How to lie without being (easily) convicted and the lengths of proofs in propositional calculus

Pavel Pudlák*[1] and Samuel R. Buss**[2]

[1] Mathematics Institute, Academy of Sciences of the Czech Republic, Prague
[2] Department of Mathematics, University of California, San Diego

Abstract. We shall describe two general methods for proving lower bounds on the lengths of proofs in propositional calculus and give examples of such lower bounds. One of the methods is based on interactive proofs where one player is claiming that he has a falsifying assignment for a tautology and the second player is trying to convict him of a lie. The second method is based on boolean valuations. For the first method, a $\log n + \log\log n - O(\log\log\log n)$ lower bound is given on the lengths of interactive proofs of certain permutation tautologies.[3]

1 Introduction

We are interested in proving lower bounds on the lengths of proofs in propositional calculus. There are two main motivations for this research.

Firstly this question is connected with the famous open problem "$\mathcal{NP} =? co\mathcal{NP}$", since a proof system for propositional calculus can be thought of as a nondeterministic procedure for the $co\mathcal{NP}$-complete set of propositional tautologies, Cook [4]. Thus proving superpolynomial lower bounds on the lengths of proofs in increasingly stronger proof systems parallels in a sense an approach to the problem $\mathcal{P} =? \mathcal{NP}$, where for restricted classes of circuits superpolynomial lower bounds are proven for the size of circuits computing $\mathcal{NP}$ sets — this is done with the hope that eventually techniques will be found which will work for all propositional proof systems and all boolean circuits.

The second motivation is that this seems to be the most promising way of proving independence of interesting sentences from fragments of arithmetic. The fragments that we have in mind are often

* Partially supported by US-Czechoslovak Science and Technology Program grant No. 93025

** Partially supported by US-Czechoslovak Science and Technology Program grant No. 93025 and by NSF grant DMS92-05181

[3] This paper has been published in: Proc. Computer Science Logic, Kazimierz, Poland, 1994, eds. L. Pacholski and J. Tiuryn, Springer LNCS 933, pages 151-162

referred to by a generic name *Bounded Arithmetic*. For many theories R of bounded arithmetic one can find an associated propositional proof system R^{prop} [4, **?**, 10]. For a given theory R of arithmetic, R^{prop} is the strongest system system whose soundness is provable in the theory R and which simulates provability in R. The simulation means that for a certain class of universal sentences, if a sentence is provable in the theory R and we translate it into a sequence of tautologies expressing finite instances of the sentence, then the tautologies have polynomial size proofs in the system R^{prop}. A superpolynomial lower bound on the size of proofs in the proof system R^{prop} would imply independence of $\mathcal{NP} =? co\mathcal{NP}$ from R. Thus even partial results in this approach to the problem $\mathcal{NP} =? co\mathcal{NP}$ may have interesting consequences.

The most important class of propositional proof systems is called *Frege systems*. This concept was defined by Cook and Reckhow [6, 5] and was intended to capture the properties of the most common propositional proof systems. Formally, a Frege system is determined by a complete finite basis of connectives and a finite set of rules

$$\frac{\varphi_1(p_1,\ldots,p_m),\ldots,\varphi_k(p_1,\ldots,p_m)}{\varphi(p_1,\ldots,p_m)} \tag{1}$$

which form a sound and implicationally complete system. Let us note that when applying the rules we use their substitutional instances, but the general rule of substitution is not allowed. A typical representative Frege system is based on finitely many axiom schemas (zero premise rules) and the Modus Ponens rule. There are two measures of complexity that one uses for such proofs: the size of the proof (which include the sizes of the formulas in it) and the number of steps (which counts only the number of formulas used in the proof). The concept of the Frege system is very robust with respect to each measure: every two systems polynomially simulate each other. Moreover they are equivalent in this sense with sequent calculi with the cut rule. For their associated theories of bounded arithmetic, see [4, 3, 9]. So far, superpolynomial lower bounds have been proved only for more restricted systems, see e.g. [1, 2].

In this paper we introduce two frameworks for proving lower bounds for Frege systems and their restricted versions. First we shall define an interactive way of proving propositional tautologies. This game is well-known, however the relation of the length of the game to the lengths of Frege proofs is (as far as we know) new. Namely, the minimal number of rounds in the game is proportional to the logarithm of the minimal number of proof steps in a Frege proof. The inspiration of this game comes from some lower bound techniques in complexity theory, the so-called adversary arguments and certain game-theoretical characterizations of circuit complexity measures. We shall show that it is trivial that most tautologies require

interactive games of at least $\log n$ rounds (all logarithms in this paper are base two). However, we shall also prove a lower bound of $\log n + \log\log n - O(\log\log\log n)$ on the number of rounds in interactive games for some tautologies consisting of randomly chosen permutations of conjunctions. Here, n is the number of distinct subformulas of the tautology.

The second approach for lower bounds on Frege proofs is based on valuations in boolean algebras. This has actually been used implicitly in [2], and other proofs can be interpreted in such a way.

For the reader who wishes to get a deeper knowledge about lower bounds in propositional calculus we recommend the forthcoming book by Krajíček [8] and a forthcoming survey by the first author [**?**].

2 Interactive proofs of tautologies

We shall introduce a game using a real life situation as an example. Suppose you are a prosecutor who wishes to convict someone at a trial. What is he saying is a blatant lie for you, but the judge, and especially the jury, need a *proof without any doubts*. In particular they will not accept a long formal proof of a contradiction in his testimony. Instead, the require you to ask the defendant several questions that eventually force the liar to say some simple contradiction.

Let us describe this game more formally. There are two players *Prover* and *Adversary*, who play the roles of a prosecutor and a lying defendant, respectively. 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 $\varphi_1, \ldots, \varphi_k$, Adversary has assigned values to the $k+1$ many formulas $\varphi_1, \ldots, \varphi_k, \circ(\varphi_1, \ldots, \varphi_k)$ which 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 shall call the game *Prover-Adversary game*. We define that a proposition φ is *provable in this game,* if Prover has a winning strategy. Furthermore a natural measure of complexity of such proofs is *the minimal number of rounds needed to convict any Adversary.* The following is easy and follows from Proposition 2, but it helps to understand the concept.

Proposition 1. *The Prover-Adversary game is a complete proof system.*

Proof. 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 suppose φ is a tautology and let Prover ask all subformulas, including the variables, of φ. □

What is more interesting is the relation of the number of rounds in the game to the number of steps in a Frege proof.

Proposition 2. *The minimal number of rounds in the game needed to prove φ is proportional to the logarithm of the minimal number of steps in a Frege proof of φ.*

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

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

We use the notations $\alpha \mapsto 1$ or $\alpha \mapsto 1$ to denote the conditions that Adversary has stated that α has truth value 1 or 0, respectively.
If Adversary tries to be consistent as long as possible, Prover needs only a constant number of questions to force him to assign 1 to an axiom. Thus he can force value $\phi_1 \mapsto 1$. Also he needs only a constant number of rounds of questions to get $\phi_k \mapsto 0$, since $\phi_k \mapsto 0$. 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. Another constant number of rounds suffices 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 ϕ_{i_j} it takes only $O(\log i)$ rounds to force $\phi_{i_j} \mapsto 1$ (or to get an elementary contradiction), since Prover can force $\psi_i \mapsto 1$ in $O(\log n)$ rounds using binary search again. Once the premises get the value 1 and the conclusion value 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 $2^{O(r)}$. It is well-known that a sequent proof can be transformed into a Frege proof with at most polynomial increase.
Consider a particular play P, let $\alpha_1, \ldots, \alpha_t$, $t \leq r$ be the formulas asserted to have value 1 by Adversary in response to Prover's questions, where we have added (or removed) negations if Adversary answered 0. (In particular α_1 is $\neg\varphi$). Thus $\alpha_1 \wedge \ldots \wedge \alpha_t$ is false, hence $\rightarrow \neg\alpha_1, \ldots, \neg\alpha_t$ is a true sequent. Moreover, as easily

seen, it has a proof with $t + O(1)$ number of lines, since there is a *simple contradiction* in the statements $\alpha_1 \dots \alpha_t$. The proof of φ is constructed by taking proofs of all such sequents and then using cuts eliminating successively all formulas except φ. This is possible due to the structure of the possible plays. Namely,

> For each play P with questions $\alpha_1, \dots, \alpha_i$, and each $j \leq i$, there is a another play P' in which the first j questions are the same as in P and in which the first $j-1$ answers are the same and the j-th answer is different.

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 is in a tree form, except possibly for constant size pieces at the leaves, which can be easily changed into such a form. Thus we get:

Corollary 3. (Krajíček [?]) *A Frege proof can be transformed into a tree-like Frege proof with at most polynomial increase of size.*

Proof. Let an arbitrary Frege proof of size n be given. First transform it into the Prover-Adversary game, thus we get a game with $O(\log n)$ rounds. Transforming it into a sequent proof we get a proof of size $2^{O(\log n)} = n^{O(1)}$. Then one can check that the translation of this proof into a Frege proof can be done so that the tree form is preserved. □

This corollary is not surprising, since the main idea of the first part of the proof of Proposition 2 is the same as in Krajíček's proof. In fact the number of rounds characterizes more precisely the log of the minimal number of steps of a proof in a *tree form.* Using these transformation we get, however, still a little more information: we get a kind of a normal form of a proof – proofs which use only cuts except for the top part (at the leaves), i.e. something dual to the cut-free proofs.

Let us stress that we can also characterize the size of Frege proofs using this game, if we count also the size of the queries. Furthermore we can impose various restrictions on the form of the queries. E.g. bounded depth queries would correspond to bounded depth Frege proofs.

A particularly interesting restriction is the restriction to *monotone formulas* which are formulas using only the connectives $\wedge$ and $\vee$. Since there are no nontrivial monotone tautologies, one has to consider proofs from assumptions. A lot of interesting tautologies can

be represented in this way, e.g. the most useful example – the Pigeon Hole Principle. (As we can take the conjunction of all assumptions, we can confine ourselves to the case of the single assumption.) In this case the game starts with Adversary claiming the assumptions to be true and the conclusion to be false. In circuit complexity the restriction to monotone circuits enabled to prove exponential lower bounds, while for nonmonotone circuits we still have only small linear lower bounds (for explicitly defined boolean functions). Thus we hope that also in propositional calculus the monotone case will be easier.

Proposition 2 suggests that there might be a similar relation between the monotone version of the game and a monotone version of propositional calculus. The most natural way to introduce the monotonicity restriction to propositional calculus is to use the sequent calculus with monotone formulas in the sequents in the whole proof. Thus rules for other connectives are forbidden. Part 1 of the above proof does not work in this case; although it can be made to work in the case of monotone tree-proofs. So Proposition 2 does apply to monotone tree-proofs and the monotone version of the game.

It is not clear, if we really need nonmonotone formulas for short proofs of monotone true sequents. Thus we have the following two questions.

Question 4. Can every (general) sequent proof of a monotone sequent be replaced by at most polynomially longer *monotone* proof?

Question 5. Can every monotone sequent proof of a monotone sequent be replaced by at most polynomially longer monotone *tree* proof?

It is conceivable that the answer to both questions is NO. Thus tree-like monotone proofs are an interesting class of proofs on which we can try lower bound methods. Let us stress that we do not have superpolynomial lower bounds even for such proofs. In Prover-Adversary game this means that the following is open.

Question 6. Are there monotone true sequents which cannot be proved in monotone Prover-Adversary game using $O(\log n)$ rounds, where n the size of a sequent?

3 An example of the adversary method

Let us consider the following formula t_n

$$\underbrace{\neg\neg\cdots\neg\neg}_{n}(p \vee \neg p).$$

Note that t_{2n} is always a tautology; this is a well-known example for which one can prove a linear lower bound on the number of steps in Frege proofs [?, 7]. We shall show an $\Omega(\log n)$ lower bound for the number of rounds in the game. This, of course, follows from the cited result and Proposition 2. Still the proof is interesting, because it is *different*, it is not just a translation. interesting reason. The direct translation only gives us a proof that Prover cannot win in r rounds for some $r = o(\log n)$. To get a winning strategy for Adversary in r rounds we have to refer to the finiteness of the game. Thus the direct translation does not give the winning strategy explicitly, so it is not a really "adversary argument".

Proposition 7. *Any proof of t_{2n} in the Prover-Adversary game requires $\log n$ rounds.*

Proof. For $m \geq 0$, let A_m be an assignment of truth values to formulas defined as follow. Let $\varphi = \varphi(t_{i_1}, \ldots, t_{i_k}, p, p_1, \ldots, p_q)$ be a formula, where the t_{i_j}'s are maximal, p stands for all other occurrences of p and $p_1, \ldots, p_q$ are all other variables. First assign some values (say 0) to p and $p_1, \ldots, p_q$. Then assign values to t_{i_j}'s as follows. If $i_j \geq m$, let $t_{i_j} \mapsto 1$, if i_j is odd and $t_{i_j} \mapsto 0$, if i_j is even. If $i_j < m$, then assign values conversely. Thus if $i_j \geq m$ we assign to t_{i_j} the incorrect value and otherwise we assign the correct value. Once the values of $t_{i_1}, \ldots, t_{i_k}, p, p_1, \ldots, p_q$ are set, evaluate the formula according to the rest of the connectives correctly.

Claim. Let $\varphi_1, \ldots, \varphi_l$ be the maximal proper subformulas of φ and suppose that the values assigned to $\varphi, \varphi_1, \ldots, \varphi_l$ according some A_m give an immediate contradiction. Then $\varphi = t_m$, (hence $l = 1$ and $\varphi_1 = t_{m-1}$).

This is easy, since if φ is not of the form t_i for some i, then every maximal t_j's is maximal also in some proper subformula of φ.

Now we can describe a strategy for Adversary. He will keep a certain set S of numbers between 0 and $2n$ in each round. He starts with S consisting of all numbers between 0 and $2n$. Suppose we are in a certain round with a set S and Prover asks formula φ. Then Adversary evaluates φ using all A_m's with $m \in S$. Then he chooses the value for φ which occurs most frequently and sets new S to consist of those m's for which he got this value. Thus the size of

S decreases at most by the factor 2. The set S has the property that the values of all queries up to this round equal to the values obtained by applying A_m to them for any $m \in S$. Hence, by the Claim, there cannot be an immediate contradiction in the answers of Adversary, if $S \geq 2$. So Adversary can be consistent at least for $\log n$ rounds. □

4 A nonconstructive lower bound

In this section we prove a slightly larger lower bound $\log n + \log\log n - O(\log\log\log n)$ on the number of rounds in the Prover-Adversary game. (Note that this is larger than previous bound only if we do not count the size of indices of variables.) We do not construct the formulas explicitly, but use a counting argument to show that they exist. Although counting arguments sometimes easily give exponential lower bounds in circuit complexity [**?**], it seems that for the propositional calculus we cannot get such strong bounds.
We consider the following formulas

$$s_{n,\pi} =_{df} \; p_{\pi(1)} \wedge \ldots \wedge p_{\pi(n)} \rightarrow p_1 \wedge \ldots \wedge p_n,$$

where π is a permutation of $\{1, \ldots, n\}$. The distribution of parentheses is not important; for definiteness let us assume that we group the conjuncts to the left. These formulas have been used by Orevkov [**?**] to prove a speedup from $\Omega(n \log n)$ to $O(n)$ of the sequence-like proofs vs. tree-like proofs (this speedup was rediscovered later by the authors, and we sketch its proof below). Theorem 8 does not follow from Orevkov's result since we do not have such a tight relation between tree-like proofs and the game.

Theorem 8. *There exists a sequence of permutations* $\{\pi_n\}_{n=1}^{\infty}$, π_n *a permutation of* $\{1, \ldots, n\}$, *such that any proof of* s_{n,π_n} *in the Prover-Adversary game requires* $\log n + \log\log n - O(\log\log\log n)$ *rounds.*

Proof. Let a winning strategy P of Prover be given. We can view P as a labeled binary tree where the nodes are labeled by the queries of Prover and the edges are labeled by the answers of Adversary. In particular, the root is labeled by the proved formula and has only one edge which is labeled by 0. For each branch there is a simple contradiction for some node labels.
The *skeleton* of P is defined to be the same tree, but with the node labels replaces by information about a simple contradiction for each branch. Namely, if $\varphi_1, \ldots, \varphi_k$, $\circ(\varphi_1, \ldots, \varphi_k)$ is a simple contradiction for a branch b, we add edges labeled by $1, \ldots, k$ pointing from

the leaf to the nodes on b which were labeled by $\varphi_1, \ldots, \varphi_k$ and an edge labeled by $\circ$ pointing to the node labeled by $\circ(\varphi_1, \ldots, \varphi_k)$.

Lemma 9. *Let S be the skeleton of some winning strategy for $s_{n,\pi}$, π any permutation of $\{1, \ldots, n\}$. Then S and n uniquely determine the permutation π.*

Proof. of Lemma. Let S and n be given. Define a unification problem as follows. Introduce a variable for each node of S and add an equation corresponding to a simple contradiction for each branch in S:

$$v \;=\; \circ(v_1, \ldots, v_k)$$

where v is the variable of the node to which an edge labeled by $\circ$ is pointing etc.
Let y be the variable corresponding to $s_{n,\pi}$. We take another variable x and add one more equation

$$y \;=\; x \to p_1 \wedge \ldots \wedge p_n.$$

Clearly this unification problem is determined solely by S and n. Consider the most general unifier of this problem and let the term ξ be the solution for x. We claim that ξ is actually $p_{\pi(1)} \wedge \ldots \wedge p_{\pi(n)}$. We know that this formula can be obtained from ξ by a substitution, as the proof whose skeleton S is, is a solution of the unification problem. If ξ was not equal to it, then there would be at least one p_i missing in it. Then, if we substitute, say a different variable for the free variables in ξ we get a proof of a non-tautology, which is a contradiction. Thus π is determined by the skeleton S and n. □

Proof. of Theorem 8. To prove the theorem it suffices to compare the number of skeletons of a given depth d (= number of rounds) and the number of permutation on n elements. W.l.o.g. we can assume that each branch has length d, thus we need only to count the number of possible markings of simple contradictions. If we have a basis B with at most k-ary connectives, then the number of possible situations on a branch of length d is $|B|d^{k+1}$. Hence the number of such skeletons is estimated by

$$(|B|d^{k+1})^{2^d} \;= 2^{2^d \log d},$$

while the number of permutations is $n! = 2^{\Omega(n \log n)}$. This gives the bound $d = \log n + \log\log n - O(\log\log\log n)$. □

Next we state and and give a quick sketch of a theorem originally proved by Orevkov [**?**] and later rediscovered by the authors. This gives a $\Omega(n \log n)$ lower bound on the length of *tree-like* Frege proofs of the tautologies $s_{n,\pi}$.

Theorem 10. *For every Frege system there exists a positive constant ε such that for every n there exists a permutation π of $\{1, \ldots, n\}$ such that every tree-like proof of $s_{n,\pi}$ has at least $\varepsilon n \log n$ steps.*

The proof of Theorem 10 is very similar to the proof of Theorem 8, and, in the setting of proofs, is a well-known technique due to Parikh [11]. For a Frege proof P we define *the skeleton* of P to be the labeled graph whose vertices correspond to the formulas, the label of a vertex v corresponding to a formula φ determines the rule by which φ was derived and the edges going into v determine from which formulas was φ derived. Furthermore the edges are ordered so that it is clear at which positions of the rule were the formulas used. Put otherwise, a skeleton contains all information about the proof except for the formulas. Similar to Lemma 9 above, we have:

Lemma 11. *Let S be the skeleton of some Frege proof of $s_{n,\pi}$. Then S and n uniquely determine the permutation π.*

The proof of Lemma 11 is similar to the proof of Lemma 9 and we leave it to the reader.

Proof. of Theorem 10. To prove the theorem it suffices to compare the number of tree-skeletons with a given number of vertices and the number of permutation on n elements. To estimate the number of skeletons we can use well-known estimates about the number of trees, but we can also estimate it easily directly. A tree-skeleton can be represented as a term where we have a function symbol for each rule and a single (constant) symbol c which we use for all leaves. Using Polish notation we can even avoid parentheses. Thus we can code tree-skeletons with $\leq L$ vertices by words of length L in an alphabet of size $r + 1$, where r is the number of rules of the Frege system. If all tautologies $s_{n,\pi}$ have proofs with at most L steps, then

$$(r + 1)^L \geq n!,$$

which gives $L = \Omega(n \log n)$. □

Theorems 8 and 10 are both proved by counting arguments. As a consequence, the stated lower bounds apply to randomly chosen permutations; however, we do not know any particular explicitly defined permutation for which the lower bounds hold.

5 A method based on boolean values

We shall discuss another method for proving lower bounds on the lengths of proofs. This method has been successfully applied in the case of proofs where the formulas have bounded depth [2]. (Here the restriction means that we use only the De Morgan basis and the number of alternations of different connectives is bounded by a constant; e.g. CNF's and DNF's are of depth ≤ 3.) Ajtai [1] and Riis [**?**] use in fact a different approach, an approach based on *forcing*, but their results can be interpreted using the boolean values method.

In model theory we use boolean values to prove independence results as follows. We take a suitable boolean algebra and assign suitable values to formulas. If a sentence gets value different from 1, then it is not provable, since we can collapse the boolean algebra to a two-element boolean algebra and get a model, where the sentence is false. In propositional calculus we are interested in lower bounds on the length of proofs of *tautologies*. A tautology gets value 1 in any boolean algebra, so we cannot use a single boolean algebra. Our approach is based on assigning boolean algebras to every small subset of a given set of formulas in a consistent way. An equivalent approach has been proposed by Krajíček [**?**], which is based on assignments in a single *partial* boolean algebra.

The concept of a homomorphism is defined for boolean algebras. We extend it to mappings of sets of formulas into boolean algebras. Namely, let a set of formulas L and a boolean algebra B be given. A mapping $\lambda : L \to B$ will be called a *homomorphism*, if it is consistent w.r.t. connectives. For instance

$$\lambda(\neg\varphi) = \neg_B \lambda(\varphi) \text{ if } \varphi, \neg\varphi \in L,$$

$$\lambda(\varphi \vee \psi) = \lambda(\varphi) \vee_B \lambda(\psi) \text{ if } \varphi, \psi, \varphi \vee \psi \in L.$$

We define the *degree* of a Frege system $\mathcal{F}$ as the maximal number of subformulas of a rule (or axiom scheme) of $\mathcal{F}$. E.g. the Modus Ponens rule has three subformulas φ, ψ and $\varphi \to \psi$, so $d \geq 3$.

Proposition 12. *Let a Frege system $\mathcal{F}$ of degree d be given, let τ be an arbitrary formula.*
Suppose that for every set of formulas Φ of size at most n which contains τ the following holds: (1) For each subset $S \subseteq \Phi$ of size

at most d we can find a boolean algebra B_S and a homomorphism $\lambda_S : S \to B_S$, and (2) for every pair T, S, with $T \subseteq S$, we can find an embedding $\kappa_{T,S} : B_T \to B_S$ such that the following diagram commutes

$$\begin{array}{ccc} T & \stackrel{\lambda_T}{\longrightarrow} & B_T \\ \downarrow id & & \downarrow \kappa_{T,S} \\ S & \stackrel{\lambda_S}{\longrightarrow} & B_S \end{array}$$

Furthermore we require that $\lambda_{\{\tau\}}(\tau) < 1$.
Then τ does not have a proof with $\leq n$ steps.

Proof. Let a proof $(\varphi_1, \ldots, \varphi_m), m \leq n$ of $\tau (= \varphi_m)$ be given. We shall show that the assumption of the proposition fails for $\Phi = \{\varphi_1, \ldots, \varphi_m\}$. Suppose that we have a system of homomorphisms as required in the proposition, except possibly for the last condition. We shall show that all $\varphi \in \Phi$ get $\lambda_{\{\varphi\}}(\varphi) = 1$, thus the last condition is not satisfied.
First observe that $B_{\{\varphi\}}$ is embedded in all B_S where $\varphi \in S$, hence $\lambda_S(\varphi) = 1$ for one S iff it holds for all such S. Let φ be an instance of a logical axiom $\psi(p_1, \ldots, p_k)$, i.e. $\varphi = \psi(\chi_1, \ldots, \chi_k)$ for some formulas $\chi_1, \ldots, \chi_k$. Let S be the set of formulas $\theta(\chi_1, \ldots, \chi_k)$, where θ runs over all subformulas of ψ. By the assumption, $|S|$ is at most the degree of the Frege system, hence we have a boolean algebra B_S and a homomorphism $\lambda_S : S \to B_S$. Since ψ is a tautology, it must get value 1 for any assignment of boolean values. Thus

$$\lambda_S(\varphi) = \lambda_S(\psi(\chi_1, \ldots, \chi_k)) = \psi(\lambda_S(\chi_1), \ldots, \lambda_S(\chi_k)) = 1.$$

Suppose that φ_i is obtained in the proof from some $\varphi_{j_1}, \ldots, \varphi_{j_k}$, $j_1, \ldots, j_k < i$ by a Frege rule, and suppose that $\varphi_{j_1}, \ldots, \varphi_{j_k}$ get all the value 1 in their algebras. Then applying the same argument as for an axiom (namely, a Frege rule is sound in any boolean algebra), we conclude that ϕ also gets the value 1. Thus, by induction, all formulas $\varphi_1, \ldots, \varphi_m$ get value 1. □

As an example, we shall describe the form of boolean algebras that one can use for proving a superpolynomial lower bound on the lengths of bounded depth proofs of the Pigeon Hole Principle, using the combinatorial arguments of Ajtai [1]. The Pigeon Hole Principle is the statement that there is no bijection between an $n+1$-element set D and an n-element set R. It is represented by the following formula

$$\bigvee_{i \neq j \in D,\ k \in R} (p_{ik} \wedge p_{jk}) \vee \bigvee_{i \neq j \in R,\ k \in D} (p_{ki} \wedge p_{kj}) \vee \bigvee_{i \in D} \bigwedge_{k \in R} \neg p_{ik} \vee \bigvee_{k \in R} \bigwedge_{i \in D} \neg p_{ik},$$

where p_{ij} determines whether the pair $\{i, j\}$ is in the alleged mapping. We think of truth assignments as bijections between D and R. There are no such real assignments, but in some cases we can still determine what would be the value of a formula under such assignments. For instance, PHP will get the value 0, since it asserts that there are no such assignments. In some cases we cannot decide the value of a formula for all such assignments, but we can decide it for all assignments which extend some *partial one-to one mapping* $g : D \to R$. In other cases it is not possible at all. The key combinatorial argument shows that for small sets Φ of bounded depth formulas there exists a partial assignments h (in fact a random h of suitable size) such that for each $\varphi \in \Phi$ its value can be determined by certain small extensions of h.

Let us forget about h. Then the statement is roughly this. There exists a constant size set $C \subseteq D \cup R$ such that the value of φ is decided by all g's whose support contains C. Now we take the boolean algebra B_C of all subsets of partial one-to one mappings g whose support contains C and which are minimal with this property (i.e. if g' is a proper subset of g, then its support does not cover C). The value of φ is the set of such g's which force φ to be true (the other g's force φ to be false). For a set $\varphi_1, \ldots, \varphi_k$ of formulas with the corresponding subsets $C_1, \ldots, C_k$, we take the boolean algebra $B_{C_1 \cup \ldots \cup C_k}$. If $C' \subseteq C$, then there exists a natural embedding of $B_{C'}$ into B_C. Thus we get the required set of homomorphisms.

References

1. Miklos Ajtai. The complexity of the pigeonhole principle. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, pages 346–355, Piscataway, New Jersey, 1988. IEEE Computer Society.
2. Paul Beame, Russell Impagliazzo, Jan Krajíček, Toniann Pitassi, Pavel Pudlák, and Alan Woods. Exponential lower bounds for the pigeonhole principle. In *Proceedings of the 24th Annual ACM Symposium on the Theory of Computing*, pages 200–221, New York, 1992. Association for Computing Machinery.
3. Samuel R. Buss. *Bounded Arithmetic.* Bibliopolis, Napoli, 1986. Revision of 1985 Princeton University Ph.D. thesis.
4. Stephen A. Cook. Feasibly constructive proofs and the propositional calculus. In *Proceedings of the Seventh Annual ACM Symposium on the Theory of Computing*, pages 83–97, New York, 1975. Association for Computing Machinery.
5. Stephen A. Cook and Robert A. Reckhow. On the lengths of proofs in the propositional calculus, preliminary version. In *Proceedings of the Sixth Annual ACM Symposium on the Theory of Computing*, pages 135–148, New York, 1974. Association for Computing Machinery.

6. Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic*, 44:36–50, 1979.
7. Jan Krajíček. Speed-up for propositional FregeFrege, G. systems via generalizations of proofs. *Commentationes Mathematicae Universitatis Carolinae*, 30:137–140, 1989.
8. Jan Krajíček. *Bounded Arithmetic, Propositional Logic and Complexity Theory*. Cambridge University Press, 1995.
9. Jan Krajíček and Pavel Pudlák. Propositional proof systems, the consistency of first-order theories and the complexity of computations. *Journal of Symbolic Logic*, 54:1063–1079, 1989.
10. Jan Krajíček and Pavel Pudlák. Quantified propositional calculi and fragments of bounded arithmetic. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 36:29–46, 1990.
11. R. Parikh. Some results on the lengths of proofs. *Transactions of the American Mathematical Society*, 177:29–36, 1973.

This article was processed using the LaTeX macro package with LLNCS style