'_k} = f_\to(v_{\varphi'_j}, v_{\varphi'_i});$$

(3) if φ_i is an instance of a quantifier axiom, say φ_i is $\Phi(t) \to \exists x \Phi(x)$, then we add the equation

$$v_{\varphi_i'} = f_{\to}(v_\alpha, f_{\exists}(v_\alpha));$$

here α is the formula obtained from $\Phi(t)$ by substituting c for atomic formulas, this is the same formula which we thus obtain from $\Phi(x)$;

(4) in the same way we add equations for quantifier rules: e.g., suppose φ_j' is $\alpha \to \beta$, φ_i' is $\exists x \alpha \to \beta$ and φ_i is derived from φ_j by the quantifier rule (6.3), then we add equations

$$v_{\varphi_j'} = f_{\to}(v_\alpha, v_\beta),$$

$$v_{\varphi_i'} = f_{\to}(f_{\exists}(v_\alpha), v_\beta);$$

(5) finally we add

$$v_{\varphi_n'} = \tau,$$

where τ is obtained from φ_n' by replacing connectives and quantifiers by the corresponding function symbols $f_{\to}, f_{\neg}, f_{\exists}, \ldots$.

Now we are ready to prove the result. Let $dp(\varphi)$ denote the *depth* of φ a formula, where we consider φ as a term but we treat atomic formulas as atoms. Let $dp(\Delta)$, for a proof Δ, denote the maximal depth of a formula in Δ.

4.2.5. Theorem. (Krajíček [1989a], Pudlák [1987]) *Let Δ be a proof of φ and suppose Δ has smallest possible size. Then*

$$dp(\Delta) = O\left(\sqrt{|\Delta| \cdot (dp(\varphi) + 1)}\right).$$

Proof. Consider a proof Δ of φ of minimum size. Let $\mathcal{U}$ be the unification problem assigned to Δ. Clearly Δ determines a unifier σ for $\mathcal{U}$ in the natural way. Let Σ be a most general unifier of $\mathcal{U}$. We shall construct a proof , $= (\psi_1, \ldots, \psi_n)$ from Σ. Let δ be the substitution such that $\sigma = \Sigma\delta$. Choose a small formula ξ which does not contain any variable which occurs in Δ, e.g., $0 = 0$ if it is in the language. Consider the i-th formula in the proof Δ, i.e., φ_i, and terms $v_{\varphi_i'}\sigma$ and $v_{\varphi_i'}\Sigma$. We have $v_{\varphi_i'}\Sigma\delta = v_{\varphi_i'}\sigma$; this means that $v_{\varphi_i'}\Sigma$ is $v_{\varphi_i'}\sigma$ with some subformulas replaced by variables. Thus we define ψ_i to be φ_i with the subformulas corresponding to variables in $v_{\varphi_i'}\Sigma$ replaced by ξ.

4.2.6. Let us consider an example. Suppose φ_i has been obtained from φ_j by the quantifier rule (6.3). Suppose φ_i is

$$\exists x(P(x) \to (Q(x) \to R(y))) \to R(y),$$

Then $v_{\varphi_i'}\sigma$ is

$$f_{\to}(f_{\exists} f_{\to}(c, f_{\to}(c, c)), c).$$

By case (4) of the definition of $\mathcal{U}$, $v_{\varphi_i}\Sigma$ has form

$$f_{\to}(f_{\exists}(t), s),$$

for some terms t and s. Because Σ is most general, s is either c or a variable and t is either as in $v_{\varphi_i'}\sigma$ or $f_{\to}(c, v_\alpha)$, or a variable. Let us suppose that $v_{\varphi_i}\Sigma$ is

$$f_{\to}(f_{\exists}f_{\to}(c, v_\alpha), c).$$

Then ψ_i is

$$\exists x(P(x) \to \xi) \to R(y).$$

Furthermore ψ_j must be

$$(P(x) \to \xi) \to R(y).$$

We see that the structure of formulas needed in axiom schemas and rules is preserved. Note that also the restrictions on variables in quantifier rules are satisfied, since ξ does not contain any variable which should be bounded. Finally we have also $\psi_n = \varphi_n = \varphi$.

4.2.7. Now we can apply Lemma 4.2.2. The terms in $\mathcal{U}$ have constant depth (where the constant is determined by our choice of the proof system) except for the last equation where we have a term whose depth is equal to $dp(\varphi)$; thus the maximal depth is $O(dp(\varphi))$. Hence the maximal depth of a term $v_{\varphi_i'}\Sigma$ is $O(\sqrt{dp(\varphi)S})$, where $S = \sum_i |v_{\varphi_i'}\Sigma|$. Clearly also $S = O(\sum_i |\psi_i|)$. Furthermore $|, | = O(|\Delta|)$, since we have replaced some subformulas in Δ by a constant size formula ξ in , . This finishes the proof of Theorem 4.2.5. □

4.2.8. Remarks. (1) Clearly the theorem holds for a variety of other systems. In particular it holds for every Frege system, (see section 8 for the definition).

(2) In the proof we have actually constructed "a most general" proof , with the same skeleton Δ. To make it more precise, we should allow propositional variables in our first order formulas and then keep the variables v_α in , and treat them as propositional variables.

4.3. Now we consider the relation of the number of steps to the size and depth of a proof. A relation to the depth is easy to obtain, since the depth does not include information about terms. For instance we can also bound the depth of a most general unifier as follows (see Krajíček and Pudlák [1988]).

4.3.1. Lemma. *Let Σ be a most general unifier of a unification problem $\{(t_1, t_2), \ldots, (t_{2n-1}, t_{2n})\}$. Then*

$$\max_i d(t_i\Sigma) \leq \Sigma_i |t_i|.$$

□

Then using a similar proof as above derive:

4.3.2. Theorem. (Parikh [1973], Farmer [1984], Krajíček [1989a]) *If φ has a proof with n steps, then φ has a proof with n steps and depth bounded above by*

$$O(n + |\varphi|).$$

□

This result gives a bound on the size of a proof in terms of the number of steps, if we disregard terms or use a language without function symbols.

It is more difficult to bound the size of a proof using the number of steps and the size of the formula, if we use the usual definition of the size which includes terms. The technique based on unification works only in cut-free Gentzen sequent calculi. An ordinary proof must be first replaced by a cut-free proof, which results in a big increase. Again we state the result without a proof; see Krajíček and Pudlák [1988] for a more precise bound and a proof (the idea will be also sketched in the proof of Theorem 4.4.1).

4.3.3. Theorem. *There exists a primitive recursive function F such that for every sentence φ and number n, if φ has a proof with n steps, then it has a proof with size bounded by $F(\varphi, n)$.* □

4.3.4. Problem. (Krajíček and Pudlák [1988], Clote and Krajíček [1993]) Can F be elementary, i.e., bounded by a constant time iterated exponential function (in $|\varphi| + n$)?

The following interesting result of S. Buss shows very nicely that it is hard to determine the structure of terms in first order proofs. He proved this theorem for a particular version of a sequent calculus.

4.3.5. Theorem. (Buss [1991b]) *Given a number n and a sequent $, \rightarrow \Delta$, it is not decidable whether $, \rightarrow \Delta$ has a proof with $\leq n$ steps.* □

At first it may seem that this contradicts Theorem 4.3.3, however notice, that Theorem 4.3.3 does not claim that given a proof of φ with n steps, there must exist a proof of φ with size $\leq F(|\varphi|, n)$ *and* n *steps*. Consequently it is not possible to minimize the size and the number of steps at the same time. For some solvable cases see Farmer [1988].

4.4. Finally we mention a related topic which is very popular in this field and also demonstrates that the structure of terms in first order proofs is rather complex. Kreisel stated the following conjecture, see Friedman [1975] and Takeuti [1987]:

> **Kreisel's Conjecture** *Suppose for a formula $\varphi(x)$ and a number k, one can prove $\varphi(S^n(0))$ in Peano Arithmetic using $\leq k$ steps for every n. Then $\forall x \varphi(x)$ is provable in Peano arithmetic.*

Here $S^n(0)$ stands for the term obtained by applying the successor function S n-times to 0. The statement seems to be also quite sensitive on particular formalization of Peano Arithmetic. We shall sketch the idea of a proof for the case where Peano Arithmetic is replaced by a finite fragment of arithmetic. The validity of Kreisel's Conjecture for finite fragments was first proved by Miyatake [1980] using a different proof.

4.4.1. Theorem. *There exists a primitive recursive function G such that for every formula $\varphi(x)$ and numbers k, n, if $\varphi(S^n(0))$ has a proof with k steps and $n > G(\varphi, k)$ then $\forall x \varphi(S^n(x))$ is provable.*

In the theorem we use the provability in pure logic; note that this implies that the theorem is true also for any finitely axiomatized theory T as we can incorporate finitely many axioms in φ. We need to add only a very weak assumption about T in order to deduce Kreisel's conjecture.

4.4.2. Corollary. *Let T be a finite fragment of arithmetic such that*

$$T \vdash \forall x(x = 0 \vee x = S(0) \vee \ldots \vee x = S^{n-1}(0) \vee \exists y(x = S^n(y)))$$

for every n. Then Kreisel's Conjecture holds for T.

Proof-hint. By the assumption on T we have

$$T \vdash \varphi(0) \wedge \varphi(S(0)) \wedge \ldots \wedge \varphi(S^{n-1}(0)) \wedge \forall x \varphi(S_n(x)) \to \forall x \varphi(x),$$

for every formula $\varphi(x)$. □

Proof-idea of Theorem 4.4.1. Let $\varphi(x)$ and k, n be given such that $\varphi(S^n(0))$ has a proof with k steps. We shall see how large n must be.

First we transform the proof into a cut-free proof in the Gentzen system. By Corollary 5.2.2 below, the number of steps in a cut-free proof can be bounded by a constant which depends only k and $|\varphi(x)|$.

Then we apply the technique of unification. This time, however, we consider also the terms in the proof. This is done in two stages. First we consider all proof-skeletons of length K, (there are finitely many). For each of them we find a most general proof (with respect to the propositional and quantifier structure) as in the proof of Theorem 4.2.5. Then for each of these proofs we find most general terms which can be used in them. This can also be done using the theorem about a most general unifier. However, now we treat terms $S^n(0)$ in the sentence $\varphi(S^n(0))$ as unknown, which means that it is represented by a variable in the unification problem. If in terms in the most general solution remain variables for terms, we replace them by first order variables. Thus we obtain a proof whose size is bounded by a primitive recursive function in K and $\varphi(x)$, thus also in k and $\varphi(x)$. Let us denote the bound by L. (This was essentially the idea of the proof of Theorem 4.3.3, except for the treatment of the term $S^n(0)$.)

Let us have a look on what happens with $\varphi(S^n(0))$ in the most general proof. This formula is replaced by $\varphi(t)$ for some term t which has two properties

(1) $|t| \leq L$;

(2) $t\sigma = S^n(0)$, for some substitution σ.

Thus t is either $S^m(y)$ for $m \leq L$ and some variable y, or $S^n(0)$ and $n \leq L$. Hence, if we choose $n > L$, we get a proof of $\varphi(S^m(y))$, with $m < n$. Then, applying generalization, we get a proof of $\forall y \varphi(S^m(y))$ with $m < n$, which in turn implies $\forall x \varphi(S^n(x))$. □

4.4.3. If we now consider full Peano Arithmetic, we can also perform the first part of the proof. But in the second part, where we want to bound the size of terms, the proof fails. It is not possible to write the conditions on terms in the form of a unification problem. Some time ago Baaz proposed a program for proving Kreisel's Conjecture. Among the most important ideas of his are the use of Hilbert's ε-calculus and semiunification (a generalization of unification). This program has been so far realized only for a subtheory of existential induction Baaz and Pudlák [1993]; the proof uses Herbrand's theorem instead of the ε-calculus.

5. Bounds on cut-elimination and Herbrand's theorem

The undecidability of first order logic is caused by the fact that we cannot bound the size of a proof in terms of the size of the proved sentence. Nevertheless it is still possible to deduce something about the proof from the structure of the formula. (Fortunately proof theoretical studies in this direction started before the undecidability was discovered and therefore they were not hindered by this negative fact.) The theorems of this type are Herbrand's theorem, Hilbert's ε-theorem and Gentzen's cut-elimination theorem. The important consequence for all natural systems is that one can bound the quantifier complexity of the proof in terms of the quantifier complexity of the formula. This is achieved on the expense of lengthening the proof, however the lengthening can be bounded by a primitive recursive function. This raises an interesting question which we are going to deal with in this section: determine the growth rate of this function.

These theorems give more information about the structure of proofs. The most important is the cut-elimination theorem, which states that a general proof can always be replaced by a cut-free proof. Cut-free proofs have the so-called *subformula property*, which means that all formulas in the proof are subformulas of the proved formula φ. Here the concept of being a subformula is slightly weaker: the terms in the subformula may be different from those in φ. Hence there are infinitely many subformulas of φ, (even if we do not use function symbols, since there are infinitely many variables).

The three theorems are equivalent in the sense that there are easy proofs of one from another one. More important the simulations are polynomial, or at most exponential (depending on particular proof systems). Hence, if we are satisfied with

a precision up to an exponential function, it is sufficient to give bounds only to one of them.

5.1. Let us consider the important specific case of the relation of the Herbrand theorem and the cut-elimination theorem. An easy extension of the cut-elimination theorem is the *Midsequent Theorem*. It states that each proof of a formula in the prenex form can be transformed into a proof where there is a sequent above which no quantifier rule is used and below which only quantifier rules are used. This can, in fact, be easily constructed from a cut-free proof. An easy analysis of the midsequent shows that it is essentially a Herbrand disjunction (see Hájek and Pudlák [1993,Chapter V]). Recall that a *Herbrand disjunction* is a disjunction of term instances of a *Herbrand variant* of a formula, where the Herbrand variant is obtained by systematically omitting the quantifiers, starting from the outermost, and replacing each universally bounded variable x by $F(y_1, \dots, y_k)$, where F is a new function symbol and $y_1, \dots, y_k$ are the free variables of the current formula. A midsequent does not contain these new function symbols, but the dependencies among the occurrences of variables allow us to replace variables by such terms while preserving the propositional validity of the disjunction.

Now suppose we are given a Herbrand disjunction. First replace the maximal terms whose outermost function symbol is a Herbrand function symbol by distinct variables. Then omit disjunctions and interpret it as a sequent. It has a propositional proof in the sequent calculus. Now each sequent provable in the propositional sequent calculus has a proof of at most exponential size. Thus we get the upper part of the sequential proof. The lower part is obtained by applying quantifier rules in a suitable order. This is possible due to the structure of the Herbrand disjunction. The number of the proof lines with quantifier rules is, of course, bounded by the number of variables. For more details see Takeuti [1987] and Hájek and Pudlák [1993,Chapter V, section 5].

5.2. We shall use the Hilbert style system of Chapter 1. Note however that when no restrictions are posed on the complexity of formulas in the proof the Hilbert style and Gentzen's sequent calculi are equivalent up to a polynomial increase of size. By Theorem 4.1 it is true even if we take proofs in a tree form in one of them and in a sequence form in the other one.

We shall start with an upper bound to cut-elimination.

5.2.1. Theorem. *Suppose a sentence φ has a proof of size n and depth d (i.e., each formula in the proof has logical depth at most d). Then φ has a cut-free proof of size $2^n_{O(d)}$.* □

The proof can be found in Chapter I.

5.2.2. Corollary. *If φ has a proof of size n, then φ has a cut-free proof of size $2^n_{O(|\varphi|\cdot\sqrt{n})}$.*

Proof. This follows from Theorems 5.2.1 and 4.2.5. □

Now we consider a lower bound. The proof will be easy since we have already developed the theory of definable cuts.

5.2.3. Theorem. *There exists a sequence of sentences $\psi_1, \psi_2, \ldots$ such that ψ_n has a proof of size $p(n)$, $n = 1, 2, \ldots$, where p is a fixed polynomial, and there is no cut-free proof of ψ_n with less than 2_n^0 proof-lines for $n = 1, 2 \ldots$.*

Proof. Consider the following very weak fragment of arithmetic. It has the constant 0, the successor function $S(x)$, addition $+$ and exponentiation 2^x. It has axioms of equality, say those considered in section 3, and the following mathematical axioms:

$$0 + x = x$$

$$x + (y + z) = (x + y) + z,$$

$$x + S(x) = S(x + y),$$

$$2^0 = S(0),$$

$$2^{S(x)} = 2^x + 2^x.$$

Furthermore the theory contains a unary predicate symbol $I(x)$ with interpretation "an initial segment of integers without the last element". Thus we also include the axioms saying that I is inductive:

$$\begin{array}{c} I(0) \\ I(x) \to I(S(x)). \end{array} \tag{31}$$

Let us call this theory A. For a natural number n and a term t we denote by $E^n(t)$ the term defined inductively by

$$E^0(t) = t,$$

$$E^{n+1}(t) = 2^{(E^n(t))}.$$

In particular the value of $E^n(0)$ is 2_n^0. Now we define ψ_n by

$$\exists \ldots (\bigwedge A \to I(E^n(0))),$$

where $\bigwedge A$ denotes the conjunction of the axioms of A and $\exists \ldots$ denotes the existential closure.

5.2.4. Claim. *ψ_n's have polynomial size proofs.*

Proof. We shall use Theorem 3.4.1. By this theorem there exists a sequence of formulas $\varphi_0(x), \varphi_1(x), \varphi_2(x), \ldots$ with $\varphi_0(x)$ equal to $I(x)$ and

$$\varphi_{i+1}(0) \wedge \forall x(\varphi_{i+1}(x) \to \varphi_{i+1}(S(x))), \tag{32}$$

$$\forall x(\varphi_{i+1}(x) \to \varphi_i(2^x)), \tag{33}$$

having polynomial size proofs in A for $i = 0, 1, \ldots$. Combining (33) for $i = 0, \ldots, n\Leftrightarrow 1$ we get a polynomial size proof of

$$\forall x(\varphi_n(x) \to I(E^n(x)))$$

in A. The first half of (32) for $i + 1 = n$ together with the last sentence give a polynomial size proof of $I(E^n(0))$ in A, hence a polynomial size proof of ψ_n in first order logic. □

5.2.5. Claim. *Let t be a closed term of A with value in $\mathbb{N}$ equal to m. Let β be a conjunction of term instances of axioms of A such that*

$$\beta \to I(t)$$

is provable in first order logic. Then β contains at least m term instances of the axiom $I(x) \to I(S(x))$.

Proof. W.l.o.g. we may assume that all the terms in β are closed (otherwise substitute 0). Suppose β contains fewer m occurrences of the axiom. Consider the values of terms t such that $I(t) \to I(S(t))$ occurs in β. By the pigeonhole principle there is an $i_0 < m$ which is not the value of any such a term. Assign truth values to the atomic subformulas of $\beta \to I(t)$ as follows: assign an identity a truth value according to its interpretation in natural numbers, and assign $I(t)$ the value TRUE if the value of t is less then or equal to i_0 and FALSE if it is bigger. Thus all the instances of axioms of A get the value TRUE giving this value also to β, while $I(t)$ gets FALSE. Thus $\beta \to I(t)$ cannot be provable. □

Now we derive the lower bound. Let a cut-free proof d of ψ_n be given. Let γ denote $\exists \ldots (\bigwedge A \to I(E^n(0)))$. All the quantifier rules of d are the rules of $\exists$-introduction applied to a term instance of γ or a term instance of a formula obtained from γ in this way. Let d' be the proof obtained by applying the same rules to initial segments of d but omitting the quantifier rules. We have to omit also the contractions applied to formulas with $\exists$, since such formulas will not appear in the new proof d'. Thus the end sequent of d' is a sequent $\Leftrightarrow \gamma_1, \ldots, \gamma_k$ where γ_i's are term instances of γ. Let γ_i be $\alpha_i \to I(E^n(0))$, (where α_i is a term instance of $\bigwedge A$). Then

$$\alpha_1 \wedge \cdots \wedge \alpha_k \to I(E^n(0))$$

is a tautology. By Claim 5.2.5, $k \geq 2_n^0$. Since in the original proof d all $\gamma, \ldots, \gamma_k$ must eventually merge into one formula, d must contain at least $k \geq 2_n^0$ proof-lines. □

Let us note that the above proof can be applied directly to Herbrand theorem too. Namely, the above argument also shows that any Herbrand disjunction for ψ_n must have at least 2^0_n disjuncts.

5.3. The question whether mathematical reasoning as represented by Zermelo-Fraenkel set theory is consistent has intrigued a lot of mathematicians and philosophers. The approach of finitists is to discard it as meaningless and ask instead whether there is a *feasible* proof of contradiction from our axioms of set theory. We shall say more about this modified question in the next section. Now we only want to show that there are theories, not quite unnatural, which are inconsistent but in which no feasible proof of contradiction exists. Such theories have been considered by several researchers including Parikh [1971], Dragalin [1985], Gavrilenko [1984] and Orevkov [1990]; the first and the most influential was the paper of Parikh.

Let T be any fragment of arithmetic (it can be even the set of all true sentences in the standard model). Let t be a closed term whose value m is so large that no proof of size m can be ever constructed. Note that t can be quite simple, say 2^{100}. Extend T to T' by adding axioms

$$I(0),$$
$$I(x) \to I(S(x)),$$
$$\neg I(2^0_t).$$

Clearly T' is not consistent. We shall show, however, that there is no feasible contradiction in T'.

Suppose we can derive a contradiction in T' of size less than n. Then, by the bound on cut-elimination, there is a cut-free proof of contradiction of size less than 2^0_n. This means that we have such a proof of $\Leftrightarrow\!\!\to \neg \bigwedge T_0$, for a finite fragment T_0 of T'. Let T_1 be a Skolemization of T_0. Then the proof of $\Leftrightarrow\!\!\to \neg \bigwedge T_1$ is at most polynomially larger than 2^0_n (since each sentence has a polynomial size proof from its Skolemization). Thus by taking m, hence also t, only slightly larger than n, we get an upper bound 2^0_m to the *open* theory T_1. Then we use the same "interpretation" argument as in the lower bound proof above to show that such a proof cannot exist.

Let us note that we can add also other closure properties such as $I(x) \wedge I(y) \to I(x+y)$ and the same for multiplication, if we take t a little larger, since we can interpret such a theory in T' using small formulas and short proofs (see 3.5).

6. Finite consistency statements – concrete bounds

We have already remarked that there are almost no concrete examples of sentences for which one can prove nontrivial bounds on the length of proofs. There is, however, one exception; namely, the sentences expressing that a theory T does not prove a contradiction using a proof of length $\leq n$; (we shall say that the theory T is consistent up to n).

These are not real mathematical theorems, which would be interesting for an ordinary mathematician, but they are very interesting for people who study

foundations of mathematics. We shall prove bounds on the length of such a statement in the theory T itself. This could be called a *finite* (or, if you prefer the word, *feasible*) version of the second Gödel Theorem. Furthermore, these bounds (especially the lower bounds) have interesting applications.

6.1. Formalization of syntax. We shall derive a strengthening of the second Gödel Incompleteness Theorem and some speed-up results. We shall try to avoid the boring subject of the formalization of syntax as much as possible. However we have to say something about it, since the classical way of formalizing syntax cannot be used here.

6.1.1. First we need a more efficient way of representing numbers by terms. The classical numerals $S^n(0)$ cannot be used, since their length is already greater than n, while we want to bound the lengths of proofs by a polynomial in $|n|$ – the length of the binary representation of n. Thus we define the n-th numeral $\underline{n}$ as follows. If $n = \sum_{i=0}^{k} 2^i a_i,\ a_i \in \{0,1\}$, then $\underline{n}$ is the closed term

$$\underline{a}_0 + \underline{2} \cdot (\underline{a}_1 + 2 \cdot (\underline{a}_2 + \cdots (\underline{a}_{k-1} + 2 \cdot \underline{a}_k) \cdots))),$$

where $\underline{1} = S(0)$, $\underline{2} = \underline{1} + \underline{1}$.

We need also to represent sequences by numbers. A suitable one-to-one mapping from $\{0,1\}^*$ onto $\mathbb{N}$ is given by

$$(a_0, \ldots, a_n) \mapsto \sum 2^i (a_i + 1).$$

A formula φ is first represented as a $0 \Leftrightarrow 1$ sequence a, then we take the number m which codes a as the Gödel number of φ. We shall use the symbol $\ulcorner \varphi \urcorner$ for such a Gödel number of φ.

6.1.2. Suppose that we want to formalize a concept which can be represented as a subset $R \subseteq \mathbb{N}^k$. If R is formalized by a formula $\rho(x_1, \ldots, x_k)$ in a theory of T, then we clearly need that

$$(n_1, \ldots, n_k) \in R \Leftrightarrow T \vdash \rho(\underline{n}_1, \ldots, \underline{n}_k).$$

This alone is usually not sufficient. The key property for our proof is that the above formula has a proof of polynomial length. As it is an important concept, we shall define it precisely.

6.1.3. Definition. Let an axiomatization of a theory T be fixed, let $R \subseteq \mathbb{N}^k$ and let $\rho(x_1, \ldots, x_k)$ be a formula. We say that ρ *polynomially numerates* R in T, if for some polynomial p and every $n_1, \ldots, n_k \in \mathbb{N}$, the following holds: $R(n_1, \ldots, n_k)$ iff $T \vdash \rho(\underline{n}, \ldots, \underline{n}_k)$ by a proof of length $\leq p(|n_1|, \ldots, |n_k|)$.

It turns out that, for a sufficiently strong theory T, the polynomially numerable relations are just the $\mathcal{NP}$ relations.

6.1.4. Theorem. *The following are equivalent*
(1) R is $\mathcal{NP}$;
(2) R is polynomially numerable in Robinson arithmetic $\mathbf{Q}$.

Since (2) $\Rightarrow$ (1) is trivial for any finitely axiomatized theory T, the same theorem holds for any finite consistent extension of $\mathbf{Q}$.

Before we sketch the proof of the converse implication, we state a lemma whose proof we defer to section 6.3.4.

6.1.5. Lemma. *For every bounded formula $\varphi(x)$, with x the only free variable, there exists a polynomial p such that*

$$I\Delta_0 + \mathrm{Exp} \vdash \forall x \varphi(x)$$

implies that for every $n \in \mathbb{N}$,

$$\mathbf{Q} \vdash \varphi(\underline{n})$$

by a proof of length $\leq p(\log n)$.

This lemma allows us to replace $\mathbf{Q}$ by $I\Delta_0 + \mathrm{Exp}$ in the proof of the implication (1) $\Rightarrow$ (2). If we are proving some property of a concept formalized by a Δ_0 formula in $I\Delta_0 + \mathrm{Exp}$, then this statement may not be provable in $\mathbf{Q}$, but each numeric instance has a polynomial proof. Thus for instance we are free to use commutative and associative laws.

Proof-sketch of Theorem 6.1.4. Let an $R \in \mathcal{NP}$ be given. We formalize computations of a Turing machine defining R. Thus $R(n_1, \ldots, n_k)$ is equivalent to the existence of a $0 \Leftrightarrow 1$ string s whose length is bounded by a polynomial in $|n_1|, \ldots, |n_k|$ and which satisfies a certain property (namely, s codes an accepting computation). This property states that each c particular bits of s have one of some particular forms, where c is some constant. For a given s, there are polynomially many such conditions. Denote by $\sigma(x_1, \ldots, x_k, y)$ such a formula, where $x_1, \ldots, x_n$ stand for $n_1, \ldots, n_k$ and y for the string s. If $R(n_1, \ldots, n_k)$ is true, then $\sigma(\underline{n}_1, \ldots, \underline{n}_k, \underline{m})$ holds for some number m, whose length is bounded by a polynomial in $|n_1|, \ldots, |n_k|$. To prove $\sigma(\underline{n}_1, \ldots \underline{n}_k, \underline{m})$ by a polynomial proof in $\mathbf{Q}$, transform it into statements about single bits of the string encoded by m. Since the string really witnesses $R(n_1, \ldots, n_k)$, these elementary statements are true, hence provable. Finally derive $\exists y \sigma(\underline{n}_1, \ldots, \underline{n}_k, y)$ from $\sigma(\underline{n}_1, \ldots, \underline{n}_k, \underline{m})$. Thus $\exists y \sigma(x_1, \ldots, x_n)$ polynomially numerates R. $\square$

Now we apply Theorem 6.1.4 to the provability predicate. Suppose a theory T is given by an $\mathcal{NP}$, resp. $\mathcal{P}$, set of axioms. Let $R(x, y)$ denote that x is a proof of y in T. Then R is in $\mathcal{NP}$, resp. $\mathcal{P}$, also. By Theorem 6.1.4 there is a formalization Proof_T of this relation, such that every true numeric instance has a polynomial proof in $\mathbf{Q}$. Since the relation "$|m| < n$" can also be polynomially numerated, we get the following corollary:

6.1.6. Corollary. *There exists formalization* $\mathrm{Pr}_T(\underline{n}, \ulcorner\varphi\urcorner)$ *of the relation* $\|\varphi\|_T \leq n$ *such that whenever it is true that* $\|\varphi\|_T \leq n$, *then* $\mathrm{Pr}_T(\underline{n}, \ulcorner\varphi\urcorner)$ *has a proof of polynomial length in* n *in* $\mathbf{Q}$. □

Recall that $\|\varphi\| \leq n$ is a convenient notation for the statement that there exists a proof d of φ in T whose length is $\leq n$. Furthermore we shall denote by

$$\mathrm{Con}_T(x) \equiv_{df} \neg \mathrm{Pr}_T(x, \ulcorner\underline{0} = \underline{1}\urcorner),$$

the consistency of T up to the length x.

6.2. Now we are ready to prove the main lemma of the lower bound.

6.2.1. Lemma. *Let* T *be a sufficiently strong fragment of arithmetic. Let* $f : \mathbb{N} \to \mathbb{N}$ *be a polynomial time computable, increasing function. Suppose that*

for every n *and every sentence* φ *of length* $\leq \log n$, *if* $\|\varphi\|_T \leq n$ *then* $\|\mathrm{Pr}_T(n, \varphi)\|_T \leq f(n)$;

and moreover the formalization of this sentence is provable in T. *Then*

$$n = O\left(f\left(\|\mathrm{Con}_T(\underline{n})\|_T\right)\right)$$

Hence if f can be extended to an increasing function defined on positive real numbers, then we can write the conclusion as

$$\|\mathrm{Con}_T(\underline{n})\|_T = \Omega(f^{-1}(n)).$$

Proof. We shall denote by $\ulcorner\varphi(\dot{x})\urcorner$ a formalization of the function $n \mapsto$ "the Gödel number of $\varphi(\underline{n})$". (E.g. the statement *"for all* n *the formula* $\varphi(\underline{n})$ *has property* P*"* is formalized by $\forall x\ P(\ulcorner\varphi(\dot{x})\urcorner)$.) Strictly speaking, in most cases this function cannot be formalized by a term and one has to use a formula with two variables which defines the graph of this function. This would however make the notation awkward.

Using the diagonalization lemma (see Chapter II) define a formula δ such that

$$T \vdash \delta(x) \equiv \neg \mathrm{Pr}(x, \ulcorner\delta(\dot{x})\urcorner). \tag{34}$$

6.2.2. Claim. $\|\delta(\underline{n})\|_T \leq n \to \|\underline{0} = \underline{1}\|_T \leq g(n)$, *where* $g(n) = O(f(n))$.

To prove the claim assume

$$\|\delta(\underline{n})\|_T \leq n. \tag{35}$$

Substituting $\underline{n}$ in (34) we get

$$\|\delta(\underline{n}) \equiv \neg Pr(\underline{n}, \ulcorner\delta(\underline{n})\urcorner)\|_T = (\log n)^{O(1)}. \tag{36}$$

Here we implicitly use the assumption that $\ulcorner\varphi(\dot{x})\urcorner$ has the polynomial numerability property. The assumption of the lemma and $\|\delta(\underline{n})\|_T \leq n$ implies that

$$\|\mathrm{Pr}(\underline{n}, \ulcorner\delta(\underline{n})\urcorner)\|_T \leq f(n). \tag{37}$$

Thus we get from (35), (36) and (37)

$$\|\underline{0} = \underline{1}\|_T = O(n + (\log n)^{O(1)} + f(n)) = O(f(n)),$$

which proves the claim. □

Since the assumption of the lemma is formalized in T, the claim can also be proved in T, thus we get

$$T \vdash \mathrm{Pr}_T(x, \ulcorner \delta(\dot{x}) \urcorner) \to \neg\, \mathrm{Con}_T(\ulcorner g(x) \urcorner). \tag{38}$$

Since T is consistent, the claim implies in particular that $\|\delta(\underline{n})\|_T > n$. Thus it suffices to upper-bound $\|\delta(\underline{n})\|_T$ using $\|\,\mathrm{Con}_T(\underline{n})\|_T$. (Observe that the proof follows very much the structure of the proof of the second Gödel Incompleteness Theorem.) First substitute $\underline{n}$ in (38), thus we obtain

$$\|\,\mathrm{Con}_T(\ulcorner g(\underline{n}) \urcorner) \to \neg\, \mathrm{Pr}(\underline{n}, \ulcorner \delta(\underline{n}) \urcorner)\|_T = (\log n)^{O(1)}.$$

Combining it with (36) we get

$$\|\,\mathrm{Con}_T(\ulcorner g(\underline{n}) \urcorner) \to \delta(\underline{n})\|_T = (\log n)^{O(1)},$$

hence

$$\|\,\mathrm{Con}_T(\ulcorner g(\underline{n}) \urcorner)\|_T \geq \|\delta(\underline{n})\|_T \Leftrightarrow (\log n)^{O(1)} \geq n \Leftrightarrow (\log n)^{O(1)}.$$

Since $g(n) = O(f(n))$ and $\|\ulcorner g(\underline{n}) \urcorner = \underline{g(n)}\|_T = (\log n)^{O(1)}$, the conclusion of the lemma follows. □

6.2.3. Theorem. (Friedman [1979], Pudlák [1986]) *Let T be a sufficiently strong fragment of arithmetic axiomatized by an $\mathcal{NP}$ set of axioms. Then there exists $\varepsilon > 0$ such that for all n,*

$$\|\,\mathrm{Con}_T(\underline{n})\| > n^{\varepsilon}.$$

Proof. by Corollary 6.1.6 and Lemma 6.2.1. □

With a little more additional work one can reduce the assumption about the strength to the condition $T \supseteq \mathbf{Q}$. Also it is possible to give a more precise lower bound by improving the bound in Corollary 6.1.6. The best lower bound has been proved in Pudlák [1987]. In that paper we considered first order logic augmented with Rosser's C-rule, which allows to introduce names for objects whose existence has been proved. Formally it means that we can derive $\varphi(c)$ from $\exists x \varphi(x)$ for a new constant c. (This *apparently* enables to shorten some proofs, but we are not able *to prove* a speed-up of this calculus versus the ordinary one.) For such a calculus we obtained a lower bound $\Omega(n/(\log n)^2)$.

6.3. Now we turn to the upper bound. Recall that in section 3 we proved that for a sequential theory T, there exists a sequence of formulas φ_n which define satisfaction for formulas of depth $n = 1, 2, \ldots$. Moreover

$$\varphi_n(\ulcorner \alpha \urcorner, x) \equiv \alpha((x)_1, \ldots, (x)_n) \tag{39}$$

and Tarski's conditions have polynomial size proofs. The following is an immediate consequence.

6.3.1. Lemma. *(1) For every axiom α of T, $dp(\alpha) \leq n$, T proves $\forall x \varphi_n(\lceil \alpha \rceil, x)$ using a polynomially long proof.*

(2) For every n, T proves that any axiom of depth $\leq n$ is true and the truth of formulas of depth $\leq n$ is preserved by every rule, furthermore these proof are bounded by a polynomial in n.

Proof. The first part follows directly from (39). For part (2), let us consider only modus ponens. Thus we need a proof of

$$\forall x, y \, (\lceil dp(x \to y) \leq \underline{n} \rceil \wedge \forall z \varphi_n(x, z) \wedge \forall z \varphi_n(x \to y, z) \to \forall z \varphi_n(y, z)) \,. \tag{40}$$

We know that Tarski's condition

$$\lceil dp(x \to y) \leq \underline{n} \rceil \to (\varphi_n(x \to y, z) \equiv (\varphi_n(x, z) \to \varphi_n(y, z)))$$

has a polynomial proof, thus also (40) has a polynomial proof. □

6.3.2. Theorem. (Pudlák [1986]) *Let T be a sequential theory axiomatized by a finite set of axioms. Then*

$$\mathrm{Con}_T(\underline{n}) = n^{O(1)}.$$

Proof. Let n be given. Let $\alpha(x)$ be the following formula

$$\forall y, z (\text{“}y \text{ is a proof of depth } \leq \underline{n} \text{ and size } \leq x\text{”} \wedge \\ \wedge \text{“}z \text{ is a formula of } y\text{”} \to \forall v \varphi_n(y, v)).$$

Lemma 6.3.1 implies that $\alpha(\underline{0})$ and $\forall x(\alpha(x) \to \alpha(S(x))$ have polynomial size proofs. Thus by proving $\alpha(\underline{0}), \alpha(\underline{1}), \ldots, \alpha(\underline{n})$ one by one we get a polynomial proof of $\alpha(\underline{n})$. On the other hand, by (39), we have $\forall v \neg \varphi_n(\lceil \underline{0} = \underline{1} \rceil, v)$, also by a polynomial proof. Thus we have a polynomially long proof that a proof of length $\leq n$ does not contain the formula $\underline{0} = \underline{1}$. □

This theorem has been proved also for some theories which are not finitely axiomatized, namely for theories axiomatized by a certain kind of axiom schemas. These results include the theories Peano Arithmetic and Zermelo-Fraenkel set theory.

Furthermore for finitely axiomatized sequential theories it is possible to improve the bound to $O(n)$. This improvement is based on the following ideas.

Firstly, by counting more precisely, it is possible to prove that (39) and Tarski's conditions for the truth definition for formulas of depth d have proofs of size $O(d^2)$.

Secondly, by Theorem 4.2.5, a proof of contradiction of length n can be transformed into a proof of depth $O(\sqrt{n})$. Thus we need the truth definition only for such a depth and hence the auxiliary formulas have linear size proofs.

Finally, one can use a shorter way to prove $\alpha(\underline{n})$. This is because of the following lemma, which gives us a proof even much shorter than $O(n)$.

6.3.3. Lemma. *Suppose $T \supseteq \mathbf{Q}$ and $\alpha(x)$ is a formula such that T proves*

$$\alpha(0) \wedge \forall x\,(\alpha(x) \to \alpha(S(x)))\,.$$

Then

$$\|\alpha(\underline{n})\|_T = O\left((\log n)^2\right)$$

(thus $\alpha(\underline{n})$ have proofs polynomial in $\log n$).

Proof. First define a subcut α' of α by

$$\alpha'(x) \equiv_{df} \forall y \leq x \alpha(y).$$

Then take a subcut α'' of α which is closed under addition and multiplication. This is easy, if the integers in T satisfy the laws of a ring; in 3.5 we have sketched a possible definition of such an α''. This is more technical for $\mathbf{Q}$ alone, so we refer the reader to Nelson [1986]. Then, in order to prove $\alpha''(\underline{n})$, prove $\alpha''(t)$ inductively for the subterms of $\underline{n}$. There are $O(\log n)$ such subterms and they have length $O(\log n)$. Finally use the fact that $T \vdash \alpha''(x) \to \alpha(x)$. □

6.3.4. Proof-sketch of Lemma 6.1.5. We shall use a similar idea in the proof that we still owe to the reader.

First we consider the length of proof of $\varphi(\underline{n})$ in $I\Delta_0$. An easy model-theoretical argument shows that the assumption of the lemma implies

$$I\Delta_0 \vdash \forall x(\exists y = 2_k^x \to \varphi(x))$$

for some $k \in \mathbb{N}$. Take a cut $\psi(x)$ in $I\Delta_0$ such that

$$I\Delta_0 \vdash \forall x(\psi(x) \to \exists y = 2_k^x).$$

Then we have

$$I\Delta_0 \vdash \forall x(\psi(x) \to \varphi(x)).$$

Thus we only need to construct a short proof of $\psi(\underline{n})$ in $I\Delta_0$, which is done as in the lemma above.

To get the theorem for $\mathbf{Q}$, use the well-known fact that $I\Delta_0$ has an interpretation in $\mathbf{Q}$ (this is an unpublished result of Wilkie, for a proof see Nelson [1986]). □

6.4. Some applications of the bounds. The lower bound can be used to show some strengthenings of the second Gödel Incompleteness Theorem. While the original theorem says only that T is consistent with its formal inconsistency (cf. Chapter II), we shall show that T is consistent with a statement saying that there is a *short* proof of contradiction. We have two such results.

6.4.1. Corollary. (Pudlák [1985]) *Let $T \supseteq \mathbf{Q}$ be consistent and axiomatized by an $\mathcal{NP}$ set of axioms. Then the following hold:*

(1) if I is a cut in T, then

$$T + \exists x(I(x)\&\neg \operatorname{Con}_T(x))$$

is consistent,

(2) if $\beta(x)$ is a bounded arithmetical formula such that $\beta(\underline{n})$ is true for every $n \in \mathbb{N}$, then there exists $k \in \mathbb{N}$ such that

$$T + \exists x(\beta(x)\&\neg \operatorname{Con}(x^k)).$$

is consistent, (x^k is $\underbrace{x \cdot x \cdot \ldots \cdot x}_{k-\text{times}}$).

Proof. (1) Suppose that the statement is false. Then

$$T \vdash I(x) \to \operatorname{Con}_T(x).$$

But, by Lemma 6.3.3, we have that $\|I(\underline{n})\|_T = O((\log n)^2)$, whence also $\|\operatorname{Con}_T(\underline{n})\|_T = O((\log n)^2)$ which is a contradiction with $\|\operatorname{Con}_T(\underline{n})\| \geq n^\varepsilon$, $\varepsilon > 0$.

(2) We need the following lemma.

6.4.2. Lemma.

$$s(\underline{n}) = \underline{s(n)},\ \underline{m} + \underline{n} = \underline{m+n},\ \underline{m} \cdot \underline{n} = \underline{m \cdot n}$$

have proofs of size polynomial in $\log n$ and $\log m$.

The proof is easy, if we assume ring operations, otherwise we have to work in a suitable cut. □

One consequence is that the same holds for arbitrary arithmetical terms. This can be used to show that, for a bounded formula $\beta(x)$ in the language of $\mathbf{Q}$, there exists a polynomial p_1 such that

$$\|\beta(\underline{n})\|_T \leq p_1(n), \tag{41}$$

whenever $\beta(\underline{n})$ is true.

The second consequence is that there is a polynomial p_2 such that

$$\|\underline{n}^k = \underline{n^k}\|_T \leq p_2(k, \log n). \tag{42}$$

(Hint: prove $\underline{n} \cdot \underline{n} = \underline{n^2}, \underline{n} \cdot \underline{n^2} = \underline{n^3}, \ldots, \underline{n} \cdot \underline{n^{k-1}} = \underline{n^k}$.)

We continue with the proof of (2). Assume that $\beta(\underline{n})$ is true for every $n \in \mathbb{N}$. Then we have (41) for all n. Clearly

$$\|\operatorname{Con}_T(\underline{n^k})\|_T \leq O(\|\beta(\underline{n})\|_T + \|\forall x(\beta(x) \to \operatorname{Con}_T(x^k)\|_T + \|\underline{n}^k = \underline{n^k}\|_T),$$

thus for some polynomial p_3

$$\| \operatorname{Con}_T(\underline{n^k})\|_T \leq p_3(n, k, \|\forall x(\beta(x) \to \operatorname{Con}_T(x^k)\|_T). \tag{43}$$

By Theorem 6.2.3 there exists an $\varepsilon > 0$ be such that for every r

$$\| \operatorname{Con}_T(\underline{r})\|_T \geq r^{\varepsilon}. \tag{44}$$

Let d be the degree of n in $p_3(n, k, m)$. Take k so that $k\varepsilon > d$. Now suppose (2) fails for k, thus

$$T \vdash \forall x(\beta(x) \to \operatorname{Con}_T(x^k)).$$

Let m be the length of this proof. Take n so large that

$$p_3(n, k, m) < n^{k\varepsilon}.$$

Then, by (43),

$$\| \operatorname{Con}(\underline{n^k})\|_T \leq p_3(n, k, m) < n^{k\varepsilon},$$

which is a contradiction with (44). □

6.5. Let us observe that also the upper bound on $\| \operatorname{Con}_T(\underline{n})\|_T$ can be used to obtain interesting corollaries. This is because the upper and the lower bounds are quite close, especially in the case of the calculus with the C-rule, hence the results that we used in the proof of the upper bound cannot be substantially improved. There are two such results. One is Theorem 3.2.3, where more precise calculations give a bound $O(n^2)$. The second one is the bound $O(\sqrt{n})$ on the depth of the shortest proof of a fixed size formula. Due to our bounds, the first result cannot be proved for a function which is $o(n^2/(\log n)^2)$, while the second for $o(\sqrt{n}/\log n)$. (The first statement is true also for first order logic without the C-rule, see Pudlák [1987].) However we feel that it should be possible to find direct arguments showing even the sharp bounds $\Omega(n^2)$ and $\Omega(\sqrt{n})$.

Further applications of the lower bounds will be shown in the next section.

7. Speed-up theorems in first order logic

The *speed-up* phenomenon is a situation, where we have two systems (proof systems, theories) such that some theorems have much shorter proofs in one of them. After the problem of proving lower bounds on proofs of concrete statements, this is the second most interesting problem. Note that in the intuitive relation between complexity of computations and complexity of proofs, speed-up theorems should correspond to separations of complexity classes.

We have already encountered a speed-up theorem in section 5, where we showed that cut-free proofs can be much longer than the proofs with cuts or proofs in a Hilbert style calculus. We shall consider such questions about propositional logic in sections 8 and 9. In this section we shall talk about two most important speed-up

phenomena in first order logic. The one is the speed-up caused by having a stronger theory. The second one appears when we prove $\mathrm{Pr}_T(\varphi)$ in T instead of φ itself. It turns out that a dramatic speed-up is obtained in the first case almost by any extension of the theory. The second question, as we shall see, is related to the first.

From the point of view of the principal goal of proving concrete lower bounds these results are rather disturbing. Suppose that with a lot of effort we succeeded in proving that a statement, such as $\mathcal{P} \neq \mathcal{NP}$, does not have a feasible proof, say, in **ZF**, (which is extremely unlikely to happen in the near future) which would explain why we are not able to prove it. Then a simple and intuitively correct additional axiom, say $\mathrm{Con}_{\mathbf{ZF}}$, could change the situation completely, because in $\mathbf{ZF} + \mathrm{Con}_{\mathbf{ZF}}$ some proofs can be much shorter and we still would not know, why we cannot decide $\mathcal{P} = \mathcal{NP}$ in such a theory.

This suggests that a more reasonable goal is not to look for proofs of lower bounds in as strong as possible theories, but rather to consider also weak theories and to try to find out which ones are adequate for which problems.

Finally we shall consider an interesting situation where we can get a large speed-up by a *conservative* extension of a theory. A typical case is extending **ZF** (Zermelo-Fraenkel set theory with the axiom of choice) to **GB** (Gödel-Bernays set theory). Such extensions are called *predicative*.

7.1. We shall start with a very strong result of Ehrenfeucht and Mycielski which has a very simple proof.

7.1.1. Theorem. (Ehrenfeucht and Mycielski [1971]) *If the theory $T + \neg\alpha$ is undecidable, then there is no recursive function f such that*

$$\|\varphi\|_T \leq f(\|\varphi\|_{T+\alpha}),$$

for every sentence φ provable in T.

Proof. Suppose there is such an f. Suppose $T + \neg\alpha \vdash \varphi$, hence $T \vdash \alpha \vee \varphi$. Then, ignoring an additive constant, we have

$$\|\alpha \vee \varphi\|_T \leq f(\|\alpha \vee \varphi\|_{T+\alpha}) \leq f(|\alpha| + |\alpha \vee \varphi|).$$

(To prove $\alpha \vee \varphi$ from axioms $T + \alpha$, we first derive α, then $\alpha \vee \varphi$). Thus we can decide whether $T + \neg\alpha$ proves φ by checking all proofs of length $\leq f(|\alpha| + |\alpha \vee \varphi|)$ – a contradiction. □

7.1.2. Corollary. *Let T be a recursively axiomatized theory containing Robinson Arithmetic* **Q**. *Then* any *proper extension of T has arbitrary recursive speed-up over T.*

Proof. This follows from the fact that **Q** is essentially undecidable. □

The following result of Statman, improved by Buss, concerns the number of steps. We state it without a proof.

7.1.3. Theorem. (Statman [1981], Buss [1994]) *Let T be a theory axiomatized by a finite number of axiom schemas. Suppose α is a sentence undecided by T and such that $T + \neg\alpha$ is consistent. Then $T + \alpha$ has an infinite speed-up with respect to the number of steps over T, i.e., there exists an infinite set Φ of sentences and a k such that*

$$\forall \varphi \in \Phi \quad T + \alpha \vdash^{k}_{\text{steps}} \varphi,$$

but there is no m such that

$$\forall \varphi \in \Phi \quad T \vdash^{m}_{\text{steps}} \varphi,$$

□

We shall give some intuition about this theorem by an example due to Baaz. Suppose Kreisel's Conjecture holds for T. Let $\varphi(x)$ be a formula such that

1. $T \nvdash \forall x \varphi(x)$;
2. $\forall n \quad T \vdash \varphi(\underline{n})$.

A typical example of such a formula is $\mathrm{Con}_T(x)$. Then, clearly, for some constant c

$$T + \forall x \varphi(x) \vdash^{c}_{\text{steps}} \varphi(\underline{n}),$$

for every n, since we only need to substitute the term $\underline{n}$ into $\varphi(x)$. If, however, for some m

$$T \vdash^{m}_{\text{steps}} \varphi(\underline{n}),$$

for every n, we would get $T \vdash \forall x \varphi(x)$ by Kreisel's Conjecture. Thus $T + \forall x \varphi(x)$ has infinite speed-up over T.

7.2. Next we shall show that the lower bound on the length of proof of $\mathrm{Con}_T(\underline{n})$ in T (Theorem 6.2.3) can be used to obtain speed-up when T is extended to $T + \mathrm{Con}_T$ or when $Pr(\varphi)$ is used instead of φ.

Let T be a sufficiently strong fragment of arithmetic. We want to use sentences $\mathrm{Con}_T(f(\underline{n}))$ for fast-growing functions f. Such functions needn't be representable by terms in T. So we take the sentence

$$\exists y \, (\varphi(\underline{n}, y) \vee \mathrm{Con}_T(\varphi)) \tag{45}$$

instead, where φ defines the graph of f. Then we need two conditions to be satisfied

1. f is a *provably total* function in T, i.e., $T \vdash \forall x \exists! y \varphi(x, y)$;
2. φ polynomially numerates the graph of f.

Let us make a simple observation.

7.2.1. Lemma. *For every recursive function f, there exists a recursive function g such that*

1. $\forall n \in \mathbb{N} \qquad f(n) \leq g(n)$,

2. the graph of g is a polynomial time computable relation.

Proof. Let M be a Turing machine for f. Then one can construct a Turing machine M' which on input n prints 2^m in binary, where m is the number of steps of M on n. We take g to be the function computed by M'. □

If T is sufficiently strong, Lemma 7.2.1 can be formalized in T. Thus polynomial numerability is not an essential restriction. We shall abbreviate (45) by $\mathrm{Con}_T(f(\underline{n}))$.

7.2.2. Theorem. *Let T be a sufficiently strong theory. Let f be a provably total increasing recursive function in T whose graph has a polynomial numeration in T. Then there exists a $\delta > 0$ such that*

$$\| \mathrm{Con}_T(f(\underline{n}))\|_T \geq f(n)^\delta,$$

while

1. $\| \mathrm{Con}_T(f(\underline{n}))\|_{T+\mathrm{Con}_T} = O(\log n)$;
2. $\| \mathrm{Pr}_T(\ulcorner \mathrm{Con}_T(f(\underline{n}))\urcorner)\|_T = O(\log n)$.

Thus in both cases we get a speed-up by any provably total recursive function of T.

Proof. *Lower bound.* Let $\varphi(x, y)$ be the formula which polynomially numerates $f(x) = y$. Thus we want to bound

$$\|\exists y(\varphi(\underline{n}, y) \wedge \mathrm{Con}_T(y))\|_T.$$

Let $m = f(n)$. Clearly

$$\exists! y\varphi(\underline{n}, y) \wedge \varphi(\underline{n}, \underline{m}) \to \mathrm{Con}_T(\underline{m}). \tag{46}$$

Thus

$$\| \mathrm{Con}_T(\underline{m})\|_T \leq$$
$$\|\exists y(\varphi(\underline{n}, y) \wedge \mathrm{Con}_T(y))\|_T + \|\exists! y\varphi(\underline{n}, y)\|_T + \|\varphi(\underline{n}, \underline{m})\|_T + K,$$

where K is the length of the proof of (46). The proof of (46) depends only linearly on the lengths of n and m, thus $K = O(\log m)$. Similarly

$$\|\exists! y\varphi(\underline{n}, y)\|_T = O(\log n),$$

since we assume $T \vdash \forall x \exists! y \varphi(x, y)$. Finally we have a bound $(\log m)^{O(1)}$ on $\|\varphi(\underline{n}, \underline{m})\|_T$ by polynomial numerability. Thus, using Theorem 6.2.3 we have

$$m^\varepsilon \leq \| \mathrm{Con}_T(\underline{m})\|_T \leq \|\exists y(\varphi(\underline{n}, y) \wedge \mathrm{Con}_T(y))\|_T + (\log m)^{O(1)},$$

which gives the lower bound.

Upper bound (1). Recall that Con_T denotes $\forall x\, \mathrm{Con}_T(x)$ and that we assume $T \vdash \forall x \exists! y \varphi(x, y)$. Again, the proof of

$$\forall x\, \mathrm{Con}_T(x) \wedge \forall x \exists! y \varphi(x, y) \to \exists y(\varphi(\underline{n}, y) \wedge \mathrm{Con}_T(y))$$

depends only linearly on the length of n, thus

$$\|\exists y(\varphi(\underline{n}, y) \wedge \mathrm{Con}_T(y)\|_{T+\mathrm{Con}_T} = O(\log n).$$

Upper bound (2). We want to bound

$$\|\Pr_T(\ulcorner \exists y(\varphi(\underline{n}, y) \wedge \mathrm{Con}_T(y))\urcorner)\|_T \tag{47}$$

The argument will be similar to the one above. We have

$$\forall y \Pr_T(\ulcorner \mathrm{Con}_T(\dot{y})\urcorner) \wedge \Pr_T(\ulcorner \forall x \exists y \varphi(x, y)\urcorner) \to \Pr_T(\ulcorner \exists y(\varphi(\underline{n}, y) \wedge \mathrm{Con}_T(y))\urcorner),$$

by a proof of linear size in $\log n$. So it remains only to show that the conjunction in the antecedent is provable in T.

The provability of the second term follows from the assumptions.

The proof of $\forall y \Pr_T(\ulcorner \mathrm{Con}_T(\dot{y})\urcorner)$ can be constructed by formalizing the following argument (we assume that T is sufficiently strong): "Either T is inconsistent, and then it proves everything, or T is consistent, and then we can prove $\mathrm{Con}_T(\underline{n})$ by checking all proofs of length $\leq n$."

Hence (47) has a proof of length $O(\log n)$. □

For further improvements of these theorems, see Buss [1994].

7.2.3. Theorem 7.2.2 gives a worse speed-up for the length of proofs in proper extensions of T, moreover it requires that the extension proves Con_T. On the other hand it gives more explicit formulas on which the speed-up is attained. In particular it enables us to study the trade-off between the speed-up and the complexity of formulas. For instance consider statements $\mathrm{Con}_T(f(\underline{n}))$ for a primitive recursive f. Then $\mathrm{Con}_T(f(\underline{n}))$ are numeric instances of a (formalization of a) primitive recursive predicate. Thus we get *primitive recursive speed-up* on *primitive recursive formulas* etc.

We mention without a proof a related result where a speed-up is obtained for a simple formula in the fragment $T = I\Delta_0 + \Omega_1$. We shall denote 2^1_n by 2_n (the stack of n 2's).

7.2.4. Theorem. (Hájek, Montagna and Pudlák [1993]) *Let $T = I\Delta_0 + \Omega$. Then we have*

(1) $\|\Pr_T(\ulcorner \exists y(y = 2_{\underline{2_n}})\urcorner)\|_T = n^{O(1)}$;

(2) $\|\exists y(y = 2_{\underline{2_n}})\|_T = \Omega(2_n)$. □

7.3. Speed-up of GB over ZF. It is well-known that **GB** proves the same set of formulas as **ZF**. (We consider **ZF** with the axiom of choice.) Therefore it is very interesting to find out if the set formulas have proofs of approximately the same length in both theories. The answer is no; in fact there is a nonelementary speed-up as for cut-elimination. This seems to be typical for results obtained from cut-elimination or Herbrand theorem (and where a direct proof is not known).

This result is based on the following lemma due to Solovay (unpublished); a similar construction was considered by Vopěnka (unpublished).

7.3.1. Lemma. *There is a cut $I(x)$ in* **GB** *such that*

$$\mathbf{GB} \vdash \forall x(I(x) \to \mathrm{Con}_{\mathbf{ZF}}(x)).$$

It is outside of the scope of this chapter to give a proof of this lemma. Let us only very briefly describe the main idea. One can construct a sort of inner model of **ZF** in **GB** where the universe of sets is some cut. This model is constructed along with a satisfaction relation for it. Since the satisfaction relation is defined by a formula with class quantifiers, we cannot use induction to show the consistency of **ZF**. Instead we only show that the segment of numbers x such that there is no contradiction of length $\leq x$ is closed under successor. But this is exactly what we need. □

7.3.2. Theorem. (Pudlák [1986])
(1) $\|\mathrm{Con}_{\mathbf{ZF}}(2_{\underline{n}})\|_{\mathbf{GB}} = n^{O(1)}$;
(2) $\|\mathrm{Con}_{\mathbf{ZF}}(2_{\underline{n}})\|_{\mathbf{ZF}} = (2_n)^{\varepsilon}$, *for some constant* $\varepsilon > 0$.

Proof. To prove (1) we need, by Lemma 7.3.1, only to have a short proof of $I(2_{\underline{n}})$ in **GB**. The bound

$$\|I(2_n)\|_{\mathbf{GB}} = n^{O(1)}$$

follows from Theorem 3.4.1. The second part is contained in Theorem 7.2.2. □

A more precise computation gives a bound $\|\mathrm{Con}_{\mathbf{ZF}}(2_{\underline{n}})\|_{\mathbf{GB}} = O(n^2)$, which implies a lower bound on the speed-up of **GB** over **ZF** of the form $2_{\Omega(\sqrt{n})}$. An upper bound $2_{O(\sqrt{n})}$ complementing the above result was proved by Solovay [1990]. Let us observe that such bounds can be used to show that the estimate on the depth of formulas in a proof, Theorem 4.2.5, is asymptotically optimal.

8. Propositional proof systems

In this section we consider some concrete propositional proof systems. There are several reasons for studying these systems. Firstly they are natural systems which are used for formalization of the concept of a proof, in fact, they are good approximations of human reasoning. Some systems, especially resolution, are also used in automated theorem proving. Therefore it is important to know how efficient they are. Secondly they are suitable benchmarks for testing our lower bound techniques. Presently we are able to prove superpolynomial lower bounds only for the weakest systems; we shall give an example of a lower bound in section 9. Thirdly there are important connections between provability in some important theories of bounded arithmetic and the lengths of proofs in these propositional proof systems. This will be the topic of section 10.

8.1. Frege systems and its extensions. Frege systems are the most natural calculi for propositional logic and they are also used to axiomatize the propositional part of the first order logic in the Hilbert style formalizations. We have used a particular special case of a Frege system for presenting results on the lengths of proofs in first order logic.

8.1.1. To define a general Frege system, we need the concept of a Frege rule. A *Frege rule* is a pair $(\{\varphi_1(p_1,\ldots,p_n),\ldots,\varphi_k(p_1,\ldots,p_n)\},\varphi(p_1,\ldots,p_n))$, such that the implication $\varphi_1\wedge\ldots\wedge\varphi_k\to\varphi$ is a tautology. We use $p_1,\ldots,p_n$ to denote propositional variables. Usually we write the rule as

$$\frac{\varphi_1,\ldots,\varphi_k}{\varphi}.$$

When using the rule, we use actually its *instances* which are obtained by substituting arbitrary formulas for the variables $p_1,\ldots,p_n$. A Frege rule can have zero assumptions, in which case it is an *axiom schema*. A *Frege proof* is a sequence of formulas such that each formula follows from previous ones by an application of a Frege rule from a given set.

8.1.2. Definition. A *Frege system* F is determined by a finite complete set of connectives B and a finite set of Frege rules. We require that F be implicationally complete for the set of formulas in the basis B.

Recall that *implicationally complete* means that whenever an implication $\psi_1\wedge\ldots\wedge\psi_k\to\psi$ is a tautology, then ψ is derivable from $\psi_1,\ldots,\psi_k$. Rules such as modus ponens and cut ensure that the system is implicationally complete whenever it is complete.

An example of a Frege system is the propositional part of the proof system considered in section 2; it has 14 axiom schemas and one rule with two assumptions (modus ponens).

8.1.3. Note that in an application of a Frege rule (in particular also in axioms) we substitute arbitrary formulas for the variables in the rule, however we are not allowed to substitute in an arbitrary derived formula. It is natural to add such a rule. The rule is called *the substitution rule* and allows to derive from $\varphi(p_1,\ldots,p_k)$, with propositional variables $p_1,\ldots,p_k$, any formula of the form $\varphi(\psi_1,\ldots,\psi_k)$.[1]

8.1.4. Definition. A *substitution Frege system* SF is a Frege system augmented with the substitution rule.

8.1.5. The *extension rule* is the rule which allows to introduce the formula

$$p\equiv\varphi,$$

where p is a propositional variable and φ is any formula and the following conditions hold:

1. when introducing $p\equiv\varphi$, p must not occur in the preceding part of the proof or in φ;
2. such a p must not be present in the proved formula.

[1] In fact Frege used this rule originally and the idea of axiom schemas was introduced by von Neumann later.

If $\equiv$ is not in the basis, we can use an equivalent formula instead, e.g. $(p \to \varphi) \wedge (\varphi \to p)$. This rule is not used to derive a new tautology quickly, as it is the case of Frege and substitution rules, but its purpose is to abbreviate long formulas.

8.1.6. Definition. An *extension Frege system* EF is a Frege system augmented with the extension rule.[2]

8.1.7. The first question that we shall address is: how does the lengths of proofs depend on a particular choice of the basis of connectives and the Frege rules. If the basis is the same for two Frege systems F_1 and F_2, it is fairly easy to prove that they are polynomially equivalent. We have used this argument already in the previous sections. Let for instance

$$\frac{\varphi_1, \dots, \varphi_k}{\varphi}$$

be a rule in F_1. Since F_2 is implicationally complete, there exists a proof π of φ from $\varphi_1, \dots, \varphi_k$ in F_2. To simulate an instance of this rule obtained by substituting some formulas into it, we simply substitute the same formulas in π. Thus we get only linear increase of the size.

If the two bases are different, the proof is not so easy, but the basic idea is simple. One uses a well-known fact from boolean complexity theory that a formula in one complete basis can be transformed into an equivalent formula in another complete basis with at most polynomial increase in size, in fact, using a polynomial algorithm. This, of course, does not produce a proof from a proof, but one can show that it suffices to add pieces of proofs of at most polynomial size between the formulas to get one. Details are tedious, so we leave them out.

The same holds for substitution Frege and extension Frege systems. Thus we have:

8.1.8. Theorem. (Cook and Reckhow [1979], Reckhow [1976]) *Every two Frege systems are polynomially equivalent, every two substitution Frege systems are polynomially equivalent, and every two extension Frege systems are polynomially equivalent.* □

8.1.9. This still leaves three classes, moreover each can be considered also in the tree form and we can count the number of steps instead of the size, which gives altogether twelve possibilities. We shall show that these cases reduce to only three, if we identify polynomially equivalent ones (namely, Frege, extension Frege and the number of steps in substitution Frege).

The question about a speed up of sequence versus tree proofs has been solved in Theorem 4.1 for Frege systems which contain modus ponens. The same holds for extensions of such Frege systems by extension and substitution rules. We shall return to this question below. Now we shall consider the remaining ones. Let us first consider the relation of substitution Frege systems and extension Frege systems.

[2]Here I deviate slightly from the literature where the name *extended Frege system* is used, which, I think, is rather ambiguous.

8.1.10. Theorem. (Dowd [1985], Krajíček and Pudlák [1989]) *Every substitution Frege system is polynomially equivalent to every extension Frege system.*

Proof. By the theorem above, we can assume that both systems have the same language.

1. First we show a polynomial simulation of an extension Frege system by a substitution Frege system. Let an extension Frege proof of a tautology ψ be given, let $p_1 \equiv \varphi_1, \ldots, p_m \equiv \varphi_m$ be all formulas introduced by the extension rule listed in the order in which they were introduced. By Theorem 8.1.8 we can assume w.l.o.g. that our systems contain suitable connectives and suitable Frege rules. Using an effective version of the deduction theorem (whose easy proof we leave to the reader) we get, by a polynomial transformation, a proof of

$$p_1 \equiv \varphi_1 \wedge \ldots \wedge p_m \equiv \varphi_m \to \psi \tag{48}$$

which does not use the extension rule (i.e., a Frege proof). Now apply the substitution rule to (48) with the substitution $p_m \mapsto \varphi_m$. Thus we get

$$p_1 \equiv \varphi_1 \wedge \ldots \wedge p_{m-1} \equiv \varphi_{m-1} \wedge \varphi_m \equiv \varphi_m \to \psi. \tag{49}$$

From (49) we get by a polynomial size Frege proof

$$p_1 \equiv \varphi_1 \wedge \ldots \wedge p_{m-1} \equiv \varphi_{m-1} \to \psi.$$

We repeat the same until we get a proof of ψ.

2. The polynomial simulation of substitution Frege systems by extension Frege systems is not so simple. The idea of the proof is the following. Suppose we have simulated a substitution Frege proof until φ_j which is derived by a substitution from φ_i, say $\varphi_j = \varphi_i(\bar{p}/\bar{\alpha})$. Then we can derive φ_j by repeating the previous part of the proof with variables $\bar{p}$ replaced by $\bar{\alpha}$. However, repeating this, the new *Frege* proof would grow exponentially. The trick is to prove the formulas of the substitution Frege proof not for a particular substitution, but for the substitution, where the proof fails for the first time. In reality the proof does not fail (we start with a real proof), but it enables us to argue: "if it failed, then we could go on, hence we can go on in any case". Now the problem is for which propositional variables should the proof fail. Therefore we introduce an extra set of variables for each formula of the substitution Frege proof. The variables at some step will be defined using variables at the following steps of the proof. As this nesting may result in an exponential growth, we introduce them using the extension rule.

Now we shall argue formally. W.l.o.g. we can assume that the substitution Frege system has only modus ponens and axiom schemas as Frege rules. Let $(\varphi_1, \ldots, \varphi_m)$ be a substitution Frege proof. Let $\bar{p} = (p_1, \ldots, p_n)$ be all propositional variables of the proof. Take sequences $\bar{q}_i$ of length n consisting of new distinct variables, for $i = 1, \ldots, m \Leftrightarrow 1$, and denote by $\bar{q_m} = \bar{p}$. Let ψ_i be $\varphi_i(\bar{p}/\bar{q}_i)$, for $i = 1, \ldots, m$; thus $\psi_m = \varphi_m$. Let $\bar{\beta}_j$ be a sequence of n formulas defined as follows:

1. if φ_j is an axiom or is derived by modus ponens then $\bar\beta_j = \bar q_j$;
2. if φ_j is derived by substitution from φ_i, namely $\varphi_j = \varphi_i(\bar p/\bar\alpha)$, then $\bar\beta_j = \bar\alpha(\bar p/\bar q_j)$.

The extension Frege proof will start by introducing

$$q_{i,l} \equiv (\Psi_i \wedge \neg\psi_{i+1} \wedge \beta_{i+1,l}) \vee \ldots \vee (\Psi_{m-1} \wedge \neg\psi_m \wedge \beta_{m,l}),$$

where Ψ_j is $\psi_1 \wedge \ldots \wedge \psi_j$. We have to introduce these formulas in the order $i = m \Leftrightarrow 1, \ldots, 1$.

Then we add polynomial size proofs of

$$\Psi_{j-1} \wedge \neg\psi_j \to \psi_i \equiv \psi_i(\bar q_i/\bar\beta_j), \tag{50}$$

for $i < j$. To prove (50) we first derive

$$\Psi_{j-1} \wedge \neg\psi_j \to q_{i,l} \equiv \beta_{j,l}$$

from the axioms introducing $q_{i,l}$ and then successively construct proofs of the corresponding statements for subformulas of ψ_i.

Now we derive $\psi_1, \ldots, \psi_m$. Suppose we have proved $\psi_1, \ldots, \psi_{j-1}$. Consider three cases.

1. φ_j is an axiom. Then ψ_j is also an axiom.

2. φ_j was derived from φ_u, φ_v, $u, v < j$, $\varphi_u = \varphi_v \to \varphi_j$ by modus ponens. First derive Ψ_{j-1}. Then, using (50) with $i = u, v$, we get

$$\neg\psi_j \to \psi_u(\bar\beta_j) \wedge \psi_v(\bar\beta_j).$$

Since

$$\psi_u(\bar\beta_j) = \psi_v(\bar\beta_j) \to \psi_j(\bar\beta_j),$$

we get

$$\neg\psi_j \to \psi_j(\bar\beta_j).$$

As we are considering the case of modus ponens, $\bar\beta_j = \bar q_j$, hence we have derived $\neg\psi_j \to \psi_j$, whence we get ψ_j immediately.

3. φ_j was derived from φ_i, $i < j$ by substitution. Then ψ_j is just $\psi_i(\bar q_i/\bar\beta_j)$, thus (50) gives

$$\Psi_{j-1} \wedge \neg\psi_j \to \psi_i \equiv \psi_j$$

and we get ψ_j easily from $\psi_1, \ldots, \psi_{j-1}$.

Finally recall that ψ_m is the conclusion of the proof φ_m, thus we have the simulation. □

The simulation of extension Frege systems by substitution Frege systems was shown already in Cook and Reckhow [1979]. The other simulation has a simple "higher order" proof based on a relation to bounded arithmetic. Namely by Theorem 10.3.6 below, it suffices to prove the reflection principle for substitution Frege system in S^1_2, which is easy. This was observed independently by Dowd [1985] and Krajíček and Pudlák [1988].

8.1.11. It is an open problem whether Frege systems can simulate extension and substitution Frege systems. We conjecture that the answer is *no*. It seems, though it is not supported by any mathematical result, that the relation of Frege systems to extension Frege systems is the same as the relation of boolean formulas to boolean circuits in complexity theory. It is generally accepted that it is unlikely that formulas can simulate circuits with only polynomial increase in size; the uniform version of this conjecture is $\mathcal{NC}^1 \neq \mathcal{P}$; both conjectures are also open.

8.1.12. We shall now consider the number of steps in these systems. It turns out that the number of steps in Frege and extension Frege systems is, up to a polynomial, the size of extension Frege proofs.

8.1.13. Lemma. *If φ can be proved by a proof with n steps in an extension Frege system, then it can be proved by a proof with n steps in a Frege system; namely, we can omit the extension rule from the given system.*

Proof-sketch. Omit every instance of the extension rule $p \equiv \varphi$ and at the same time replace all occurrences of the variable p by φ. □

8.1.14. Lemma. *There exists a polynomial $f(x)$ such that for every tautology φ and every extension Frege proof of φ with n steps, there exists an extension Frege proof of φ whose size is $\leq f(|\varphi| + n)$.*

The idea of the proof is to introduce propositional variables for each *relevant subformula* of the proof and work with these variables instead of formulas. A subformula is relevant, if it is in constant depth from the root of some formula in the proof, where the constant is determined by the Frege system. We leave out further details. □

From these two lemmas we get immediately:

8.1.15. Theorem. (Statman [1977], Cook and Reckhow [1979]) *For every tautology φ, the minimal number of steps of a proof of φ in a Frege system, the minimal number of steps of a proof of φ in an extension Frege system and the minimal size of a proof of φ in an extension Frege system are polynomially related.* □

8.1.16. It can be shown that the minimal number of steps in a proof of a tautology in an extension Frege system can be exponentially larger than in a substitution Frege system Tsejtin and Čubarjan [1975], Krajíček [1989b]. Consider the tautology $\neg^{2^n}(p \vee \neg p)$, where $\neg^{2^n}$ denotes 2^n-times $\neg$. It is not very hard to show that the number of steps needed to prove it in a Frege system is $\Omega(2^n)$ (it is also possible to prove it by defining a winning strategy for Adversary in the game below). Thus the minimal size of an extension Frege proof for $\neg^{2^n}(p \vee \neg p)$ must be $2^{\Omega(n)}$. On the other hand it can be proved using only $O(n)$ steps in a substitution Frege system. This is based on the fact that it is possible to derive $q \to \neg^{2^{k+1}} q$ from $q \to \neg^{2^k} q$ in constant

number of steps: first derive $\neg^{2^k} q \to \neg^{2^{k+1}} q$ by substitution $q \mapsto \neg^{2^k} q$, and then use the transitivity of implication.

8.2. A game. The following game was devised as an approach to proving lower bounds on the lengths of propositional proofs. So far we were not able to get new lower bounds result using it; in fact it is even not so easy to interpret the known lower bounds using this game. However the game can be used at least to prove something about the structure of propositional proofs.

8.2.1. We shall call the game *Prover-Adversary game.* The game is determined by a complete set of propositional connectives B. There are two players *Prover* and *Adversary.* The aim of Prover is to prove a proposition φ and the aim of Adversary is to pretend that, for some assignment, the formula φ can have value 0 (=false). The game starts with Prover's asking φ and Adversary answering 0, and then Prover asks other propositions and Adversary assigns values to them. The game ends when there is a simple contradiction in the statements of the Adversary which means the following. Suppose we consider propositions in a basis of connectives B. Then a *simple contradiction* means that for some connective $\circ \in B$, and propositions $\varphi_1, \dots, \varphi_k$, Adversary has assigned values to $\varphi_1, \dots, \varphi_k$, $\circ(\varphi_1, \dots, \varphi_k)$ and they do not satisfy the truth table of $\circ$; e.g., he assigned 0 to φ, 1 to ψ and 1 to $\varphi \wedge \psi$.

We define that a proposition φ is *provable in this game,* if Prover has a winning strategy. A natural measure of complexity of such proofs is *the minimal number of rounds needed to convict any Adversary.*

It is easy to prove that the Prover-Adversary game as a proof system is sound and complete, (however it does not satisfy the definition of a propositional proof system 2.5). To prove the soundness, suppose φ is not a tautology. Then Adversary can simply evaluate the propositions on an input a for which $\varphi[a] = 0$. To prove the completeness, let Prover ask all subformulas of φ, including the variables.

The most interesting fact about the Prover-Adversary game is the relation of the number of rounds in the game to the number of steps in a Frege proof.

8.2.2. Proposition. *The minimal number of rounds in the Prover-Adversary game needed to prove φ is proportional to the logarithm of the minimal number of steps in a Frege proof of φ.*

More precisely, for every basis B and every Frege system F, there are constants c_1, c_2 such that for every tautology φ,

(i) if it has a proof with k steps in F, then it can be proved in $\leq c_1 \log k$ rounds and

(ii) if it can be proved in r rounds, then it can be proved in F in k steps with $\log k \leq c_2 r$.

Proof. 1. Let a Frege proof of φ be given, say $\varphi_1, \dots, \varphi_k$, with $\varphi_k = \varphi$. Consider conjunctions

$$\psi_i = (\cdots(\varphi_1 \wedge \varphi_2) \wedge \cdots) \wedge \varphi_i.$$

If Adversary tries to be consistent as long as possible, Prover needs only a constant numbers of questions to force him to assign 1 to an axiom. Thus he can force value 1 for ψ_1. Also he needs only a constant number of questions to get 0 for ψ_k, since $\varphi_k \mapsto 0$. Then he uses binary search to find an i such that $\psi_i \mapsto 1$ and $\psi_{i+1} \mapsto 0$. This takes $O(\log k)$ rounds. A constant number of rounds is needed to get $\varphi_{i+1} \mapsto 0$. Suppose φ_{i+1} was derived from $\varphi_{i_1}, \ldots, \varphi_{i_l}$, $i_1, \ldots, i_l \leq i$. For each of these premises it takes only $\leq \log i$ rounds to force 1 (or to get an elementary contradiction), since $\psi_i \mapsto 1$, – use binary search again. Once the premises got 1's and the conclusion 0, Prover needs only a constant number of questions to force an elementary contradiction.

2. Let a winning strategy for Prover be given, suppose it has r rounds in the worst case. We construct a sequent calculus proof of φ of size $2^{O(r)}$, which, as we know, can be transformed into a Frege proof with at most polynomial increase; we shall consider this transformation in more details below.

Consider a particular play P, let $\alpha_1, \ldots, \alpha_t$, $t \leq r$ be the questions asked by Prover, where we have added (or removed) negations, if Adversary answered 0 (in particular α_1 is $\neg\varphi$). Thus $\alpha_1 \wedge \cdots \wedge \alpha_t$ is false, hence $\rightarrow \neg\alpha_1, \ldots, \neg\alpha_t$ is a true sequent. Moreover, as easily seen, it has a proof with constant number of lines, since there is a *simple contradiction* in the statements $\alpha_1, \ldots, \alpha_t$. The proof of φ is constructed by taking proofs of all such sequents and then using cuts eliminating successively all formulas except of φ. This is possible due to the structure of the possible plays. Namely,

1. for each play P there is another play P' in which all the questions and answers are the same except for the answer which corresponds to the last question of P; P' may be longer than P;
2. for every two plays P, P', if they have the same questions up to the i-th one, say $\alpha_1, \ldots, \alpha_i$, then they have the same answers up to the $i \Leftrightarrow$1-st one and different i-th answer.

Finally observe that the number of such sequents is at most 2^r, which gives the bound. □

Let us note that the proof constructed from the game has a very special structure. Firstly it is in a *tree form*; secondly, it is like a dual to cut-free proofs, since it uses everywhere *only the cut rule*, except for the leaves of the proof tree.

Let us note also that we can characterize the *size* of proofs in Frege systems in a similar way, we have only to add the logarithm of the maximal size of a query to the cost of the play.

8.2.3. We return to the problem about the relation of the lengths of proofs as sequences and lengths of proofs as trees. We would like to use the proof obtained by transforming a general proof into the Prover-Adversary game and then back to a proof. The resulting proof has the number of steps polynomial in the original number of steps k and it is in a tree form, but it is a sequent proof. We shall analyze its transformation into a Frege proof in order to see that the tree structure can be preserved also in the Frege form.

First we shall assume that we have a Frege system F_0 with suitable rules. Let $\to \neg\alpha_1, \ldots, \neg\alpha_t$ be a sequent on the leaf of the sequent proof. We shall replace it by

$$(\cdots(\neg\alpha_1 \vee \neg\alpha_2)\cdots) \vee \neg\alpha_t. \tag{51}$$

We shall start with proofs of such sequents. We know that some $(\cdots(\neg\alpha_{i_1} \vee \neg\alpha_{i_2})\cdots) \vee \neg\alpha_{i_c}$, where c is a constant determined by the basis that we use, is a tautology. Moreover this tautology has a constant size proof, as it comes from a simple contradiction. Hence it also has a constant size tree proof. To get (51) we have to add the remaining disjuncts using a tree proof. It is quite easy, if we use the ideas shown in section 4.

Let us note that $t = O(\log k)$, thus also the proof of (51) is $O(\log k)$.

The rest of the proof is the same as in the sequent case, provided that we have a cut rule in the form

$$\frac{A \vee B,\ C \vee \neg B}{A \vee C}.$$

If we have a general Frege system F, we have to simulate each application of a rule of F_0 by several, however a constant number, of rules of F. Structurally it means that we replace each node with its in-going edges of the original tree by a constant size tree. Thus in general we get a larger tree; but the point is that the new tree has depth also $O(\log k)$, thus it has size also polynomial in k. Hence have proved:

8.2.4. Theorem. (Krajíček [1994a]) *For* every *Frege system there exists a polynomial $p(x)$ such that for every tautology φ*

$$\|\varphi\|^{tree}_{steps} \leq p(\|\varphi\|^{sequence}_{steps}).$$

□

8.3. Resolution. The most important propositional calculus for automated theorem proving is the resolution system. It is fairly easy to implement and there is a variety of heuristics there that one can try in the proof search.

The idea can be simply explained as follows. Suppose that we want to prove a tautology which is a DNF. Thus it suffices to derive a contradiction from its negation, which is a CNF, say $\bigwedge_{i\in I} \delta_i$. This is the same as to derive a contradiction from the set $\{\delta_i\}_{i\in I}$. If we think of disjunctions as obtained by applying the *set* operator of disjunction to a *set* of variables and its negations, then we need only a single rule – the cut. The contradiction then would be the disjunction of an empty set.

In the usual terminology we call variables and negated variables *literals*; the disjunctions are represented simply as sets of literals and they are called *clauses*, the cut rule is called *resolution*. As we are proving a contradiction from assumptions, we rather talk about a *refutation* than a proof. Thus a *resolution refutation* of a set of clauses $\mathcal{C}$ is a sequence starting with the clauses of $\mathcal{C}$, the following clauses are derived by resolution and the last clause should be $\emptyset$.

8.4. Extended resolution. Though a lot of interesting tautologies are DNF's, we would like to be able to prove also others. There is a natural way, in which we can extend the resolution system to be able to talk about arbitrary formulas; namely, we introduce variables for formulas and add the defining clauses.

Formally *extended resolution* for the basis $\{\wedge, \vee, \neg\}$ and variables $p_1, \ldots, p_n$ is resolution augmented with the clauses obtained from the CNF's of

$$q_{p_i} \equiv p_i,\ q_{\neg\varphi} \equiv \neg q_\varphi,\ q_{\varphi_1\wedge\varphi_2} \equiv q_{\varphi_1} \wedge q_{\varphi_2},\ q_{\varphi_1\vee\varphi_2} \equiv q_{\varphi_1} \vee q_{\varphi_2},$$

for all formulas in the language $\{p_1, \ldots, p_n, \wedge, \vee, \neg\}$, where q's are some new distinct variables. E.g., $q_{\neg\varphi} \equiv \neg q_\varphi$ is replaced by the two clauses $\{q_{\neg\varphi}, q_\varphi\}$ and $\{\neg q_{\neg\varphi}, \neg q_\varphi\}$. We define it for other bases similarly.

While resolution is much weaker than Frege systems, the extended resolution system is polynomially equivalent to extension Frege systems. The simulation of extension Frege system by extended resolution is based on essentially the same idea as Lemma 8.1.14.

8.5. Bounded depth Frege systems. Intermediate between the resolution system and the Frege systems are bounded depth Frege systems. They are very important for bounded arithmetic, see section 10. Also they are the strongest systems for which we are able to prove exponential lower bounds.

Consider formulas in basis $\{\wedge, \vee, \neg\}$. We define inductively classes Σ_i and Π_i of such formulas. Σ_0 and Π_0 are just literals. A formula φ is in Σ_{i+1}, if it is the disjunction of formulas from Σ_{i+1} or the conjunction of formulas from Σ_i or a negation of a formula from Σ_i. The classes Π_{i+1} are defined dually. A formula has *depth* d, if it is in $\Sigma_d \cup \Pi_d$.

A *depth* d *Frege proof* is a Frege proof, where all formulas are depth d. If a suitable set of rules is chosen such a system is complete for depth d tautologies.

Krajíček [1994a] has shown that there are depth d tautologies which have polynomial size tree-like proofs in a depth $d+1$ Frege system, but only exponential size tree-like proofs in a depth d Frege system, and, conversely, there are depth d tautologies which have polynomial size (general) proofs in a depth d Frege system, but only exponential size tree-like proofs in a depth d Frege system. (More precisely, one has to use refutations instead of proofs.) It is not known, if there is such a speed up for sequence-like proofs. Also it is an open problem, if there is a d_0 such that for every $d \geq d_0$, depth d and $d+1$ systems can be separated in such a way using tautologies of depth $\leq d_0$. On the other hand there is a sequence of tautologies of depth 3 which have polynomial size (unbounded depth) Frege proofs, but only exponentially large depth d Frege proofs for every constant d (see Buss [1987], Krajíček, Pudlák and Woods [1995], Pitassi, Beame and Impagliazzo [1993] and Beame et al. [1992]). The tautologies express a very simple theorem – the pigeonhole principle. We shall prove a lower bound for resolution refutations of sets of clauses expressing the pigeonhole principle in the next section.

8.6. Propositional sequent calculus. We have already mentioned that sequent proof systems are polynomially equivalent to the Frege system that we considered in the first part of the chapter, hence to all Frege systems. Thus it remains to mention the cut-free propositional sequent calculus. Since we know about nonelementary speed-up in the case of first order logic, it is not surprising that there is a speed-up also for propositional logic. The speed-up is exponential Takeuti [1990], and, trivially, cannot be larger. An exponential speed-up (slightly worse) follows also from the speed-up of unbounded depth Frege system versus bounded depth Frege system (using the fact that a cut-free proof of a bounded depth tautology is also bounded depth).

8.7. Propositional natural deduction. The natural deduction system is essentially a Frege system with an additional rule which allows to prove an implication $\varphi \to \psi$ by taking φ as an assumption and deriving ψ. The fact that this rule can be simulated in a Frege system is called *the deduction theorem* and the rule is called *the deduction rule*. Mutual simulations of the sequent calculus and natural deduction were shown by Gentzen [1935] and they are actually polynomial simulations, see Eder [1992]. The power of the deduction rule has been investigated in more detail by Bonet and Buss [1993].

8.8. Quantified propositional proof systems. It seems unlikely that there is a proof system for propositional logic which can polynomially simulate all other proof systems (see Krajíček and Pudlák [1989] for the relation of this question to problems in computational complexity). Thus it is interesting to look for stronger and stronger proof systems. How can one construct a system stronger than extension Frege systems? One possible way is to extend the expressive power of the language used in the proofs and the most natural extension is to take quantified propositional formulas.[3]

The language of quantified propositional logic consists of *quantified propositional formulas* which are usual propositional formulas with quantifiers binding some propositional variables. The semantics of such formulas is clear. E.g., the following is a quantified propositional tautology

$$\forall p, q \exists r((p \to q) \to (p \to r) \wedge (r \to q)).$$

As a logical calculus we simply modify either a Hilbert style or Gentzen sequent first order calculus. Again, we give only an example. Consider the axiom schema (5.1) of section 2

$$\Phi(t) \to \exists x \Phi(x).$$

We use this schema in quantified propositional logic as it stands, the only point is that now there is no distinction between terms and subformulas. So precisely stated it is as follows. Let $\varphi(p)$ be a quantified propositional formula with a free variable p

[3]It is interesting that a quantified propositional calculus was introduced by Russell [1906] as a *"theory of implication"*.

and let ψ be any quantified propositional formula, then the following formula is an axiom

$$\varphi(p/\psi) \to \exists x \varphi(x).$$

It is interesting to investigate the proof systems for all of quantified propositional logic, but we would also like to know, if such systems enable us to prove ordinary propositions faster. This seems plausible, as quantified propositional formulas can define functions in $\mathcal{PSPACE}$, thus very likely they have stronger expressive power than ordinary propositional formulas. But even if this were true, it would not necessarily imply that, say, the quantified propositional sequent calculus has shorter proofs for some propositional tautologies. Also we cannot exclude that the quantified propositional sequent calculus is stronger, but at the same time quantified propositional formulas of polynomial size define the same functions as ordinary propositional formulas of polynomial size. The only relation that we know for sure is that it polynomially simulates substitution (hence also extension) Frege systems, see Krajíček and Pudlák [1990].

Important applications in bounded arithmetic were found by Dowd [1979] and Krajíček and Takeuti [1990]; for further applications in bounded arithmetic, see section 10 and Krajíček and Pudlák [1990].

A related question is, how strong is the propositional part of the first order calculus. Let us consider the Hilbert style calculus of section 2. If we had only propositional variables, then it is just a Frege system. However, if we have some predicate, say $P(x)$, then we can code the propositional variable p_i using the first order variable x_i as $P(x_i)$. This enables us to code all *quantified* propositional formulas, thus we get at least the power of the quantified propositional calculus. However, using a suitable representation we can simulate arbitrarily strong propositional proof system, see 10.4.1 below.

8.9. "Mathematical" proof systems. Let us have look at the problem about the length of proofs in propositional logic from the point of view of complexity theory. The set of propositional tautologies is a *co*$\mathcal{NP}$-complete sets, say L. A proof system is a relation $R(x, y)$ computable in polynomial time such that

$$x \in L \;\equiv\; \exists y R(x, y).$$

A proof of x is a y such that $R(x, y)$. Thus we can take an arbitrary *co*$\mathcal{NP}$-complete set and an arbitrary R for it and ask what are the lengths of such proofs. We shall consider three examples of such calculi.

8.9.1. The Hajós calculus. In the first example the set L consists of graphs which cannot be colored by three colors. Hajós [1961] has proved that every such graph can be obtained as follows.

1. Start with K_4, the complete graph on four vertices, and apply the following operations:

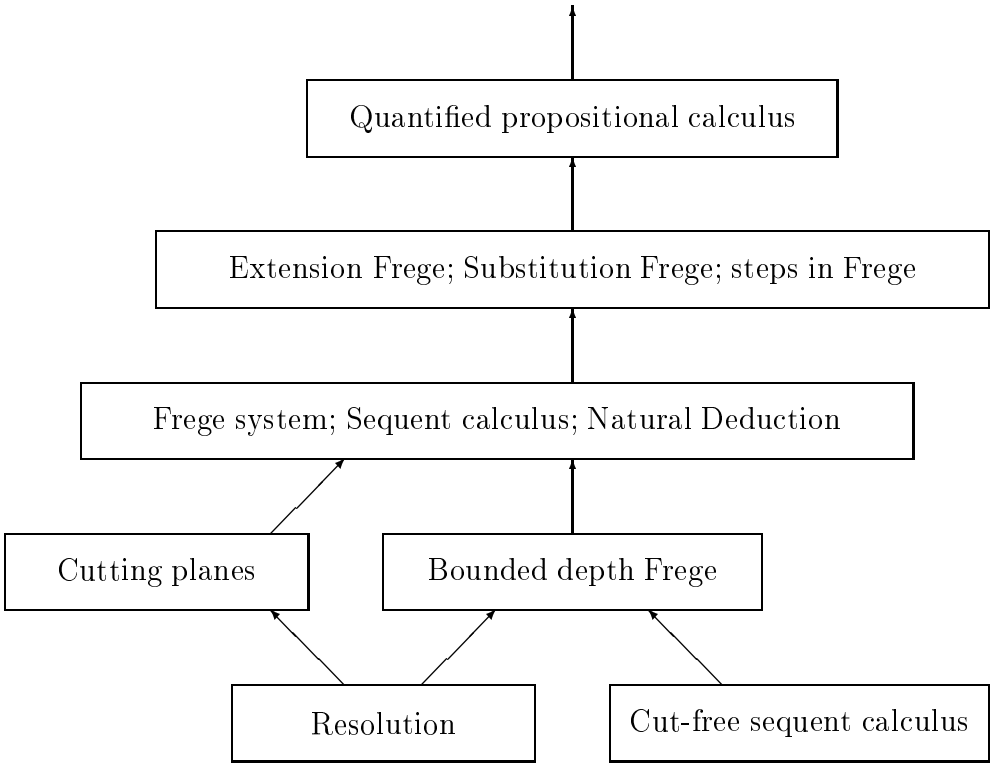

Figure 1: The hierarchy of propositional proof systems

2. edge/vertex introduction: add any new vertices and any new edges to a constructed graph;

3. join: if G_1 and G_2 has already been constructed, G_1 and G_2 with disjoint sets of vertices, (a_1, b_1) an edge in G_1, (a_2, b_2) an edge in G_2, then construct a new graph by contracting a_1 with a_2, deleting the edges $(a_1, b_1), (a_2, b_2)$ and adding the edge (b_1, b_2);

4. contraction: contract any two non-adjacent vertices in a constructed graph.

On the other hand it is quite easy to prove that no graph obtained in this way is 3-colorable. A *proof* of the fact that G is not 3-colorable in the Hajós calculus is a sequence where K_4 is used as an axiom, the three rules above are used to construct new graphs and where the last graph is G. Hajós' theorem asserts that this calculus is complete for graphs which are not 3-colorable.

Surprisingly Pitassi and Urquhart [1992] have shown that the Hajós calculus is polynomially equivalent to extension Frege systems. This means that

1. there is a polynomial time computable function which to each tautology φ and its extension Frege proof d assigns a graph G and a proof h in the Hajós calculus that G is not 3-colorable;
2. and *vice versa,* there is a polynomial time computable function which to each graph G and a proof h in the Hajós calculus that G is not 3-colorable assigns a tautology φ and its extension Frege proof d .

This shows that the concept of extension Frege systems is quite robust and that it will be very hard to prove that there is no polynomial bound on shortest proofs in the Hajós calculus.

8.9.2. Nullstellensatz. The second example are systems of algebraic equations over finite fields. Let

$$\begin{aligned} f_1(x_1, \ldots, x_n) &= 0 \\ &\vdots \\ f_m(x_1, \ldots, x_n) &= 0 \end{aligned} \tag{52}$$

be a system of algebraic equations over a field F. The famous Hilbert's Nullstellensatz says that (52) does *not* have a solution in $\overline{F}$ (the algebraic closure of F) iff there exist polynomials $g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n)$ such that

$$\sum_{i=1}^{m} g_i(x_1, \ldots, x_n) f_i(x_1, \ldots, x_n) = 1 \tag{53}$$

in the ring of polynomials over F.

We shall make some additional assumptions. We shall assume that F is finite and that equations (52) can have solutions only in F. The last condition can be ensured by adding equations $\prod_{a \in F}(x_i \Leftrightarrow a) = 0$ for $i = 1, \ldots, n$. Then such sets of unsolvable systems of equations are $co\mathcal{NP}$-complete. Furthermore we shall assume that polynomials are given as sums of monomials. Then (53) can be decided in

polynomial time, since we can expand the sum of products into a sum of monomials where the number of monomials is polynomial in the number of monomials of polynomials g_i and f_i, $i = 1, \ldots, n$. Thus we can think of the system of polynomials $g_1, \ldots, g_m$ as *a proof that (52) is unsolvable.* (Let us remark that in this special case the proof of the Nullstellensatz is easy, so this proof system is not based on a deep result.)

This system is not known to be equivalent to another proof system, it is weaker than Frege systems and there are superpolynomial lower bounds for a sequence of unsolvable systems. The main application of this approach is in the works of Beame et al. [1996] and Buss et al. [1996/1997], proving independence of counting principles in bounded depth Frege systems and in bounded arithmetic (this was first proved by Ajtai [1994b] using a different, and very deep proof).

A related system, called the *polynomial calculus*,[4] was introduced in Clegg, Edmonds and Impagliazzo [1996]. In this system we derive equations sequentially using additions and multiplications by arbitrary polynomials. Alternatively, it is just equational calculus with no variables allowed. For a given bound d on degree of polynomials occurring in the proof, the system is stronger than the Nullstellensatz system. If d is a constant, it is still decidable in polynomial time, if there is a proof of a given polynomial from a given set of polynomials.

Finally we consider a proof system which uses ideas of *linear programming* which was introduced in W. Cook, Coullard and Turán [1987].

8.9.3. Cutting plane proof system. This system is, in a sense, an extension of resolution; in particular it is also a refutation system for a set of clauses. However, instead of clauses we use linear inequalities which adds power to the system.

A *proof line* is an expression

$$a_1 p_1 + \cdots + a_n p_n \geq B, \tag{54}$$

where $a_1, \ldots, a_n, B$ are integers. We allow also expressions of the form $0 \geq B$. For a given clause C we represent literals p_i identically and $\neg p_i$ by $1 \Leftrightarrow p_i$. Let $f_1, \ldots, f_k$ be the linear terms expressing the literals of C. Then we represent C by the expression

$$f_1 + \cdots + f_k \geq 1.$$

(Of course, to get an expression of the form (54), we have to collect the constant terms on the right hand side; also we collect constant and other terms after each application of a rule.) The axioms and derivation rules are

1. axioms are all translations of the clauses in question and the expressions $p_i \geq 0$, $\Leftrightarrow p_i \geq \Leftrightarrow 1$;

2. addition: add two lines;

3. multiplication: multiply a line by a positive integer;

[4] Another name proposed for this calculus is the *Groebner proof system.*

4. division: divide a line (54) by a positive integer c which divides evenly $a_1, \ldots, a_k$ and *round-up the constant term on the right hand side*, i.e., we get

$$\frac{a_1}{c}p_1 + \cdots + \frac{a_n}{c}p_n \geq \left\lceil \frac{B}{c} \right\rceil .$$

(Note that on the left hand side we have integers, thus rounding up is sound.) A contradiction is obtained, when we prove $0 \geq 1$.

We suggest to the reader, as an easy exercise, to check that this system simulates resolution. Goerdt [1991] proved that Frege systems polynomially simulate the cutting plane proof system. Furthermore, Buss and Clote [1996] proved that the cutting plane system with the division rule restricted to the division by 2 (or any other constant > 1) polynomially simulates the general system. Recent success in proving exponential lower bounds on the lengths of cutting plane proofs (see section 9.3) gives us also interesting separations. The cutting plane proof system cannot be simulated by bounded depth Frege systems as it proves the pigeonhole principle (see Cook, Coullard and Turán [1987]) using polynomial size proofs. The cutting plane proof system does not polynomially simulate bounded depth Frege systems Bonet, Pitassi and Raz [1997a], Krajíček [1997a], Pudlák [1997].

9. Lower bounds on propositional proofs

In this section we give an example of a lower bound proof in propositional logic. Our lower bound will be an exponential lower bound on the size of resolution proofs of the *pigeonhole principle*. The first such bound for unrestricted resolution was proved by Haken [1985]. Unfortunately his proof cannot be generalized to stronger systems, (at least nobody has succeeded in doing it). Therefore we shall apply a technique of Ajtai [1994a], which he used for bounded depth Frege systems. The case of resolution, which can be considered as a depth one Frege system, is simpler than for larger depths and thus can serve as a good introduction to more advanced results.

9.1. A general method. Before we consider the concrete example, we shall present a general framework for lower bound proofs, which can be applied to some existing proofs and, maybe, can be also used for some new proofs. A general description of what is going on in lower bound proofs is always useful, since, when proving a lower bound, we are working with nonexisting things (the short proofs whose existence we are disproving) and therefore it is difficult to give any intuition about them.

The basic idea of our approach is as follows. Suppose that we want to show that $(\alpha_1, \alpha_2, \ldots, \alpha_m)$ is not a proof of α. Let L be the set of subformulas of $\alpha_1, \alpha_2, \ldots, \alpha_m$ and α. L is a partial algebra with operations given by the connectives. Suppose that we have a boolean algebra B and a homomorphism $\lambda : L \to B$ such that $\lambda(\alpha) \neq 1_B$. Then α cannot be among $\alpha_1, \ldots, \alpha_m$, since $\lambda(\varphi) = 1_B$ for every axiom and this is preserved by Frege rules. In this form the method cannot work: if α is a tautology (and we are interested only in tautologies), then $\lambda(\alpha) = 1_B$. Therefore we have to

modify it. We take only some subsets $L_i \subseteq L$ and $\lambda_i : L_i \to B_i$ for different boolean algebras B_i.

Now we shall describe this method in details. Let

$$\frac{\varphi_1(p_1, \dots, p_\ell), \dots, \varphi_k(p_1, \dots, p_\ell)}{\varphi(p_1, \dots, p_\ell)}$$

be a Frege rule R. We shall associate with it the set L_R of all subformulas of $\varphi_1, \dots, \varphi_k$ and φ. If

$$\frac{\varphi_1(\psi_1, \dots, \psi_\ell), \dots, \varphi_k(\psi_1, \dots, \psi_\ell)}{\varphi(\psi_1, \dots, \psi_\ell)}$$

is an instance of R, we associate with it the set

$$L_{R(\vec{\psi})} = L_{R(\psi_1,\dots,\psi_\ell)} = \{\alpha(\psi_1, \dots, \psi_\ell); \alpha(p_1, \dots, p_\ell) \in L_R\}.$$

Let B be a boolean algebra. A *homomorphism* $\lambda : L_{R(\vec{\psi})} \to B$ is a mapping which maps connectives onto corresponding operations in B, i.e.,

$$\lambda(\neg\varphi) = \neg_B \lambda(\varphi)$$

$$\lambda(\varphi \vee \psi) = \lambda(\varphi) \vee_B \lambda(\psi)$$

etc.

The following lemma formalizes our method.

9.1.1. Lemma. *Let $(\alpha_1, \alpha_2, \dots, \alpha_m)$ be a Frege proof using a set of assumptions S. Suppose the following conditions are satisfied:*

1. *For every formula α_i of the proof we have a boolean algebra B_i and an element $b_i \in B_i$. Furthermore, if $\alpha_i \in S$, then $b_i = 1_{B_i}$.*
2. *For every instance of a rule $R(\vec{\psi})$ of the proof we have a boolean algebra $B_{R(\vec{\psi})}$ and a homomorphism $\lambda_{R(\vec{\psi})} : L_{R(\vec{\psi})} \to B_{R(\vec{\psi})}$.*
3. *For every formula α_i of the proof and and every instance of a rule $R(\vec{\psi})$ where $\alpha_i \in L_{R(\vec{\psi})}$, we have an embedding $\kappa_{i,R(\vec{\psi})} : B_i \to B_{R(\vec{\psi})}$ so that $\kappa_{i,R(\vec{\psi})}(b_i) = \lambda_{R(\vec{\psi})}(\alpha_i)$.*

Then

$$b_1 = 1_{B_1}, \dots, b_m = 1_{B_m}.$$

The proof of this lemma is based on the following observation:

9.1.2. Lemma. *A Frege rule is sound in any boolean algebra.*

Proof. Suppose for some assignment of values from B we get the value 1_B for the assumptions but a value $b < 1_B$ for the conclusion. Take a homomorphism $\kappa : B \to \{0, 1\}$ such that $\kappa(b) = 0$. Then we get a contradiction with the soundness of the rule for the algebra $\{0, 1\}$. □

Proof of Lemma 9.1.1. We shall use induction. If $\alpha_1 \in S$, then $b_1 = 1_{B_1}$, otherwise α_1 is an instance of a logical axiom, say, $R(\vec{\psi})$ (a rule without assumptions). Thus, by Lemma 9.1.2, $\lambda_{R(\vec{\psi})}(\alpha_1) = 1_{B_{R(\vec{\psi})}}$. Hence

$$\kappa_{1,R(\vec{\psi})}(b_1) = \lambda_{R(\vec{\psi})}(\alpha_1) = 1_{B_R(\vec{\psi})}.$$

Since $\kappa_{1,R(\vec{\psi})}$ is an embedding, $b_1 = 1_{B_1}$. The induction step is similar. □

It may seem at the first glance that it does not make sense to talk about boolean algebras B_i, since we can simply take all of them to be 4-element algebras, and thus isomorphic. It turns out, however, that in applications nontrivial boolean algebras B_i appear quite naturally. They have to reflect the properties of the tautologies that we consider.

Let us remark that this is only one possible interpretation of some lower bound proofs and there are other interpretations. In particular other interpretations are based on the idea of forcing, due to Ajtai [1994a], and partial boolean algebras, due to Krajíček [1994b]; let us note that using partial boolean algebras one can characterize (up to a polynomial) the length of proofs in Frege systems.

Another tool which we shall use are *random restrictions*. They have been successfully applied for proving lower bounds on bounded depth boolean circuits and later also for lower bounds on proofs in bounded depth Frege systems by Ajtai [1994a,1990,1994b], Beame et al. [1992], Beame and Pitassi [1996], Bellantoni, Pitassi and Urquhart [1992], Krajíček [1994a], Krajíček, Pudlák and Woods [1995] and Pitassi, Beame and Impagliazzo [1993]. The idea is to assign more or less randomly 0's and 1's to some variables. Then many conjunctions and disjunctions became constant and thus the circuits, or the formulas, can be simplified. For the reduced formulas it is then much easier to construct boolean algebras with the required properties.

9.2. An exponential lower bound on the pigeonhole principle in resolution. Let D and R be disjoint sets with cardinalities $|D| = n+1$ and $|R| = n$. We shall consider the proposition PHP_n stating that there is no $1 \Leftrightarrow 1$ mapping from D onto R. (This is a weaker proposition than just: "there is no $1 \Leftrightarrow 1$ mapping of D *into* R", thus the lower bound is a stronger statement.) We denote by $p_{ij}, i \in D, i \in R$ propositional variables (meaning that i maps onto j in the alleged mapping). We shall consider clauses made of p_{ij} and $\neg p_{ij}$, furthermore we shall use the true clause $\top$ and the false, or empty, clause $\bot$. PHP_n is the negation of the following set of clauses

$$\begin{array}{lll} \bigvee_{j\in R} p_{ij}, & \text{for} & i \in D; \\ \bigvee_{i\in R} p_{ij}, & \text{for} & j \in R; \\ \neg p_{ij} \vee \neg p_{ik}, & \text{for} & j \in D, j,k \in R, j \neq k; \\ \neg p_{ji} \vee \neg p_{ki}, & \text{for} & j,k \in D, j \neq k, i \in R. \end{array}$$

Let M be the set of partial one-to-one mappings $D \to R$. We shall consider boolean algebras determined by subsets $T \subseteq D \cup R, |T| < n$, as follows. Let V_T be a subset of partial matchings g such that

1. $T \subseteq \mathrm{dom}(g) \cup \mathrm{rng}(g)$;
2. $\forall (i,j) \in g (i \in T \vee j \in T)$.

The boolean algebra associated with T is $P(V_T)$, the boolean algebra of subsets of V_T.

In order to be able to assign a value to a clause Δ in $P(V_T)$, a certain relation of T to Δ must be satisfied. We define that Δ is *covered* by T if

1. $p_{ij} \in \Delta \Rightarrow i \in T \vee j \in T$;
2. $\neg p_{ij} \in \Delta \Rightarrow i \in T \wedge j \in T$.

The clauses $\top$ and $\bot$ are covered by any set T. Suppose Δ is covered by T, then the *value* of Δ in $P(V_T)$ is the set

$$b^T_\Delta = \{g \in V_T; \quad g(i) = j \text{ for some } p_{ij} \in \Delta, \quad \text{or} \quad g(i) \neq j \text{ for some } \neg p_{ij} \in \Delta$$
$$\text{or} \quad g^{-1}(j) \neq i \text{ for some } \neg p_{ij} \in \Delta\}.$$

The following can be easily checked:

9.2.1. Lemma. *If Δ is one of the clauses of PHP_n and T covers Δ, $|T| < n$, then $b^T_\Delta = 1_{P(V_T)}$.* □

Suppose $T \subseteq T'$, $|T'| < n$, then there is a natural mapping $\lambda_{T,T'} : P(V_T) \to P(V_{T'})$ defined by

$$\lambda_{T,T'}(b) = \{g' \in V_{T'} : \exists g \in b (g \subseteq g')\}\,.$$

9.2.2. Lemma. *Let $T \subseteq T'$, $|T'| < n$. Then $\lambda_{T,T'}$ is an embedding of the boolean algebra $P(V_T)$ into $P(V_{T'})$.*

Proof. All properties are trivial except for the following one: $\lambda_{T,T'}$ is injective. This property follows from the fact that each $g \in V_T$ can be extended to a $g' \in V_T$, which holds due to the fact that $|T'| < n$. □

Consider an instance of the cut rule

$$\frac{, \vee p_{ij} \qquad \Delta \vee \neg p_{ij}}{, \vee \Delta}$$

Suppose we have chosen $P(V_{T_i}), i = 1,2,3$ as the boolean algebra for $, \vee p_{ij}$, $\Delta \vee \neg p_{ij}$ and $, \vee \Delta$ respectively. Then we choose $P(V_T)$ with $T = T_1 \cup T_2 \cup T_3$ for this rule. We only have to ensure that $|T| < n$. Since T covers all subformulas involved, we can define their values in $P(V_T)$. The condition that this is a homomorphism of $\neg$ and $\vee$ is easy to check. By Lemma 9.2.2 we have also the necessary embeddings.

The simplest way to ensure $|T| < n$ for the rules is to choose the covering sets of the formulas of size $< n/3$. This is not always possible (take e.g. $p_{11} \vee p_{22} \vee \ldots p_{nn}$), therefore we apply random restrictions.

Suppose we assign 0's and 1's to some variables p_{ij} and leave the other as they are. If we do it for all formulas in a proof, the resulting sequence will be a proof again. However some initial clauses may reduce to $\bot$, so we cannot argue that $\bot$

cannot be derived from them by a short proof. Therefore the restrictions must reflect the nature of the tautology in question.

Let $g \in M$ be a partial one-to-one mapping. We shall associate with g the partial assignment defined by

$$\begin{array}{ll} p_{ij} \to 1 & \text{if} \quad (i,j) \in g; \\ p_{ij} \to 0 & \text{if} \quad i \in \mathrm{dom}(g) \text{ or } j \in \mathrm{rng}(g), \quad \text{but} \quad (i,j) \notin g; \\ p_{ij} \to p_{ij} & \text{otherwise.} \end{array}$$

Given a clause Δ, we define Δ^g to be

1. $\top$ if some $p_{ij} \in \Delta$ is mapped to 1 or some p_{ij} such that $\neg p_{ij} \in \Delta$ is mapped to 0,
2. otherwise it is the clause consisting of all literals which are not 0.

Let us denote by $D' = D \Leftrightarrow \mathrm{dom}(g), R' = R \Leftrightarrow \mathrm{rng}(g), n' = |R'|$. Clearly Δ^g is a clause with variables $p_{ij}, i \in D', j \in R'$. If Δ is a clause of PHP_n then Δ^g is either $\top$ or becomes a clause of $PHP_{n'}$ (on D' and R'). Denote by $M_{n,n'}$, the set of all partial one-to-one mappings of size $n \Leftrightarrow n'$. The following is the key combinatorial lemma for the proof of the lower bound.

9.2.3. Lemma. *Let $n' = \lfloor n^{1/3} \rfloor$, let Δ be an arbitrary clause. Then for a $g \in M_{n,n'}$, chosen with uniform probability, the probability that Δ^g can be covered by a set of size $< \frac{1}{3}n'$ is at least*

$$1 \Leftrightarrow 2^{\varepsilon n^{1/3}},$$

where $\varepsilon > 0$ is a constant.

We shall use the following simple estimate.

9.2.4. Lemma. *Let $a, b, l \leq n, A \subseteq \{1, \ldots, n\}, |A| = a$. Take a random $B \subseteq \{1, \ldots, n\}, |B| = b$, with uniform probability. Then*

$$\mathrm{Prob}(|A \cap B| \geq l) \leq \left(\frac{eab}{nl}\right)^l.$$

Proof.

$$\begin{aligned} \mathrm{Prob}(|A \cap B| \geq l) &\leq \sum_{\{a_1,\ldots,a_\ell\} \subseteq A} \mathrm{Prob}(a_1 \in B, \ldots, a_\ell \in B) \\ &= \binom{a}{l} \cdot \frac{b}{n} \cdot \frac{b \Leftrightarrow 1}{n \Leftrightarrow 1} \cdot \cdots \cdot \frac{b \Leftrightarrow l+1}{n \Leftrightarrow l+1} \\ &\leq \left(\frac{ea}{l}\right)^l \cdot \left(\frac{b}{n}\right)^l \leq \left(\frac{eab}{nl}\right)^l. \end{aligned}$$

□

Proof of Lemma 9.2.3. Let us denote by $l = \lfloor \frac{1}{3}n' \rfloor$. Let Δ be given. We shall simplify the situation by replacing each $\neg p_{ij} \in \Delta$ by

$$\bigvee_{i' \neq i} p_{i'j} \vee \bigvee_{j' \neq j} p_{ij'}.$$

This operation commutes with the restriction and the new clause is covered by $T, |T| \leq l$, iff the old one is, since $\ell < n' \Leftrightarrow 2$. Thus we can assume that Δ contains only positive literals. Such a Δ is determined by the graph

$$E = \{(i,j); p_{ij} \in \Delta\}.$$

Let

$$a = \frac{n^{2/3}}{40}.$$

From now on we shall omit the integer part function and assume that all numbers are integers. This introduces only inessential errors. Furthermore denote by

$$A = \{j \in R; \deg_E(j) \geq 2a\}.$$

We shall consider two cases.

Case 1: $|A| \geq 2a$. We shall show that in this case $\Delta^g = \top$ with high probability. First we estimate $|A \cap \mathrm{rng}(g)|$. Note that $\mathrm{rng}(g)$ is a random subset of R of size $n \Leftrightarrow n'$, thus also $R' = R \setminus \mathrm{rng}(g)$ is a random subset of size n'. Hence we can apply Lemma 9.2.4.

$$\begin{aligned} \mathrm{Prob}(|A \cap \mathrm{rng}(g)| < a) &= \mathrm{Prob}(|A \cap R'| \geq |A| \Leftrightarrow a) \\ &\leq \left(\frac{e|A|n^{1/3}}{n(|A| \Leftrightarrow a)} \right)^{|A|-a} \leq \left(\frac{2e}{n^{2/3}} \right)^{a}. \end{aligned} \tag{55}$$

The probability that Δ^g is not $\top$ is bounded by

$$\begin{aligned} &\mathrm{Prob}\left(\forall j \in A \cap \mathrm{rng}(g)((g^{-1}(j), j) \notin E)\right) \leq \\ &\quad \mathrm{Prob}(|A \cap \mathrm{rng}(g)| < a) + \\ &\quad \mathrm{Prob}\left(\forall j \in A \cap \mathrm{rng}(g)((g^{-1}(j), j) \notin E) \quad | \quad |A \cap \mathrm{rng}(g)| \geq a\right). \end{aligned} \tag{56}$$

The second term can be estimated by

$$\max_{C \subseteq A,\ |C| \geq a} \mathrm{Prob}\left(\forall j \in A \cap \mathrm{rng}(g)((g^{-1}(j), j) \notin E) \mid A \cap \mathrm{rng}(g) = C\right),$$

thus it suffices to consider a fixed such C and bound the probability. Let $C = \{j_1, j_2, \ldots, j_{|C|}\}$; think of the vertices $g^{-1}(j_1), g^{-1}(j_2), \ldots, g^{-1}(j_{|C|})$ as chosen one by one independently, except that they must be different.

$$\begin{aligned} &\mathrm{Prob}\left((g^{-1}(j_{t+1}), j_{t+1}) \notin E \quad | \quad g(i_1) = j_1, \ldots, g(i_t) = j_t\right) = \\ &= 1 \Leftrightarrow \frac{|E^{-1}(j_{t+1}) \Leftrightarrow \{i_1, \ldots, i_t\}|}{n+1 \Leftrightarrow t} \leq 1 \Leftrightarrow \frac{\deg_E(j_{t+1}) \Leftrightarrow t}{n+1} \leq 1 \Leftrightarrow \frac{2a \Leftrightarrow t}{n+1}. \end{aligned}$$

Thus the probability that $(g^{-1}(j_t), j_t) \notin E$ for all $t = 1, \ldots, |C|$ is

$$\leq \left(1 \Leftrightarrow \frac{2a}{n+1}\right)\left(1 \Leftrightarrow \frac{2a \Leftrightarrow 1}{n+1}\right) \ldots \left(1 \Leftrightarrow \frac{2a \Leftrightarrow |C|+1}{n+1}\right) \leq \left(1 \Leftrightarrow \frac{a}{n+1}\right)^a.$$

Since $\frac{a}{n+1} \sim \frac{1}{n^{1/3}}$ and $a \sim n^{2/3}$, this expression is $e^{-\Omega(n^{1/3})}$. The first term of (56) is estimated in (55) and is even smaller. Thus in Case 1 the probability is $1 \Leftrightarrow e^{-\Omega(n^{1/3})}$ as required.

Case 2: $|A| < 2a$. In this case we cover Δ^g by the set

$$(A \cap R') \cup (E^{-1}(R' \backslash A) \cap D').$$

We need only to estimate the probability that the size of the two sets in the union is small. We shall use Lemma 9.2.4 again.

$$\text{Prob}\left(|A \cap R'| > \frac{\ell}{2}\right) \leq \left(\frac{e 2a n^{1/3}}{n \cdot n^{1/3}/6}\right)^{n^{1/3}/6} = \left(\frac{12e}{40n^{1/3}}\right)^{n^{1/3}/6} = e^{-\Omega(n^{1/3})}. \tag{57}$$

To estimate the second set, first observe that

$$|E^{-1}(R' \backslash A)| \leq |R'| \cdot 2a = n^{1/3} \cdot 2\frac{n^{2/3}}{40} = \frac{n}{20}.$$

Thus

$$\begin{aligned} \text{Prob}(|E^{-1}(R' \backslash A) \cap D'| > \frac{\ell}{2}) &\leq \left(\frac{e \cdot \frac{n}{20} \cdot (n^{1/3}+1)}{(n+1)n^{1/3}/6}\right)^{n^{1/3}/6} \\ &= \left(\frac{3e}{10} \cdot \frac{n(n^{1/3}+1)}{(n+1)n^{1/3}}\right)^{n^{1/3}/6} = e^{-\Omega(n^{1/3})}, \end{aligned} \tag{58}$$

since the term in the parentheses converges to $\frac{3e}{10} < 1$. By (57) and (58) we get the required bound in Case 2. □

Now we are ready to prove the lower bound which was originally proved by Haken [1985] with a better exponent than we give here.

9.2.5. Theorem. (Haken [1985]) *Every resolution proof of PHP_n has size at least $2^{\varepsilon n^{1/3}}$, where $\varepsilon > 0$ is a constant.*

Proof. Suppose a proof of size $< 2^{\varepsilon n^{1/3}}$ is given. Take a random $g \in M_{n'}, n' = \lfloor n^{1/3} \rfloor$. Then, by Lemma 9.2.2, for every formula Δ of the proof the probability that Δ^g is not covered by a set of size $< \frac{n'}{3}$ is at most $2^{\varepsilon n^{1/3}}$. Thus we have positive probability that, for some $g \in M_{n,n'}$, all formulas are covered by sets $< \frac{n'}{3}$. Hence there is at least one such a g.

Consider the proof restricted using such a g; it is a derivation of $\perp$ from clauses of $PHP_{n'}$. Choose a covering set of size $< \frac{n'}{3}$ for each clause in this proof. Then take

boolean algebras $P(V_T)$ for clauses and for each application of the rule as described above. As we have observed, the clauses of $PHP_{n'}$ get value 1 in their boolean algebras. Now we can apply Lemma 9.1.1. The conclusion should be that $\bot$ gets also 1. But $\bot$ gets the value 0 by the definition of the boolean algebras.

Hence the proof must have size $\geq 2^{\varepsilon n^{1/3}}$. □

9.3. Lower bounds based on effective interpolation theorems. We are going to discuss an approach which is not based on such *ad hoc* proofs, but instead it uses some general theorems interesting in their own right. These theorems are versions of the *interpolation theorem,* a classical result of Craig [1957a,1957b], see Chapter I. The interpolation theorem has a first order logic version and a propositional version. Recently some strengthenings of the propositional interpolation theorem have been successfully applied to prove lower bounds on the length of propositional proofs.

The propositional interpolation theorem states that for a given propositional tautology $\Phi(\bar{p},\bar{q}) \to \Psi(\bar{p},\bar{r})$, where $\bar{p},\bar{q},\bar{r}$ are disjoint strings of propositional variables, there exists a formula $I(\bar{p})$, which contains only the common variables $\bar{p}$, such that both $\Phi(\bar{p},\bar{q}) \to I(\bar{p})$ and $I(\bar{p}) \to \Psi(\bar{p},\bar{r})$ are also tautologies. Such a formula $I(\bar{p})$ is called an *interpolant* of $\Phi(\bar{p},\bar{q}) \to \Psi(\bar{p},\bar{r})$. The proof of this statement is trivial: Take the quantified boolean formula $\exists\bar{x}\ \Phi(\bar{p},\bar{x})$ (or $\forall\bar{x}\ \Psi(\bar{p},\bar{x})$); clearly, it interpolates $\Phi(\bar{p},\bar{q}) \to \Psi(\bar{p},\bar{r})$. As any boolean function can be defined by an ordinary propositional formula, there is a propositional formula $I(\bar{p})$ equivalent to $\exists\bar{x}\ \Phi(\bar{p},\bar{x})$.

Craig gave constructive proofs of his theorems, i.e., he showed how to construct an interpolant $I(\bar{p})$ from a proof d of $\Phi(\bar{p},\bar{q}) \to \Psi(\bar{p},\bar{r})$. Thus the complexity of $I(\bar{p})$ depends on the complexity of the proof d. This led Krajíček [1994a] to propose a method of lower bounds proofs whose idea can be stated as follows: suppose we can show that $\Phi(\bar{p},\bar{q}) \to \Psi(\bar{p},\bar{r})$ does not have a simple interpolant, then it cannot have a simple proof.

Another relationship of interpolation theorems to questions in complexity theory had earlier been considered by Mundici [1984], but he did not consider the lengths of proofs.

9.3.1. The original proof of Craig was based on cut-elimination, so the constructed interpolant can be exponentially large. His proof can be used to get a good bound on interpolants for cut-free sequent propositional proofs, but we have to consider a different measure of the complexity of interpolants. The new idea is that we can look at an interpolant as a boolean function and then we can apply any of the measures of complexity of boolean functions. Here the right measure is the size of the smallest *circuit* computing the boolean function.

9.3.2. Theorem. (Krajíček [1997a]) *Let d be a cut-free proof with k lines of a sequent*

$$\Phi_1(\bar{p},\bar{q}),\dots,\Phi_m(\bar{p},\bar{q}) \Leftrightarrow\!\!\to \Psi_1(\bar{p},\bar{r}),\dots,\Psi_l(\bar{p},\bar{r})$$

where $\bar{p}, \bar{q}, \bar{r}$ are disjoint sets of propositional variables (i.e. no $\bar{q}$ occurs in the consequent and no $\bar{r}$ occurs in the antecedent). Then it is possible to construct a an interpolant $I(\bar{p})$ of $\bigwedge_i \Phi_i(\bar{p}, \bar{q}) \to \bigvee_j \Psi_j(\bar{p}, \bar{r})$ which is a boolean circuit of size $k^{O(1)}$.

The proof is essentially the original one of Craig [1957a,1957b]. The idea is to construct interpolants for each sequent in the proof successively starting with the initial sequents and going down to the end sequent. As the proof is cut-free, each sequent contains only formulas containing either only variables $\bar{p}, \bar{q}$, or only variables $\bar{p}, \bar{r}$, so it makes sense to talk about an interpolant for it. □

The reason for using circuit size is because we consider proofs in the sequence form. For *tree-like proofs* we actually get a polynomial size *formula* as an interpolant.

9.3.3. Suppose we have an interpolant $I(\bar{p})$ for $\Phi(\bar{p}, \bar{q}) \to \Psi(\bar{p}, \bar{r})$ i.e., the implications $\Phi(\bar{p}, \bar{q}) \to I(\bar{p})$ and $I(\bar{p}) \to \Psi(\bar{p}, \bar{r})$ are true. Let a truth assignment $\bar{a}$ to the variables $\bar{p}$ be given. Then either $\neg\Phi(\bar{p}, \bar{a})$ or $\Psi(\bar{a}, \bar{r})$ is true. The interpolant can be used to decide which of the two possibilities holds, namely, if $I(\bar{a})$ is true, then $\Psi(\bar{a}, \bar{r})$ is true, otherwise $\neg\Phi(\bar{a}, \bar{r})$ is true. (It is possible that both $\neg\Phi(\bar{p}, \bar{a})$ and $\Psi(\bar{a}, \bar{r})$ are true, in which case $I(\bar{a})$ could be true or false.) Thus there is an alternative way of looking at interpolant: Let $\alpha(\bar{p}, \bar{q}) \vee \beta(\bar{p}, \bar{r})$ be a valid disjunction; an interpolant is a procedure which produces one of the two disjuncts which becomes a tautology after assigning given truth values to $\bar{p}$.

We can look at the interpolation theorem even more abstractly (see Razborov [1994]). Let A and B be disjoint $\mathcal{NP}$ sets. Then we can define the set of input strings $\bar{a}$ of length n which are not in A, resp. not in B, by a polynomial size formula $\alpha_n(\bar{p}, \bar{q})$, resp. $\beta_n(\bar{p}, \bar{r})$ ($\bar{a}$ is not in A if $\alpha_n(\bar{a}, \bar{q})$ is a tautology, similarly for B, see next section). Since A and B are disjoint i.e., the complements cover all inputs, the disjunction $\alpha_n(\bar{p}, \bar{q}) \vee \beta_n(\bar{p}, \bar{r})$ is a tautology. If we have a polynomial time computable set C which separates A from B i.e., $A \subseteq C$, $C \cap B = \emptyset$, then we have a polynomial time decision algorithm for finding a true disjunct from $\alpha_n(\bar{a}, \bar{q}) \vee \beta_n(\bar{a}, \bar{r})$. Cook's theorem implies that then there exists also a polynomial size circuit $C_n(\bar{p})$ for this problem. Clearly, $C_n(\bar{p})$ is an interpolant for $\alpha_n(\bar{p}, \bar{q}) \vee \beta_n(\bar{p}, \bar{r})$.

9.3.4. The most interesting application of the effective interpolation is in the case of resolution.

9.3.5. Theorem. (Krajíček [1997a]) *Let d be a resolution proof of the empty clause from clauses $A_i(\bar{p}, \bar{q}), i \in I$, $B_j(\bar{p}, \bar{r}), j \in J$ where $\bar{p}, \bar{q}, \bar{r}$ are disjoint sets of propositional variables. Then it is possible to construct a circuit $C(\bar{p})$ such that for every 0-1 assignment $\bar{a}$ for $\bar{p}$*

$$C(\bar{a}) = 0 \quad \Rightarrow \quad A_i(\bar{a}, \bar{q}), i \in I \text{ are unsatisfiable, and}$$

$$C(\bar{a}) = 1 \quad \Rightarrow \quad B_j(\bar{a}, \bar{r}), j \in J \text{ are unsatisfiable;}$$

the size of the circuit C is bounded by $O(|d|)$.

Moreover, one can construct a resolution proof of the empty clause from clauses $A_i(\bar{a},\bar{q}), i \in I$ *if* $C(\bar{a}) = 0$*, respectively from* $B_j(\bar{a},\bar{r}), j \in J$ *if* $C(\bar{a}) = 1$*, whose size is at most the size of* d*.*

We shall sketch two proofs of this theorem. The idea of the first one, due to Krajíček [1997a], is to reduce it to Theorem 9.3.2. This looks strange, as resolution proofs consist only of cuts and we know that cut-elimination does not work. The trick is to eliminate cuts by replacing them by conjunctions.

For each initial clause $s_1 \vee \ldots \vee s_k$, where s_i are literals, first prove the sequent $\leftrightarrow\!\!\rightarrow \bigwedge_{i=1}^k \overline{s_i}, s_1, \ldots, s_k$; we denote by $\overline{s_i}$ the literal complementary to s_i. Our goal is to derive a sequent consisting only of such conjunctions obtained from initial clauses $\leftrightarrow\!\!\rightarrow \ldots, \bigwedge_{i=1}^k \overline{s_i}, \ldots$. Thus we want to replace the refutation proof by a proof of the corresponding tautology (DNF). We shall not quite succeed, we have to add also conjunctions of the form $s_i \wedge \overline{s_i}$, which are, however, false, hence do not influence interpolants at all.

In transforming the resolution proof into a cut-free sequent proof we follow the given resolution proof, but instead of applying cut with some cut literal s_i, we introduce $s_i \wedge \overline{s_i}$. Thus a general sequent in the proof will consist of conjunctions of negated literals of $A_i(\bar{p},\bar{q})$'s and $B_j(\bar{p},\bar{r})$'s, conjunctions of complementary literals $s_i \wedge \overline{s_i}$ and single literals. The single literals of the sequent are just the literals of the corresponding clause in the resolution proof. In the last sequent, as in the resolution proof, the single literals will be eliminated and we are left only with conjunctions of negated literals of the initial clauses $A_i(\bar{p},\bar{q})$'s and $B_j(\bar{p},\bar{r})$'s and conjunctions of complementary literals. Since conjunctions of complementary literals are false, an interpolant for this sequent is also an interpolant for the sequent without them. So we can apply Theorem 9.3.2 to get an interpolant for this sequent which is an interpolant for $A_i(\bar{p},\bar{q}), i \in I$, $B_j(\bar{p},\bar{r}), j \in J$ in the sense of the theorem. □

The idea of the second proof (Pudlák [1997]) is to construct a refutation proof either from $A_i(\bar{a},\bar{q}), i \in I$, or from $B_j(\bar{a},\bar{r}), j \in J$ for every given truth assignment. If there is a polynomial time algorithm for constructing such a proof, then there is one also for deciding which of the two sets is unsatisfiable, hence also a polynomial size circuit. Substitute the truth assignment $\bar{a}$ into the initial clauses and discard those which contain a literal which is true under the truth assignment $\bar{a}$ and delete the $\bot$ produced by the substitution from the others. Then we follow the proof. What we want is to never mix variables $\bar{q}$ with variables $\bar{r}$. So when we should resolve along a variable q_i or r_i we do it, since this will not produce a mixed clause. However, if we should resolve along some p_i, we must do something else. Now we simply take the clause which corresponds to an original clause where the literal p_i, resp. $\overline{p_i}$, is false under the truth assignment $\bar{a}$. This clause will be a subclause of the next original clause, hence we can continue and eventually obtain an empty clause. Since variables $\bar{q}$ and $\bar{r}$ are never mixed, the new proof will split into at least two disconnected parts. We can backtrack which initial clauses are actually needed to get the empty clause

(they must be of the same kind) and take only that component as the new proof. □

A closer analysis of this proof shows that we can use the directed graph of the proof as the graph for the circuit, provided we take suitable connectives. So the relation between the proof and the circuit is very close.

One can also easily show that we have to use circuits instead of formulas, unless formulas are as powerful as circuits (which most researcher doubt) Krajíček [1994a]. Namely, for every circuit we can write a tautology stating that the computation is unique. We use variables $\bar{p}$ for the input values of the circuit, variables $\bar{q}$ for the values at the gates in the first computation and variables $\bar{r}$ for the values of at the gates in the second computation. The tautology asserts that if the output value, say q_k, in the first computation is 1, then the output value r_k in the second computation is also 1. Clearly, any interpolant of this tautology computes the same function as the circuit. On the other hand, the tautology has a resolution proof of linear size.

9.3.6. In order to apply Theorem 9.3.5 we need to have good lower bounds on the size of circuits computing some explicitly defined boolean functions. Presently all the known lower bounds for explicitly defined functions are only linear. Fortunately there is a version of the theorem which can be combined with currently known lower bounds. Quite surprisingly a very mild condition on the clauses implies that the interpolating circuits can be constructed monotone. A *monotone boolean circuit* is a circuit in the basis $\{\wedge, \vee, 0, 1\}$, i.e., a circuit whose gates are monotone boolean functions.

9.3.7. Theorem. (Krajíček [1997a]) *Assume that clauses as in Theorem 9.3.5 are given. Suppose moreover that either all variables $\bar{p}$ occur in $A_i(\bar{p}, \bar{q}), i \in I$ only positively or all variables $\bar{p}$ occur in $B_j(\bar{p}, \bar{r}), j \in J$ only negatively, then there exists a circuit C satisfying the conclusion of Theorem 9.3.5 which is moreover monotone.*

The proof of this theorem is obtained by inspection of either of the proofs of Theorem 9.3.5. □

There are well-known exponential lower bounds for the monotone circuit complexity of explicit boolean functions. This alone would not suffice to get an exponential lower bound on resolution proofs. By another lucky coincidence the lower bounds on monotone circuits give more: they actually show that some pairs of disjoint $\mathcal{NP}$ sets cannot be separated by monotone circuits.

In particular such a lower bound can be derived for tautologies related to the clique problem. Let $Clique_{n,k}(\bar{p}, \bar{q})$ denote a set of clauses expressing that the graph with n vertices coded by $\bar{p}$ has a clique of size at least k coded by $\bar{q}$. The variables $\bar{p}$ represent edges of the graph and the variables $\bar{q}$ represent the graph of a one-to-one function from a k-element set into the set of vertices of the graph. Formally, we take

variables $p_{i,j}$, $1 \le i < j \le n$, $q_{i,r}$, $1 \le i \le n$, $1 \le r \le k$ and clauses

$$\begin{array}{ll} \bigvee_i q_{i,r} & \text{for all } 1 \le r \le k; \\ \neg q_{i,r} \vee \neg q_{i,r'} & \text{for all } 1 \le i \le n,\ 1 \le r < r' \le k; \\ \neg q_{i,r} \vee \neg q_{i',r'} \vee p_{i,i'} & \text{for all } 1 \le i < i' \le n,\ 1 \le r < r' \le k. \end{array}$$

Let $Color_{n,l}(\bar{p}, \bar{r})$ denote a set of clauses expressing that the graph with n vertices coded by $\bar{p}$ is l-colorable. The variables $\bar{r}$ code a mapping from the set of vertices of the graph into a set of size l such that no edge is mapped on a single point. This can be expressed by a similar set of clauses as above.

If $k > l$, the set of graphs containing a k-clique is disjoint with the set of l-colorable graphs (a clique needs at least k colors), hence the two sets of clauses $Clique_{n,k}(\bar{p}, \bar{q})$ and $Color_{n,l}(\bar{p}, \bar{r})$ cannot be satisfied simultaneously. For suitable parameters it has been shown that these sets of graphs cannot be separated by small monotone circuits.

9.3.8. Theorem. (Razborov [1985], Alon and Boppana [1987]) *Let $l < k$ and $\sqrt{kl} \le \frac{n}{8 \log n}$. Then every monotone circuit which outputs 1 on graphs with a k-clique and 0 on l-colorable graphs has size $2^{\Omega(\sqrt{l})}$.* □

9.3.9. Corollary. (Krajíček [1997a]) *Any resolution refutation of the set of clauses $Clique_{n,k}(\bar{p}, \bar{q}) \cup Color_{n,l}(\bar{p}, \bar{r})$ has size $2^{\Omega(\sqrt{l})}$.* □

Using this approach we do not avoid combinatorial technicalities, since the proof of Theorem 9.3.8 is nontrivial. Its advantage is that an exponential lower bound on the length of resolution proofs is easily accessible to those who already know lower bounds on the size of monotone circuits.

9.3.10. Another advantage of this approach is that it can be applied to cutting plane proofs, where the random restriction method does not seem to work.

The version of Theorem 9.3.5 for cutting plane proofs is almost identical. There are two versions of the monotone case, Theorem 9.3.7, for cutting plane proofs. The first one (Bonet, Pitassi and Raz [1997a], Krajíček [1997a]) gives monotone boolean circuits, but requires that the coefficients in the proof are polynomially bounded by its size (put otherwise, the size of the monotone circuit is bounded not only by the number of lines but also by the size of the coefficients). The second version (Pudlák [1997]) works without any restriction on the coefficients, but the interpolating circuit is not an ordinary monotone boolean circuit. We have to consider circuits which are monotone and compute with *arbitrary real numbers*. Again fortunately, the known proofs of the lower bounds for monotone boolean circuits can be easily extended to the more general model (Pudlák [1997], Haken and Cook [n.d.]). In particular, an exponential lower bound can be proved for the clauses $Clique_{n,k}(\bar{p}, \bar{q}) \cup Color_{n,l}(\bar{p}, \bar{r})$ presented as inequalities in the cutting plane proof system.

9.3.11. At first this approach to lower bounds looked very promising. Unfortunately, it became clear very soon that it cannot be extended much further beyond resolution. We do not know, if an effective interpolation theorem in the style of Theorem 9.3.5 holds for bounded depth Frege systems and we rather think it does not hold even for such weak proof systems (cf. Krajíček [1997a] for some arguments). For Frege systems we have strong evidence that it does not hold. Namely one can prove that such a theorem does not hold for Frege systems using the widely accepted conjecture that factoring of integers is not in polynomial time.

9.3.12. Theorem. (Bonet, Pitassi and Raz [1997b]) *There exists a sequence of tautologies of the form* $\alpha_n(\bar{p},\bar{q}) \vee \beta_n(\bar{p},\bar{r})$ *which have polynomial size Frege proofs, but for which there is no sequence of polynomial size interpolation circuits, provided that factoring of integers is not in polynomial time.* □

Instead of proving this theorem we shall explain in general terms the rather surprising connection between propositional calculus and cryptography. The basic concept of cryptography is the *one-way function*, which is, roughly speaking, a function which can be easily computed (in polynomial time) but whose inverse function is hard.[5] It is not known if such functions exist; in fact, we even do not know how to prove their existence assuming $\mathcal{P} \neq \mathcal{NP}$. We do know, however, that a one-way function exists iff there exist disjoint $\mathcal{NP}$ sets which cannot be separated by a set in $\mathcal{P}$. We shall see in the next section (see 10.3) that arithmetical theorems of certain logical complexity can be translated into a sequence of propositional tautologies. Furthermore for each first order theory we can construct a propositional proof system where the translations of such theorems have polynomial size proofs. Now, if we have a pair of disjoint $\mathcal{NP}$ sets A, B which cannot be separated by a set in $\mathcal{P}$, we can take a theory T in which this fact is provable (just include this statement as an axiom). Hence in the propositional proof system P derived from T we can prove the tautologies derived from A, B. On the other hand, polynomial time interpolation for P would give us a separating set for A, B as noted above. (For sake of simplicity we are talking about polynomial time algorithms instead of polynomial size circuits; the distinction between the two concepts is not essential for our argument.)

A weaker version of Theorem 9.3.12, which gave the result only for extension Frege systems, was originally proved by taking the conjectured one-way function $x \mapsto g^x \bmod n$ and proving in S^1_2, which is a theory associated with extension Frege proof systems, that the corresponding pair of $\mathcal{NP}$ sets is disjoint (see Krajíček and Pudlák [1998] for a full proof).

9.4. Other lower bounds. The method of random restrictions (exemplified in section 9.2) has been extended by Ajtai [1994a] to any fixed depth Frege system. It gives, however, only slightly superpolynomial lower bounds. In order to get

[5] For practical cryptography one needs *hard in the average;* here we consider only the worst case complexity.

exponential lower bounds one needs a more substantial change in which the concept of covering sets is replaced by certain decision trees and a Switching Lemma, of the type used by Yao [1985] and Håstad [1986], is applied to reduce the depth of formulas; see Beame et al. [1992], Krajíček, Pudlák and Woods [1995] and Pitassi, Beame and Impagliazzo [1993].

Let us define at least the concept of the *decision tree* which is used in these bounds for PHP_n. We use the same notation as above. Such a tree is a labelled rooted tree, where the vertices are labelled by elements of $D \cup R$, except for the leaves, which are labelled by 0 – reject, and 1 – accept; the edges are labelled by pairs (i,j), $i \in D$, $j \in R$. We require that for a nonleaf vertex v with a label $i \in D$, resp. $i \in R$, the outgoing edges are labelled by (i,j), resp (j,i), one edge for every j which does not occur on the path leading to v. Consequently, the edge labels on every branch are independent, i.e., they form a partial one-to-one mapping.

In the lower bound proof we assign to each formula the boolean algebra of all subsets of leaves of such a tree and the value of the formula $\lambda(\varphi)$ is the subset of leaves labelled by 1.

The intuitive meaning of this concept is the following. We think of truth values of the propositional variables $p_{i,j}$ as given by some imaginary one-to-one mappings from D onto R. In fact, in a nonstandard model with n infinite, there are such external mappings. The decision tree enables us to decide in a natural way if such a mapping is accepted or not. Then all the boolean algebras defined by trees can be embedded into a single one which is the boolean algebra of subsets of one-to-one mappings from D onto R. Put otherwise our logic is a logic of one-to-one mappings from D onto R.

PHP_n is not the only sequence for which one can prove exponential lower bounds on bounded depth Frege proofs. Another such sequence is PAR_n — the parity principle — where PAR_n, n odd, expresses that a set of cardinality n cannot be partitioned into pairs. Similarly, one can consider the counting principle $COUNT_{p,n}$ which expresses that a set of size n, n not divisible by p, cannot be partitioned into blocks of size p. Ajtai [1990] has shown that PAR_n does not have polynomial size bounded depth proofs, even if we use instances of PHP_m as premises, and similar independence results have been proved for the counting principles by Ajtai [1994b], Beame et al. [1996], Buss et al. [1996/1997].

Together with exponential lower bounds for cutting plane proof systems and degree lower bounds for the polynomial calculus, these are the strongest results so far. For unrestricted Frege system we have only an $\Omega(n^2)$ lower bound for tautologies such as $\neg^{2n}(p \vee \neg p)$. The proof is based on the claim that all subformulas of this tautology must occur essentially (i.e., in a constant depth) in the proof. This is essentially the same idea as in Claim 4.2.3, see also 8.1.16. Apart from this rather simple proof we do not have anything for Frege and stronger systems.

10. Bounded arithmetic and propositional logic

In this section we shall show an important relation between the lengths of proofs of propositional tautologies and provability in fragments of arithmetic. By this connection certain arithmetical formulas can be translated to a sequence of propositions, and if the formula is provable in some theory, then the propositions have small (e.g., polynomial size) proofs in some propositional proof system associated with the theory. Surprisingly, there are pairs of such a theory T and a propositional proof system P where both the theory T and the proof system T are quite natural. In this situation we can think of T and P to be just two facets of a single concept, where T is a *uniform* version of the *nonuniform* P. This is just another parallel to boolean circuit complexity, where the uniform model is the Turing machine and the nonuniform model is a sequence of boolean circuits.

The main application of this relation is in showing independence results. If we could prove superpolynomial lower bounds on strong propositional proof systems, then we could show interesting independence results in bounded arithmetic such as unprovability of $\mathcal{NP} = co\mathcal{NP}$.

There is also practical use of this relation which is necessary to take into account even if you are not interested in first order theories. It might be fairly difficult to find and describe short proofs of some tautologies directly, while in a bounded arithmetic we can often see easily that the corresponding first order formula is provable. This was used, e.g. in Pudlák [1991], to disprove a conjecture saying that formulas expressing Ramsey's theorem in propositional logic do not have polynomial size proofs in Frege systems. Similarly, it is possible to prove the existence of a polynomial simulation of a proof system P by a proof system Q by proving the reflection principle (see below) for P in a theory associated with Q. In such a way the polynomial simulation of substitution Frege by extension Frege system was discovered by Dowd [1985] and Krajíček and Pudlák [1989].

This subject requires some familiarity with fragments of arithmetic considered in bounded arithmetic. The reader, who does not know that subject should consult Chapter II Buss [1986], Hájek and Pudlák [1993] or Krajíček [1995].

10.1. There are basically two translations of bounded formulas into propositions. They are determined by the particular way in which we represent truth assignments. A truth assignment is a finite sequence $\bar{a}$ of 0's and 1's. We can code it either by a subset of a finite segment of integers or by a number whose binary representation is $1\bar{a}$. We start with the simpler one.

10.2. First translation. Let $L_0(\alpha)$ be the language of arithmetic with nonlogical symbols $\underline{0}, S, +, \cdot, \leq$ augmented with a second order variable α for l-ary relations. We consider the class $\Delta_0(\alpha)$ of bounded formulas in the language $L_0(\alpha)$. Assume that we use the same connectives in the first order language and the propositional calculus and they include $\wedge, \vee$; moreover we shall assume that we have propositional constants $\bot, \top$ in propositional logic and that the propositional variables are indexed

by l-tuples of nonnegative integers.

Let $\theta \in \Delta_0(\alpha)$ be a formula with k free variables. Then for each sequence $n_1, \ldots, n_k$ of nonnegative integers we define *a propositional formula*

$$\langle \theta \rangle_{n_1,\ldots,n_k}$$

inductively as follows.

1. for terms $s(n_1, \ldots, n_k), t(n_1, \ldots, n_k)$, we define

$$\langle s(n_1, \ldots, n_k) = t(n_1, \ldots, n_k) \rangle_{n_1,\ldots,n_k} =_{df} \bot \text{ (resp. } = \top),$$

   if $s(n_1, \ldots, n_k) = t(n_1, \ldots, n_k)$ is false, (resp. true); we use the same definition for $\leq$ in place of $=$;

2. for terms $t_1(x_1, \ldots, x_k), \ldots, t_l(x_1, \ldots, x_k)$, we define

$$\langle \alpha(t_1(x_1, \ldots, x_k), \ldots, t_l(x_1, \ldots, x_k)) \rangle_{n_1,\ldots,n_k} =_{df} p_{i_1,\ldots,i_l},$$

   where $i_1, \ldots, i_l$ are the values of $t_1(n_1, \ldots, n_k), \ldots, t_l(n_1, \ldots, n_k)$;

3. propositional connectives are translated identically, e.g.,

$$\langle \theta_1 \wedge \theta_2 \rangle_{n_1,\ldots,n_k} =_{df} \langle \theta_1 \rangle_{n_1,\ldots,n_k} \wedge \langle \theta_2 \rangle_{n_1,\ldots,n_k};$$

4. bounded quantifiers are translated to long disjunctions and conjunctions, thus

$$\langle \exists y \leq s(x_1, \ldots, x_k)\, \theta(x_1, \ldots, x_k, y) \rangle_{n_1,\ldots,n_k} =_{df}$$

$$\langle \theta(x_1, \ldots, x_k, y) \rangle_{n_1,\ldots,n_k,0} \vee \cdots \vee \langle \theta(x_1, \ldots, x_k, y) \rangle_{n_1,\ldots,n_k,m},$$

   where m is the value of $s(n_1, \ldots, n_k)$; in the case of bounded universal quantifier the propositional formula is defined dually.

10.2.1. Example. Let $\theta(x)$ be the formula expressing the pigeonhole principle for the binary relation α (for sake of simplicity we use a little stronger form than in section 9):

$$\exists u \leq S(x) \forall v \leq x(\neg\alpha(u, v)) \vee \exists u_1, u_2 \leq S(x) \exists v \leq x(u_1 \neq u_2 \wedge \alpha(u_1, v) \wedge \alpha(u_2, v)).$$

For a given n, the translation $\langle \theta(x) \rangle_n$ has form:

$$\bigvee_{i \leq n+1} \bigwedge_{j \leq n} \neg p_{ij} \;\vee \bigvee_{i_1, i_2 \leq n+1} \bigvee_{j \leq n} \bar{\delta}_{i_1,i_2} \wedge p_{i_1,j} \wedge p_{i_2,j},$$

where $\bar{\delta}_{i_1,i_2}$ denotes $\bot$ if $i_1 = i_2$ and denotes $\top$ otherwise. The constants $\top$ and $\bot$ can be easily eliminated; namely, the formula is equivalent, using a polynomial size bounded depth Frege proof, to

$$\bigvee_{i \leq n+1} \bigwedge_{j \leq n} \neg p_{ij} \;\vee \bigvee_{i_1, i_2 \leq n+1, i_1 \neq i_2} \bigvee_{j \leq n} p_{i_1,j} \wedge p_{i_2,j}.$$

We have obtained the usual form of the propositional formula expressing the pigeonhole principle.

Let us observe, which is quite clear from the example, that the translation is a formula of polynomial size in the indices $n_1, \ldots, n_k$ and, moreover, the depth is bounded by a constant, namely by the depth of the first order formula.

Let $I\Delta_0(\alpha)$ denote $I\Delta_0$ with the induction schema extended to all $\Delta_0(\alpha)$ formulas.

10.2.2. Theorem. (implicit in Paris and Wilkie [1985]) *If $I\Delta_0(\alpha)$ proves $\forall x_1 \ldots \forall x_k \theta(x_1, \ldots, x_k)$, where $\theta(x_1, \ldots, x_k) \in \Delta_0(\alpha)$, then there exists a polynomial p and a constant d such that the propositions $\langle \theta(x_1, \ldots, x_k) \rangle_{n_1,\ldots,n_k}$ have Frege proofs of size $\leq p(n_1, \ldots, n_k)$ and depth $\leq d$.*

Proof-sketch. Suppose $\forall x_1 \ldots \forall x_k \theta(x_1, \ldots, x_k)$ is provable in $I\Delta_0(\alpha)$, let $n_1, \ldots, n_k$ be given. By cut elimination in the sequent calculus formalization of $I\Delta_0(\alpha)$, we have a free-cut-free proof of this sentence. From this proof we get a proof of the sequent $\rightarrow \theta(a_1, \ldots, a_k)$ which contains only $\Delta_0(\alpha)$ formulas. Starting at the bottom, i.e., with $\rightarrow \theta(a_1, \ldots, a_k)$, we shall gradually translate the first order proof into a propositional proof. The structural and propositional rules are, of course, translated identically.

Consider an instance of the induction rule

$$\frac{A(b), , \ \rightarrow \Delta, A(S(b))}{A(\underline{0}), , \ \rightarrow \Delta, A(t)},$$

where we have already translated the part of the proof from the lower sequent on. Suppose that in the course of translation we have assigned numbers $m_1, \ldots, m_r$ to the free variables of the lower sequent. Observe that $\langle A(b_1, \ldots, b_r, t(b_1, \ldots, b_r)) \rangle_{m_1,\ldots,m_r}$ is equal to $\langle A(b_1, \ldots, b_r, b_{r+1}) \rangle_{m_1,\ldots,m_r,m_{r+1}}$, where m_{r+1} is the value of $t(m_1, \ldots, m_r)$. We take m_r translations of the upper sequent with indices $m_1, \ldots, m_r, m$, $m = 0, \ldots, m_r \Leftrightarrow 1$ (m stands for the free variable b). The translation of the lower sequent follows from them by applying $r \Leftrightarrow 1$ cuts.

The quantifier rules for bounded quantifiers are treated similarly. Eventually we reach initial sequents which are translated to initial sequents in propositional logic. □

This theorem can be used, as mentioned above, to construct short bounded depth Frege proofs, but, what is more interesting, also to prove, for instance that the pigeonhole principle for a free second order variable α is not provable in $I\Delta_0(\alpha)$. The first proof of this independence, by Ajtai [1994a], was based on model theory and a lower bound on the length of proofs of the propositional pigeonhole principle was derived as a corollary. Nowadays it is clear that the right and simpler way is to prove the lower bound for propositional logic first, see Beame et al. [1992], Pitassi, Beame and Impagliazzo [1993] and Krajíček, Pudlák and Woods [1995].

Let us mention by passing another parallel with computational complexity. The results for theories augmented with an extra free second order variable are alike to

the oracle results in complexity theory. The "absolute" results, e.g. unprovability of the pigeonhole principle for Δ_0-formulas in $I\Delta_0$, are beyond present means, as well as unrelativized separation results in computational complexity theory.

10.2.3. The same translation can be applied to second order theories where we have also true second order axioms. For U^1_1 we get a bound $2^{(\log n)^{O(1)}}$ on the size of Frege proofs (with a $(\log n)^{O(1)}$ bound on the depth); for V^1_1, Krajíček [1994b] gives a polynomial bound on the size of extension Frege proofs.

10.3. Second translation. For the second translation we consider the language L_2 of the theories S_2 and T_2 introduced by Buss [1986], see Chapter II. This language extends L_0 by $\lfloor x/2 \rfloor, x\#y, |x|$. The interpretation of these function symbols is $|x| = \lceil \log_2(x+1) \rceil$, $x\#y = 2^{|x|\cdot|y|}$. The $\#$ function is used to obtain faster growth rate of terms, namely $2^{p(\log x)}$, p a polynomial. This means that the *lengths* of the numbers increase polynomially, which renders formalization of polynomial time computations possible. The $|x|$ function is used to define *sharply bounded* quantifiers

$$\forall x \le |t|, \qquad \exists x \le |t|,$$

where t is a term. The basic property is that there are only polynomially many elements x less than or equal than $|t|$, since the outermost function in this term is, essentially, the logarithm.

The class Π^b_1 consists of formulas of L_2 which contain only sharply bounded quantifiers and strong bounded quantifiers (positive occurrences of universal bounded and negative occurrences of existential bounded); the other classes Π^b_i, Σ^b_i are defined similarly.

We want to define propositional translations of a Π^b_1 formula $\varphi(x_1, \dots, x_k)$. The translation will be denoted by $[\varphi(x_1, \dots, x_k)]_{n_1,\dots,n_k}$. Now we index the translation with strings of integers again, but the meaning is that we express propositionally that the sentence $\varphi(x_1, \dots, x_k)$ holds for all $x_1, \dots, x_k$ with $|x_1| \le n_1, \dots, |x_k| \le n_k$. The intuition behind the translation is the following. We identify truth assignments with (binary representations of) numbers. Since the terms are polynomial time computable functions, we can express atomic first order formulas by polynomial size propositions. Sharply bounded quantifiers are translated to polynomial size disjunctions and conjunctions. The strong bounded quantifiers are represented by sequences of propositional variables; this is a correct interpretation, since, by definition, a propositional tautology must be satisfied *for all* truth assignments.

A formal definition is fairly involved, thus most authors do not give a full definition, and we shall also only sketch how to resolve some technical problems of the definition.

First consider an atomic formula, say, with only one free variable, $s(x) = t(x)$. Let n be given for which we want to express propositionally that the sentence $s(x) = t(x)$ holds for all x with $|x| \le n$. Ideally we would take propositional variables $\bar{p} = (p_1, \dots, p_n)$ representing such numbers and formulas $\sigma_i(\bar{p}), \tau_i(\bar{p})$, $i = 1, \dots, m$,

($m = n^{O(1)}$), representing the bits of $s(x)$ resp. $t(x)$, and define

$$[s(x) = t(x)]_n \ =_{df} \bigwedge_{i=1,\ldots,m} \sigma_i(\bar{p}) \equiv \tau_i(\bar{p}).$$

There are such formulas of polynomial size for each of the basic functions, hence by composing them we get polynomial size formulas for all terms. Probably one can use these formulas, but it would require to find short extension Frege proofs of basic properties of these functions, which is by no means obvious for such a formalization. Therefore, instead of it, we take the natural circuits for the functions and introduce propositional variables for the functions computed at the vertices of the circuits. Then the translation will be an implication with the antecedent being the conjunction of simple clauses relating the values of the vertices of the circuits and consequent being

$$\bigwedge_{i=1,\ldots,n} q_i \equiv r_i,$$

where q_i and r_i are the propositional variables for the outputs of the circuits for $s(x)$ and $t(x)$ respectively. Thus the translation will have a polynomial number of extra variables which do not code bits of the numbers representing the free variables of the first order formula. For such a formalization it is much easier to prove the basic properties of the translation

As explained above, the strong bounded quantifiers are simply omitted, (except that the bounds on the variables are left as a part of the formula) and the sharply bounded quantifiers are translated using disjunctions and conjunctions. Consider for instance a formula Φ starting with a sharply bounded quantifier followed by a universal bounded quantifier, say

$$\exists y \leq |t(x)| \forall z \leq s(x,y)\ \varphi(x,y,z),$$

where φ is an open formula. We want to define the translation $[\Phi]_n$. We first replace the quantified variable y by the numerical instances, and then translate

$$\bigvee_{i=0,\ldots,|t(n)|} (\underline{i} \leq |t(x)| \to \varphi(x, \underline{i}, z_i)),$$

where $z_0, \ldots, z_{|t(n)|}$ are new distinct variables.

After this example it should not be difficult for the reader to go on and handle more complex cases.

10.3.1. S_2 is a theory based on a finite number of basic open axioms with induction for bounded formulas of the form

$$\varphi(\underline{0}) \wedge \forall x(\varphi(\lfloor x/2 \rfloor) \to \varphi(x)) \ \to \forall x \varphi(x).$$

The most important fragment of bounded arithmetic S_2 is the theory S_2^1 where the induction schema is restricted to Σ_1^b formulas. This theory is adequate for formalization of polynomial time computations, see Buss [1986]. Furthermore it is related to extension Frege proof systems:

10.3.2. Theorem. (Cook [1975], Buss [1986]) *If S^1_2 proves $\forall x_1 \dots \forall x_k \varphi(x_1, \dots, x_k)$, where $\varphi(x_1, \dots, x_k) \in \Pi^b_1$, then there exists a polynomial p such that the propositions $[\varphi(x_1, \dots, x_k)]_{n_1,\dots,n_k}$ have extension Frege proofs of size $\leq p(n_1, \dots, n_k)$.* □

The proof of this theorem is similar to the proof of Theorem 10.2.2, but much more involved due to the difficulties with the basic axioms.

10.3.3. This theorem naturally rises the question: is extension Frege proof system the weakest system for which we can prove this theorem? We do not know; it is possible that one can construct some pathological counterexample, but there is another reason for associating extension Frege systems with S^1_2, which we shall consider next. Following Krajíček and Pudlák [1990], we shall define a natural relation between theories and propositional proof systems.

10.3.4. Definition. (1) For a propositional proof system P we denote by $RFN(P)$ (the reflection principle for P) the $\forall\Pi^b_1$ sentence

$$\forall d, u((d : P \vdash u) \to Taut(u)),$$

where $Taut(x)$ is a Π^b_1 formula defining the set of propositional tautologies. Note that $d : P \vdash u$ (d is a P proof of a proposition u) can be written as a Σ^b_1 formula, since it is a polynomial time computable predicate.

(2) A propositional proof system P *simulates a theory* T, if for every $\varphi(x) \in \Pi^b_1$

$$T \vdash \forall x \varphi(x) \quad \Rightarrow \quad S^1_2 \vdash \forall y \exists d\ (d : P \vdash [\varphi(x)]_{|y|}).$$

(3) A propositional proof system P is *associated to* a theory T, if P simulates T and $T \vdash RFN(P)$.

Probably in (2) you expected rather a statement like in Theorems 10.2.2 and 10.3.2. In fact the condition (2) is stronger: by Buss's Theorem II.3.2, the provability of such a Π_2 statement in S^1_2 implies that it can be witnessed by a polynomial time computable function. Thus, in particular, the P proofs of $[\varphi(x)]_n$'s must be of polynomial size. So (2) means that there is a polynomial bound on the lengths of P proofs of $[\varphi(x)]_n$'s *provably in a weak theory.*

Let us also note that $RFN(P)$ is equivalent to the consistency of P assuming some "mild conditions" on P.

We shall denote by G the quantified propositional proof system based on the sequent calculus, see 8.8. Let G_i denote the subsystem of G obtained by imposing the restriction of at most i alternations of quantifiers in each formula of a proof. Let G^*_i denote G_i where we allow only tree-like proofs.

The following theorem gives some known pairs of a proof system associated to a theory (for definitions of the theories see Chapter II).

10.3.5. Theorem. (Cook [1975], Krajíček and Takeuti [1990], Krajíček and Pudlák [1990]) *The following are pairs of a theory and a proof system associated to it:* $(S^1_2, \textit{extension Frege})$, (S^i_2, G^*_i) *for* $i \geq 1$, (T^i_2, G_i) *for* $i \geq 1$, (U^1_2, G). □

Note that for U^1_2 we have two related systems, depending on which translation we take. Further results of this type were proved in Clote [1992].

Next theorem shows that under reasonable conditions the associated propositional proof system is determined up to polynomial simulation.

10.3.6. Theorem. (Krajíček and Pudlák [1990]) *Let P be a propositional proof system associated to a theory T. Suppose T contains S^1_2 and the following is provable in S^1_2: P simulates extension Frege systems and it is closed under modus ponens. Then P polynomially simulates any propositional proof system for which T proves the reflection principle.*

Thus, e.g. by Theorem 10.3.5, extension Frege systems and G^*_1 are polynomially equivalent.

Proof. Suppose $T \vdash RFN(Q)$. Let $\rho_Q(x,y)$ be the Π^b_1 formula which defines the reflection principle, i.e.,

$$\rho_Q(d,u) \;\equiv\; d : Q \vdash u \to Taut(u).$$

By the assumptions $S^1_2 \vdash \forall z(P \vdash [\rho_Q(x,y)]_z)$. We now we argue in the theory S^1_2. Thus we have

$$P \vdash [x : Q \vdash y \;\to Taut(y)]_z.$$

Since $[x : Q \vdash y \;\to Taut(y)]_z$ is $[x : Q \vdash y]_z \to [Taut(y)]_z$ and P is closed under modus ponens, we get

$$P \vdash [x : Q \vdash y]_z \quad \to \quad P \vdash [Taut(y)]_z.$$

We have also

$$P \vdash [Taut(y)]_z \quad \to \quad P \vdash y,$$

since it is true already for extension Frege systems (we leave this claim without a proof). Thus we have obtained in S^1_2

$$P \vdash [x : Q \vdash y]_z \quad \to \quad P \vdash y.$$

Back in the real world, by Buss's witnessing theorem it means that one can construct in polynomial time a proof of φ in P from a proof of $[d : Q \vdash \varphi]_n$ in P.

Now suppose that we are given a proof d of φ in Q. Substituting the numbers which encode d and φ we get *a true variable-free propositional formula* $[\underline{d} : Q \vdash \underline{\varphi}]_n$. Such formulas always have polynomial size proofs even in a Frege system. Thus we get a P proof of φ in polynomial time. □

The meaning of this theorem is that the proof system associated to a theory T is, from the point of view of T, the strongest proof system, i.e., stronger systems may be inconsistent. Let us state it formally:

10.3.7. Corollary. *Under the same assumptions as in Theorem 10.3.6, if* $T \vdash \mathcal{NP} = co\mathcal{NP}$*, then* P *is polynomially bounded.*

Proof. Since the set of propositional tautologies is $co\mathcal{NP}$-complete, the assumption $T \vdash \mathcal{NP} = co\mathcal{NP}$ means that

$$T \vdash \forall x(\sigma(x) \equiv Taut(x)), \tag{59}$$

for some $\sigma(x) \in \Sigma_1^b$. So σ defines a polynomially bounded propositional proof system Q (proofs are the witnesses for the existential bounded quantifiers). The sentence (59) implies $T \vdash RFN(Q)$. Hence, by Theorem 10.3.6, P polynomially simulates Q. But if Q is polynomially bounded, then also P must be. □

As we believe that $\mathcal{NP} \neq co\mathcal{NP}$, we expect that the corollary will be used in the contrapositive form. Let us state the nicest special case of it (proved directly by Wilkie in 1987, unpublished; as observed in Krajíček and Pudlák [1989] it also follows from results of Cook [1975] and Buss [1986]).

10.3.8. Corollary. *If extension Frege proofs are not polynomially bounded, then* S_2^1 *does not prove* $\mathcal{NP} = co\mathcal{NP}$*.* □

10.4. Optimal proof systems and consistency statements. The second translation can be used to show a link between a fundamental problem about the lengths of proofs of finite consistency statements and the existence of an optimal propositional proof system. Furthermore there is a statement from structural complexity theory which is equivalent to these problems. A set $Y \subseteq \{0,1\}^*$ is called *sparse*, if for every n, the size of $Y \cap \{0,1\}^n$ is bounded by a polynomial.

10.4.1. Theorem. (Krajíček and Pudlák [1989]) *The following are equivalent:*

1. *There exists a consistent finitely axiomatized theory* $T \supseteq S_2^1$ *such that for every consistent finitely axiomatized theory* S

$$\|Con_S(\underline{n})\|_T = n^{O(1)}.$$

2. *There exists an* optimal propositional proof system*, i.e., a propositional proof system* P *such that for every propositional proof system* Q

$$\|\varphi\|_Q = \|\varphi\|_P^{O(1)},$$

for every tautology φ*.*

3. *For every* $co\mathcal{NP}$*-set* X *there exists a nondeterministic Turing machine which accepts* X *and uses only polynomial time on every sparse subset* $Y \subseteq X$*,* $Y \in \mathcal{P}$*.*

The proof of the equivalence of 1. and 2. is based on the following two constructions. If T is an optimal theory in the sense of 1., we take a propositional proof system P defined by:

$$d : P \vdash \varphi \ \equiv_{df} \ d : T \vdash Taut(\underline{\varphi}).$$

If, on the other hand, P is an optimal propositional proof system, we take the theory T defined by:

$$T =_{df} S_2^1 + RFN(P).$$

We omit the rest of the proof. □

Given a propositional proof system P which is not polynomially bounded, we can produce, using this theorem, a sequence of tautologies which surely do not have polynomial size proofs in P. Unfortunately, the tautologies will be rather complex artificial statements, thus not amenable to a combinatorial analysis. However, as noted by Krajíček [1995], one can use the polynomial reductions, by which $\mathcal{NP}$ completeness results are proved, to turn these tautologies into simple combinatorial statements. For instance one can construct a sequence of nonhamiltonian graphs, such that there are no polynomial size proofs in P of the tautologies expressing that the graphs are nonhamiltonian. Thus the problem reduces to finding a class of nonhamiltonian graphs for which it is difficult to prove in P that they are nonhamiltonian.

11. Bibliographical remarks for further reading

In this section we shall give a few more references which have not been mentioned in the main text. This should serve to the reader who is interested in the history of the subject or who wants to learn more about it. Our aim is not to complete the list of references about *results* on the lengths of proofs, rather we want to partially complement the above presentation which concentrated on *methods* used in this research area. Thus, in particular, we shall not repeat results described above.

Probably the oldest recorded paper on the subject is Gödel [1936]. In this two-page abstract he stated the result that there is a speed-up between the lengths of proofs of formulas provable in i-th order and $i+1$-st order arithmetics. To quote him: *The transition to the logic of the next higher type not only results in certain previously unprovable propositions becoming provable, but also in it becoming possible to shorten extraordinarily infinitely many of the proofs already available.* The length of proofs is considered to be the number of steps and the speed-up is $\phi(n)$ for any function ϕ "computable" in the lower system. There was no proof given in the paper. For a full proof of this statement see Buss [1994].

Another important writing of Gödel which was discovered only a few years ago, is the letter by Gödel [1993]. In that letter he posed the question whether one can decide in linear, quadratic, etc. time in n whether a given formula has a proof of length (= number of symbols) n. Now we know that this problem is $\mathcal{NP}$-complete. See Buss [1995a] for a discussion and a proof of an unproven claim of Gödel.

Looking at the literature it seems that the subject lay dormant for several decades. I think that many people thought about problems on the lengths of proofs, but the things that they actually could prove did not look interesting enough, especially when compared with other fancy topics like set theory. Furthermore some basic concepts were missing (one of such crucial things was the distinction between polynomial size

and exponential size). This can be documented by a remark of Kreisel [1967,page 241], who mentions a conversation with Gödel where Gödel asked the question of what are the lengths of proofs of finite consistency statements. No paper had been written about it until Friedman [1979], but he did not consider it to be worth publishing.

At the early stages, Georg Kreisel was one of the main proponents of this field. His student Statman [1978] determined the increase of the lengths of proofs in cut-elimination and Herbrand's Theorem. Another of his students, Baaz (see Baaz and Pudlák [1993], Baaz and Zach [1995]), made significant progress in Kreisel's Conjecture. As seen on Kreisel's Conjecture, Kreisel was more interested in *positive* results in the sense of deriving more information from the proofs than just the mere fact that the statement is true. Logic should help mathematicians to get more or better results, rather than only to show impossibilities of certain proofs, see e.g. Kreisel [1990]. From this point of view, one of the greatest successes in proof theory was the result of Luckhardt [1989], deriving explicit bounds on approximation of algebraic numbers by rational numbers (Roth's theorem), using Herbrand's theorem.

Originally the interest in the lengths of proofs was based mainly on philosophical and methodological considerations. With the advent of computers a new practical reason appeared: *automated theorem proving.* The main tool in automated theorem proving is the resolution system for first order logic, see e.g. Chang and Lee [1973]. For us, theoreticians, most of the papers are too much applied, however there are several results which are important also for theory. Such a notable result is the exponential lower bound for propositional *regular* resolution of Tsejtin [1968]. The question about the efficiency of proof-search strategies are often nontrivial mathematical problems, let us mention at least some results of this type Baaz and Leitsch [1992,1994]. There are several books about the complexity of logical calculi, e.g. Eder [1992]; they deal mainly with the first order logic.

The next important stimulus was the rise of complexity theory. The lengths of proofs is just one of several research areas which combine logic and complexity theory. Another one, which is closely related to it, is the complexity of logical theories. The problem is how efficiently can we decide if a sentence is provable in a given decidable theory T (e.g., Presburger arithmetic). Note that an upper bound on the lengths of proofs in T gives an upper bound on a *nondeterministic* procedure for decidability. Often this bound is not very far from the best. We refer the reader to the surveys Rabin [1977] and Compton and Henson [1990].

We can say that the research into complexity of proofs really started with the seminal paper of Parikh [1971] which introduced several important concepts and proved basic results about them: *speed-up for first order theories, theories which are inconsistent but are consistent for practical purposes, and bounded arithmetic.* Soon after it, he published a basic result on Kreisel's Conjecture in Parikh [1973]. He proved that the conjecture is true, if we take Peano arithmetic with $+$ and $\times$ as ternary relations instead of function symbols. That proof has been a paradigm for all subsequent proofs of instances of Kreisel's Conjecture.

After that several people started to work on these subjects. One of the most

influential researchers in this field has been Orevkov. We shall mention only the most important papers of the many that he published. Orevkov [1982] gave a different proof of the lower bounds on the lengthening of proofs in cut-elimination and Orevkov [1986] gave more precise upper bounds. Orevkov [1987b] introduced explicitly the concept of the skeleton and Orevkov [1987a] proved several results related to Kreisel's Conjecture. All these results, and many more, are covered in Orevkov [1993].

There are more results on the complexity of first order proofs. Of those that we have not presented yet, let us mention the dissertation of Ignjatović [1990]. He proved a nonelementary speed up between Primitive Recursive Arithmetic and $I\Sigma_0$.

Currently the most active area is propositional logic and bounded arithmetic. The fundamental paper is Cook [1975], where a relation of the lengths of proofs in propositional logic and provability in arithmetic was considered for the first time. The most influential papers in bounded arithmetic after Parikh [1971] were written by Paris and Wilkie; let us mentioned at least the Paris and Wilkie [1985] paper on counting problems which influenced very much research on the complexity of propositional logic. The basic book on bounded arithmetic is due to Buss [1986]. Another fundamental paper is by Ajtai [1994a], where he introduced the method of random restrictions into propositional logic, which had already been used in complexity theory. This development has been partially described in this chapter and also in Chapter II; much more can be found in the monograph by Krajíček [1995], which covers the whole area in detail except for the most recent results. As this manuscript is being finalized, new exciting results are being obtained on the polynomial calculus by Razborov [n.d.], Krajíček [1997b] and Riis and Sitharam [1997].

Acknowledgments

I would like to thank Sam Buss for helping me with the preparation of the manuscript and suggesting several improvements and Jan Krajíček for checking the manuscript. The preparation of the article was supported by grant #A1019602 of the Academy of Sciences of the Czech Republic and the cooperative research grant INT-9600919/ME-103 of the U.S. National Science Foundation and the Czech Republic Ministry of Education.

References

M. Ajtai

[1990] Parity and the pigeonhole principle, in: *Feasible Mathematics: A Mathematical Sciences Institute Workshop held in Ithaca, New York, June 1989*, S. R. Buss and P. J. Scott, eds., Birkhäuser, Boston, pp. 1–24.

[1994a] The complexity of the pigeonhole principle, *Combinatorica*, 14, pp. 417–433. Extended abstract in *Proc. 29th Annual IEEE Symposium on Foundations of Computer Science*, 1988, pp. 346-355.

[1994b] The independence of the modulo p counting principles, in: *Proceedings of the 26th Annual ACM Symposium on the Theory of Computing*, Association for Computing Machinery, New York, pp. 402–411.

[1995] On the existence of modulo p cardinality functions, in: *Feasible Mathematics II*, P. Clote and J. B. Remmel, eds., Birkhäauser, Boston, pp. 1–14.

N. Alon and R. Boppana

[1987] The monotone circuit complexity of boolean functions, *Combinatorica*, 7, pp. 1–22.

S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy

[1992] Proof verification and hardness of approximation problems, in: *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, IEEE Computer Society, Piscataway, New Jersey, pp. 14–23.

M. Baaz and A. Leitsch

[1992] Complexity of resolution proofs and function introduction, *Annals of Pure and Applied Logic*, 20, pp. 181–215.

[1994] On Skolemization and proof complexity, *Fundamenta Mathematicae*, 20.

M. Baaz and P. Pudlák

[1993] Kreisel's conjecture for $L\exists_1$, in: *Arithmetic Proof Theory and Computational Complexity*, P. Clote and J. Krajíček, eds., Oxford University Press, pp. 30–39.

M. Baaz and R. Zach

[1995] Generalizing theorems in real closed fields, *Annals of Pure and Applied Logic*, 75, pp. 2–23.

P. Beame, R. Impagliazzo, J. Krajíček, T. Pitassi, and P. Pudlák

[1996] Lower bounds on Hilbert's Nullstellensatz and propositional proofs, *Proceedings of the London Mathematical Society*, 73, pp. 1–26.

P. Beame, R. Impagliazzo, J. Krajíček, T. Pitassi, P. Pudlák, and A. Woods

[1992] Exponential lower bounds for the pigeonhole principle, in: *Proceedings of the 24th Annual ACM Symposium on the Theory of Computing*, Association for Computing Machinery, New York, pp. 200–221.

P. Beame and T. Pitassi

[1996] Exponential separation between the matching principles and the pigeonhole principle, *Annals of Pure and Applied Logic*, 80, pp. 195–228.

S. Bellantoni, T. Pitassi, and A. Urquhart

[1992] Approximation and small-depth Frege proofs, *SIAM Journal on Computing*, 21, pp. 1161–1179.

E. W. Beth

[1959] *The Foundations of Mathematics*, North-Holland, Amsterdam.

M. L. Bonet and S. R. Buss

[1993] The deduction rule and linear and near-linear proof simulations, *Journal of Symbolic Logic*, 58, pp. 688–709.

M. L. Bonet, T. Pitassi, and R. Raz

[1997a] Lower bounds for cutting planes proofs with small coefficients, *Journal of Symbolic Logic*, 62, pp. 708–728. An earlier version appeared in *Proc. Twenty-Seventh Annual ACM Symposium on the Theory of Computing*, 1995, pp. 575-584.

[1997b] No feasible interpolation for TC^0-Frege proofs, in: *Proceedings of the 38th Annual Symposium on Foundations of Computer Science*, IEEE Computer Society, Piscataway, New Jersey, pp. 254–263.

S. R. Buss

[1986] *Bounded Arithmetic*, Bibliopolis, Napoli. Revision of 1985 Princeton University Ph.D. thesis.

[1987] Polynomial size proofs of the propositional pigeonhole principle, *Journal of Symbolic Logic*, 52, pp. 916–927.

[1991a] Propositional consistency proofs, *Annals of Pure and Applied Logic*, 52, pp. 3–29.

[1991b] The undecidability of k-provability, *Annals of Pure and Applied Logic*, 53, pp. 75–102.

[1994] On Gödel's theorems on lengths of proofs I: Number of lines and speedup for arithmetics, *Journal of Symbolic Logic*, 59, pp. 737–756.

[1995a] On Gödel's theorems on lengths of proofs II: Lower bounds for recognizing k-symbol provability, in: *Feasible Mathematics II*, P. Clote and J. B. Remmel, eds., Birkhäauser, Boston, pp. 57–90.

[1995b] Some remarks on lengths of propositional proofs, *Archive for Mathematical Logic*, 34, pp. 377–394.

S. R. Buss and P. Clote

[1996] Cutting planes, connectivity and threshold logic, *Archive for Mathematical Logic*, 35, pp. 33–62.

S. R. Buss, R. Impagliazzo, J. Krajíček, P. Pudlák, A. A. Razborov, and J. Sgall

[1996/1997] Proof complexity in algebraic systems and constant depth Frege systems with modular counting, *Computational Complexity*, 6, pp. 256–298.

S. R. Buss and T. Pitassi

[1997] *Resolution and the Weak Pigeonhole Principle.* Typeset manuscript, to appear in *CSL'97*.

S. R. Buss and Gy. Turán

[1988] Resolution proofs of generalized pigeonhole principles, *Theoretical Computer Science*, 62, pp. 311–317.

C.-L. Chang and R. C.-T. Lee

[1973] *Symbolic Logic and Mechanical Theorem Proving*, Academic Press, New York.

M. Clegg, J. Edmonds, and R. Impagliazzo

[1996] Using the Groebner basis algorithm to find proofs of unsatisfiablility, in: *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing*, Association for Computing Machinery, New York, pp. 174–183.

P. Clote

[1992] ALOGTIME and a conjecture of S. A. Cook, *Annals of Mathematics and Artificial Inteligence*, 6, pp. 57–106.

P. Clote and J. Krajíček

[1993] *Arithmetic, Proof Theory and Computational Complexity*, Oxford University Press.

K. J. Compton and C. W. Henson

[1990] A uniform method for proving lower bounds on the computational complexity of logical theories, *Annals of Pure and Applied Logic*, 48, pp. 1–79.

S. A. Cook

[1975] Feasibly constructive proofs and the propositional calculus, in: *Proceedings of the Seventh Annual ACM Symposium on the Theory of Computing*, Association for Computing Machinery, New York, pp. 83–97.

S. A. Cook and R. A. Reckhow

[1979] The relative efficiency of propositional proof systems, *Journal of Symbolic Logic*, 44, pp. 36–50.

W. Cook, C. R. Coullard, and Gy. Turán
[1987] On the complexity of cutting plane proofs, *Discrete Applied Mathematics*, 18, pp. 25–38.

W. Craig
[1957a] Linear reasoning. A new form of the Herbrand-Gentzen theorem, *Journal of Symbolic Logic*, 22, pp. 250–268.
[1957b] Three uses of the Herbrand-Gentzen theorem in relating model theory and proof theory, *Journal of Symbolic Logic*, 22, pp. 269–285.

M. Dowd
[1979] *Propositional Representation of Arithmetic Proofs*, PhD thesis, University of Toronto.
[1985] *Model-Theoretic Aspects of $P \neq NP$.* Typewritten manuscript.

A. G. Dragalin
[1985] Correctness of inconsistent theories with notions of feasibility, in: *Computation Theory, Fifth Symposium Proceedings*, A. Skowron, ed., vol. 108 of Lecture Notes in Computer Science #208, Springer-Verlag, Berlin, pp. 58–79.

E. Eder
[1992] *Relative Complexities of First Order Calculi*, Verlag Vieweg.

A. Ehrenfeucht and J. Mycielski
[1971] Abbreviating proofs by adding new axioms, *Bulletin of the American Mathematical Society*, 77, pp. 366–367.

W. M. Farmer
[1984] *Length of Proofs and Unification Theory*, PhD thesis, University of Wisconsin, Madison.
[1988] A unification algorithm for second order monadic terms, *Annals of Pure and Applied Logic*, 39, pp. 131–174.

J. Ferrante and C. W. Rackoff
[1979] *The Computational Complexity of Logical Theories*, Lecture Notes in Mathematics #718, Springer-Verlag, Berlin.

H. M. Friedman
[1975] One hundred and two problems in mathematical logic, *Journal of Symbolic Logic*, 40, pp. 113–129.
[1979] *On the consistency, completeness, and correctness problems.* Ohio State University, unpublished.

Yu. V. Gavrilenko
[1984] Monotone theories of feasible numbers, *Doklady Akademii Nauk SSSR*, 276, pp. 18–22.

G. Gentzen
[1935] Untersuchungen über das Logische Schliessen, *Mathematische Zeitschrift*, 39, pp. 176–210 and 405–431.

J.-Y. Girard
[1989] *Proofs and Types*, Cambridge University Press.

K. Gödel
[1936] Über die Länge von Beweisen, *Ergebnisse eines Mathematischen Kolloquiums*, pp. 23–24. English translation in *Kurt Gödel: Collected Works, Volume 1*, pages 396-399, Oxford University Press, 1986.
[1993] A letter to von Neumann, March 20, 1956, in: *Arithmetic Proof Theory and Computational Complexity*, P. Clote and J. Krajíček, eds., Oxford University Press, pp. vii–ix.

A. Goerdt

[1991] Cutting plane versus Frege proof systems, in: *Computer Science Logic: 4th workshop, CSL '90*, E. Börger and et al., eds., Lecture Notes in Computer Science #533, Springer-Verlag, Berlin, pp. 174–194.

A. Grzegorczyk

[1974] *An Outline of Mathematical Logic*, D. Reidel Publishing Co., Dordrecht-Boston, Mass., PWN-Polish Scientific Publishers, Warsaw. Translation of *Zarys logiki matematicznej*, Państwowe Wyadwnictwo Naukowe, 1969.

P. Hájek, F. Montagna, and P. Pudlák

[1993] Abbreviating proofs using metamathematical rules, in: *Arithmetic Proof Theory and Computational Complexity*, P. Clote and J. Krajíček, eds., Oxford University Press, pp. 197–221.

P. Hájek and P. Pudlák

[1993] *Metamathematics of First-order Arithmetic*, Perspectives in Mathematical Logic, Springer-Verlag, Berlin.

G. Hajós

[1961] Über eine Konstruktion nicht n-färberer Graphen, *Wiss. Zeitschr. M. Luther Univ. Halle-Wittenberg*, A 10, pp. 116–117.

A. Haken

[1985] The intractability of resolution, *Theoretical Computer Science*, 39, pp. 297–308.

A. Haken and S. A. Cook

[n.d.] *An Exponential Lower Bound for the Size of Monotone Real Circuits.* To appear in *J. of Computer and System Science.*

J. Håstad

[1986] *Computation Limits of Small Depth Circuits*, MIT Press.

D. Hilbert and W. Ackermann

[1928] *Grundzüge der theoretischen Logik*, Springer-Verlag, Berlin.

D. Hilbert and P. Bernays

[1934] *Grundlagen der Mathematik I*, Springer-Verlag, Berlin.

[1939] *Grundlagen der Mathematik II*, Springer-Verlag, Berlin.

A. Ignjatović

[1990] *Fragments of First and Second Order Arithmetic and Length of Proofs*, PhD thesis, University of California, Berkeley.

R. Impagliazzo, P. Pudlák, and J. Sgall

[1997] *Lower Bounds for the Polynomial Calculus and the Groebner Basis Algorithm*, Tech. Rep. TR97-042, Electronic Colloquium on Computational Complexity (ECCC).

J. Johannsen

[1997] *Lower Bounds for Monotone Real Circuit Depth and Formula Size and Tree-like Cutting Planes*, Tech. Rep. TR97-032, Electronic Colloquium on Computational Complexity, http://www.eccc.uni-trier.de/eccc/.

J. Krajíček

[n.d.] *Discretely Ordered Modules as a First-Order Extension of the Cutting Planes Proof System.* To appear in the *J. of Symbolic Logic.*

[1989a] On the number of steps in proofs, *Annals of Pure and Applied Logic*, 41, pp. 153–178.

[1989b] Speed-up for propositional Frege systems via generalizations of proofs, *Commentationes Mathematicae Universitatis Carolinae*, 30, pp. 137–140.

[1994a] Lower bounds to the size of constant-depth propositional proofs, *Journal of Symbolic Logic*, 59, pp. 73–86.

[1994b] On Frege and extended Frege proof systems, in: *Feasible Mathematics II*, J. Krajíček and J. B. Remmel, eds., Birkhäuser, Boston, pp. 284–319.

[1995] *Bounded Arithmetic, Propositional Logic and Complexity Theory*, Cambridge University Press.

[1997a] Interpolation theorems, lower bounds for proof systems and independence results for bounded arithmetic, *Journal of Symbolic Logic*, 62, pp. 457–486.

[1997b] *On the Degree of Ideal Membership Proofs from Uniform Families of Polynomials over a Finite Field.* Typeset manuscript.

J. Krajíček and P. Pudlák

[1988] The number of proof lines and the size of proofs in first-order logic, *Archive for Mathematical Logic*, 27, pp. 69–84.

[1989] Propositional proof systems, the consistency of first-order theories and the complexity of computations, *Journal of Symbolic Logic*, 54, pp. 1063–1079.

[1990] Quantified propositional calculi and fragments of bounded arithmetic, *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 36, pp. 29–46.

[1998] Some consequences of cryptographical conjectures for S_2^1 and EF, *Information and Computation*, 140, pp. 82–94.

J. Krajíček, P. Pudlák, and A. Woods

[1995] An exponential lower bound to the size of bounded depth Frege proofs of the pigeonhole principle, *Random Structures and Algorithms*, 7, pp. 15–39.

J. Krajíček and G. Takeuti

[1990] On bounded Σ_1^1-polynomial induction, in: *Feasible Mathematics*, S. R. Buss and P. J. Scott, eds., Birkhäuser, Boston, pp. 259–280.

G. Kreisel

[1967] Mathematical logic: What has it done for the philosophy of mathematics, in: *Bertrand Russell: Philosopher of the Century, Essays in his Honour*, R. Shoenemann, ed., George Allen and Unwin, pp. 201–272.

[1990] Logical aspects of computation: Contributions and distractions, in: *Logic and Computer Science*, Academic Press, New York, pp. 205–278.

H. Luckhardt

[1989] Herbrand-Analysen zweier Beweise des Satzes von Roth: polynomiale Anzahlschranken, *Journal of Symbolic Logic*, 54, pp. 234–263.

T. Miyatake

[1980] On the length of proofs in formal systems, *Tsukuba Journal of Mathematics*, 4, pp. 115–125.

D. Mundici

[1984] NP and Craig's interpolation theorem, in: *Logic Colloquium '82*, G. Lolli, G. Longo, and A. Marcja, eds., North-Holland, Amsterdam, pp. 345–358.

E. Nelson

[1986] *Predicative Arithmetic*, Princeton University Press.

V. P. Orevkov

[1982] Lower bounds on the increase in complexity of deductions in cut elimination, *Journal of Soviet Mathematics*, 20. Original Russian version in *Zap. Nauchn. Sem. L.O.M.I. 88* (1979), pp.137-162.

[1986] Upper bound on the lengthening of proofs by cut elimination, *Journal of Soviet Mathematics*, 34, pp. 1810–1819. Original Russian version in *Zap. Nauchn. Sem. L.O.M.I. 137* (1984), pp.87-98.

[1987a] Lower bounds on the lengths of derivations in arithmetic in terms of the complexity of terms involved in the derivations, *Soviet Mathematics Doklady*, 35, pp. 579–582. Original Russian version in *Dokl. Akad. Nauk. 294/4* (1987).

[1987b] Reconstruction of a proof from its scheme, *Soviet Mathematics Doklady*, 35, pp. 326–329. Original Russian version in Dokl. Akad. Nauk. **293** (1987) 313-316.

[1990] Correctness of short proofs in theory with notions of feasibility, in: *COLOG-88: International Conference on Computer Logic, Tallinn, USSR, Dec. 1988, Proceedings*, P. Martin-Löf and G. E. Mints, eds., Lecture Notes in Computer Science #417, Springer-Verlag, Berlin, pp. 242–245.

[1993] *Complexity of Proofs and Their Transformations in Axiomatic theories*, vol. 128 of Translations of Mathematical Monographs, American Mathematical Society, Providence, Rhode Island.

R. Parikh

[1971] Existence and feasibility in arithmetic, *Journal of Symbolic Logic*, 36, pp. 494–508.

[1973] Some results on the lengths of proofs, *Transactions of the American Mathematical Society*, 177, pp. 29–36.

J. B. Paris and A. J. Wilkie

[1985] Counting problems in bounded arithmetic, in: *Methods in Mathematical Logic, Proceedings of the 6-th Latin American Symposium, Caracas, Venezuella*, C. A. Di Prisco, ed., Lecture Notes in Mathematics #1130, Springer-Verlag, Berlin, pp. 317–340.

T. Pitassi, P. Beame, and R. Impagliazzo

[1993] Exponential lower bounds for the pigeonhole principle, *Computational Complexity*, 3, pp. 97–140.

T. Pitassi and A. Urquhart

[1992] The complexity of the HajósnameindexHajós, G. calculus, in: *Proceedings of the 33th Annual Symposium on Foundations of Computer Science*, IEEE Computer Society, Piscataway, New Jersey, pp. 187–196.

D. Prawitz

[1970] Ideas and results in proof theory, in: *Proceedings of the Second Scandinavian Logic Symposium*, J. E. Fenstad, ed., North-Holland, Amsterdam.

P. Pudlák

[1985] Cuts, consistency statements and interpretation, *Journal of Symbolic Logic*, 50, pp. 423–441.

[1986] On the lengths of proofs of finitistic consistency statements in first order theories, in: *Logic Colloquium '84*, J. B. Paris, A. J. Wilkie, and G. M. Wilmers, eds., North-Holland, Amsterdam, pp. 165–196.

[1987] Improved bounds to the lengths of proofs of finitistic consistency statements, in: *Logic and Combinatorics*, S. G. Simpson, ed., vol. 65 of Contemporary Mathematics, American Mathematical Society, Providence, Rhode Island, pp. 309–331.

[1991] Ramsey's theorem in bounded arithmetic, in: *Computer Science Logic '90*, E. Börger and et al., eds., Lecture Notes in Computer Science #533, Springer-Verlag, Berlin, pp. 308–312.

[1997] Lower bounds for resolution and cutting planes proofs and monotone computations, *Journal of Symbolic Logic*, 62, pp. 981–998.

M. O. Rabin

[1977] Decidable theories, in: *Handbook of Mathematical Logic*, J. Barwise, ed., North-Holland, Amsterdam, pp. 595–629.

A. A. Razborov

[n.d.] *Lower Bounds for the Polynomial Calculus.* To appear in *Computational Complexity.*

[1985] Lower bounds on the monotone complexity of some boolean functions, *Doklady Akademii Nauk SSSR*, 282, pp. 1033–1037. English translation in: Soviet Mathem. Doklady, 31, pp. 354-357.

[1994] *On provably disjoint NP-pairs*, Tech. Rep. RS-94-36, Basic Research in Computer Science Center, Aarhus, Denmark, November. http://www.brics.dk/index.html.

[1996] Lower bounds for propositional proofs and independence results in Bounded Arithmetic, in: *Automata, languages and programming: 23rd international colloquium, ICALP '96*, F. Meyer auf der Heide and B. Monien, eds., Lecture Notes in Computer Science #1099, Springer-Verlag, Berlin, pp. 48–62.

A. A. Razborov, A. Widgerson, and A. C.-C. Yao

[1997] Read-once branching programs, rectangular proofs of the pigeonhole principle and the transversal calculus, in: *Proceedings of the 29th Annual ACM Symposium on the Theory of Computing*, Association for Computing Machinery, New York, pp. 739–748.

R. A. Reckhow

[1976] *On the Lengths of Proofs in the Propositional Calculus*, PhD thesis, Department of Computer Science, University of Toronto. Technical Report #87.

S. Riis and M. Sitharam

[1997] *Non-constant Degree Lower Bounds imply Linear Degree Lower Bounds*, Tech. Rep. TR97-048, Colloquium on Computation Complexity, ECCC, http://www.eccc.uni-trier.de/eccc/.

B. Russell

[1906] The theory of implication, *American Journal of Mathematics*, 28, pp. 159–202.

R. M. Smullyan

[1968] *First-Order Logic*, Springer-Verlag, Berlin.

R. M. Solovay

[1990] *Upper Bounds on the Speedup of GB over ZF.* preprint.

R. Statman

[1977] Complexity of derivations from quantifier-free Horn formulae, mechanical introduction of explicit definitions, and refinement of completeness theorems, in: *Logic Colloquium '76*, R. O. Gandy and J. M. E. Hyland, eds., North-Holland, Amsterdam, pp. 505–517.

[1978] Proof search and speed-up in the predicate calculus, *Annals of Mathematical Logic*, 15, pp. 225–287.

[1981] Speed-up by theories with infinite models, *Proceedings of the American Mathematical Society*, 81, pp. 465–469.

G. Takeuti

[1987] *Proof Theory*, North-Holland, Amsterdam, 2nd ed.

[1990] Some relations among systems for bounded arithmetic, in: *Mathematical Logic, Proceedings of the Heyting 1988 Summer School*, P. P. Petkov, ed., Plenum Press, New York, pp. 139–154.

A. Tarski

[1936] Der Warheitsbegriff in den formalisierten Sprachen, *Studia Philosophica, Commentarii Societatis Philosophicae Polonorum*, 1, pp. 261–405.

G. S. Tsejtin

[1968] On the complexity of derivations in propositional calculus, in: *Studies in mathematics and mathematical logic, Part II*, A. O. Slisenko, ed., pp. 115–125. in Russian.

G. S. Tsejtin and A. A. Čubarjan

[1975] On some bounds to the lengths of logical proofs in classical propositional calculus, *Trudy Vyčisl. Centra AN ArmSSR i Erevan. Univ.*, 8, pp. 57–64. In Russian.

R. L. Vaught
[1967] On axiomatizability by a schema, *Journal of Symbolic Logic*, 32, pp. 473–479.

A. C.-C. Yao
[1985] Separating the polynomial time hierarchy by oracles, in: *Proceedings of the 26th Annual Symposium on Foundations of Computer Science*, IEEE Computer Society, Piscataway, New Jersey, pp. 1–10.

Name Index

Subject Index

Discard this page.