Minimum Propositional Proof Length is NP-Hard to Linearly Approximate
Preliminary - Not for circulation

Michael Alekhnovich
Faculty of Mechanics & Mathematics
Moscow State University, Russia
michael@mail.dnttm.ru

Sam Buss*
Department of Mathematics
University of California, San Diego
sbuss@ucsd.edu

Shlomo Moran†
Department of Computer Science
Technion, Israel Institute of Technology
Haifa, Israel 32000
moran@cs.technion.ac.il

Toniann Pitassi‡
Computer Science Department
University of Arizona
Tucson, AZ 85721
toni@cs.arizona.edu

February 2, 1998

Abstract

We prove that the problem of determining the minimum propositional proof length is NP-hard to approximate within any constant factor. These results hold for all Frege systems, for all extended Frege systems, for resolution and Horn resolution, and for the sequent calculus and the cut-free sequent calculus. Also, if NP is not in $QP = DTIME(n^{\log^{O(1)} n})$, then it is impossible to approximate minimum propositional proof length within a factor of $2^{\log^{(1-\varepsilon)} n}$ for any $\varepsilon > 0$. We introduce the Monotone Minimum (Circuit) Satisfying Assignment problem and prove the same hardness results for Monotone Minimum (Circuit) Satisfying Assignment. All these

*Supported in part by NSF grant DMS-9503247 and grant INT-9600919/ME-103 from NSF and MŠMT (Czech Republic)

†Research supported by the Bernard Elkin Chair for Computer Science and by US-Israel grant 95-00238

‡Supported in part by NSF grant CCR-9457782, US-Israel BSF grant 95-00238, and grant INT-9600919/ME-103 from NSF and MŠMT (Czech Republic)

hardness of approximation results apply to proof length measured either by number of symbols or by number of inferences, for tree-like or dag-like proofs.

1 Introduction

This paper proves lower bounds on the hardness of finding short propositional proofs of a given tautology and on the hardness of finding short resolution refutations. When considering Frege proof systems, which are textbook-style proof systems for propositional logic, the problem can be stated precisely as the following optimization problem:

Minimum Length Frege Proof:
Instance: A propositional formula φ which is a tautology.
Solution: A Frege proof P of φ.
Objective function: The number of symbols in the proof P.

For a fixed Frege system $\mathcal{F}$, let $\min_{\mathcal{F}}(\varphi)$ denote the minimum number of symbols in an $\mathcal{F}$-proof of φ. An algorithm M is said to approximate the Minimum Length Frege Proof problem within factor α, if for all tautologies φ, $M(\varphi)$ produces a Frege proof of φ of length $\leq \alpha \cdot \min_{\mathcal{F}}(\varphi)$. (Here, α may be a constant or may be a function of the length of φ.)

We are interested only in *polynomial time* algorithms for solving this problem. However, there is a potential pitfall here since the shortest proof of a propositional formula could be substantially longer than the formula itself,[1] and in this situation, an algorithm with runtime bounded by a polynomial of the length of the input could not possibly produce a proof of the formula. In addition, it seems reasonable that a "feasible" algorithm which is searching for a proof of a given length ℓ should be allowed runtime polynomial in ℓ, even if the formula to be proved is substantially shorter than ℓ. Therefore we shall only discuss algorithms that are polynomial time in the length of the shortest proof (or refutation) of the input.

Note that an alternative approach would be to consider a similar problem, **Minimum Length Equivalent Frege Proof**, an instance of which is a Frege proof of some tautology φ, and the corresponding solutions are (preferably shorter) proofs of φ. While our results are all stated in terms

[1] Is is known that $NP \neq coNP$ implies that some tautologies require superpolynomially long Frege proofs.

of finding a short proof to a given tautology, they hold also for that latter version where the instance is a proof rather than a formula.

Yet a different approach could be studying algorithms which output the *size* (ie, number of symbols) of a short proof of the input formula, rather than the proof itself. In this case a polynomial time algorithm could be possible even if the size of the shortest proof is exponential in the size of the formula. We will observe that for this version, substantially stronger non-approximability results can be obtained for a variety of proof systems.

A related minimization problem concerns finding the shortest Frege proof when proof length is measured in terms of the number of steps, or lines, in the proof:

Minimum Step-Length Frege Proof:
Instance: A propositional formula φ which is a tautology.
Solution: A Frege proof P of φ.
Objective function: The number of steps in the proof P.

Resolution is a propositional proof system which is popular as a foundation for automated theorem provers. Since one is interested in finding resolution refutations quickly it is interesting to consider the following problem:

Minimum Length Resolution Refutation
Instance: An unsatisfiable set Γ of clauses.
Solution: A resolution refutation R of Γ.
Objective function: The number of inferences (steps) in R.

The main results of this paper state that a variety of minimum propositional proof length problems, including the Minimum Length Frege Proof, the Minimum Step-Length Frege Proof and the Minimum Length Resolution Refutation problems, cannot be approximated to within a constant factor by any polynomial time algorithm unless $P = NP$. Furthermore, for these proof systems and for every constant ϵ, the Minimum Length Proof problems cannot be approximated to within a factor of $\epsilon \ln n$ unless $NP \subseteq DTIME(n^{O(\log \log n)})$ or to within a factor of $2^{\log^{(1-\epsilon)} n}$ unless $NP \subseteq QP$, where QP, quasi-polynomial time, is defined to equal $DTIME(2^{(\log n)^{O(1)}})$. Our results apply to all Frege systems, to all extended Frege systems, to resolution, to Horn clause resolution, to the sequent calculus, and to the cut-free sequent calculus; in addition, they apply whether proofs are measured in terms of symbols or in terms of steps (inferences), and the apply to either dag-like or tree-like versions of all these systems.

We let $\mathcal{F} \overset{k}{\vdash} \varphi$ mean that φ has an $\mathcal{F}$-proof of $\leq k$-symbols. One of the first prior results about the hardness of finding optimal length of Frege proofs was the second author's result [9] that, for a particular choice of Frege system $\mathcal{F}_1$ with the language $\wedge$, $\vee$, $\neg$ and $\rightarrow$, there is no polynomial time algorithm which, on input a tautology φ and a $k > 0$, can decide whether $\mathcal{F}_1 \overset{k}{\vdash} \varphi$, unless P equals NP. This result however applies only to a particular Frege system, and not to general Frege systems. It also did not imply the hardness of approximating Minimum Length Frege Proofs to within a constant factor.

A second related result, which follows from the results of Krajíček and Pudlák [17], is that if the RSA cryptographic protocol is secure, then there is no polynomial time algorithm for approximating the Minimum Step-Length Frege Proof problem to within a polynomial.

Another closely related prior result is the striking connection between the (non)automatizability of Frege systems and the (non)feasibility of factoring integers that was recently discovered by Bonet-Pitassi-Raz [8]. A proof system T is said be *automatizable* provided there is an algorithm M and a polynomial p such that whenever $T \overset{n}{\vdash} \varphi$ holds, $M(\varphi)$ produces some T-proof of φ in time $p(n)$ (see [11]). Obviously the automatizability of Frege systems is closely related to the solution of the Minimum Length Frege Proof problem. Our theorems give a linear or quasi-linear lower bound on the automatizability of the Minimum Proof Length problem based on the assumption that $P \neq NP$ or that $NP \not\subseteq QP$. It has recently been shown by Bonet-Pitassi-Raz [8] that Frege systems are not automatizable unless Integer Factorization is in P. Their result provides stronger non-approximability conclusion, but requires assuming a much stronger complexity assumption.

For resolution, the first prior hardness result was Iwama-Miyano's proof in [15] that it is NP-hard to determine whether a set of clauses has a read-once refutation (which is necessarily of linear length). Subsequently, Iwama [14] proved that it is in NP-hard to find shortest resolution refutations; unlike us, he did not obtain an approximation ratio bounded away from 1.

In section 2, we introduce the Monotone Minimum Satisfying Assignment and Monotone Minimum Circuit Satisfying Assignment problems and discuss the relevant prior results about the hardness of approximating NP-optimization problems. Section 3 contains the main results about the hardness of approximating minimum length Frege proofs. Section 4 discusses the main results about the hardness of approximating minimum length resolution refutations. Section 5 briefly discusses the hardness of approximating shortest sequent calculus proofs and cut-free sequent calculus proofs. The proofs in sections 3 through 5 depend critically on reducing the Minimum Length Frege Proof problem to the Minimum Set Cover and Minimal Label Cover problems, via the Monotone Minimum (Circuit) Satisfying Assignment problem.

2 Monotone Minimum Satisfying Assignment

The section introduces the Monotone Minimum Satisfying Assignment problem and shows it is harder to approximate than the Minimum Set Cover problem and the Minimum Label Cover. (The latter is needed for proving hardness of approximation within a superlinear factor). The reader can find a general introduction to and survey of the hardness of approximation and of probabilistically checkable proofs in [3] and [2]. Recall that an A-reduction, as defined by [16], is a polynomial-time Karp-reduction which preserves the non-approximating ratio to within a constant factor.

Consider the following NP-optimization problems:

Monotone Minimum Satisfying Assignment:
Instance: A monotone formula $\varphi(x_1, \ldots, x_n)$ over the basis $\{\vee, \wedge\}$.
Solution: An assignment $\langle v_1, \ldots, v_n \rangle$ such that $\varphi(v_1, \ldots, v_n) = \top$.
Objective function: The number of v_i's which equal $\top$.

We henceforth let $\rho(\varphi)$ denote the value of the optimal solution for the Monotone Minimum Satisfying Assignment problem for φ i.e., the minimum number of variables v_i which must be set *True* to force φ to have value *True*.

We will also consider the Monotone Minimum Circuit Satisfying Assignment problem which is to find the minimum number of variables which must be set *True* to force a given monotone circuit over the basis $\{\wedge, \vee\}$ evaluate *True*. It does not matter whether we consider circuits with bounded fanin or unbounded fanin since they can simulate each other. It is apparent that Monotone Minimum Circuit Satisfying Assignment is at least as hard as Monotone Minimum Satisfying Assignment.

Recall the Minimum Hitting Set problem, which is:

Minimum Hitting Set:
Instance: A finite collection $\mathcal{S}$ of subsets of a finite set U.
Solution: A subset V of U that intersects every member of $\mathcal{S}$.
Objective function: The cardinality of V.

It is easy to see that Monotone Minimum Satisfying Assignment is at least as hard as Minimum Hitting Set: namely Minimum Hitting Set can be reduced (via an A-reduction) to the special case of Monotone Minimum Satisfying Assignment where the propositional formula is in conjunctive normal form. Namely, given $\mathcal{S}$ and U, identify members of U with propositional variables and form a CNF formula which has, for each set in $\mathcal{S}$, a conjunct containing exactly the members of that set.

Lund and Yannakakis [18] noted that Minimum Hitting Set is equivalent to Minimum Set Cover (under A-reductions). Furthermore, it is known that the problem of approximating Minimum Set Cover to within any constant factor is not in polynomial time unless $P = NP$ [4]. If one makes a stronger complexity assumption, then one can obtain a better non-approximability result for Minimum Set Cover; namely, Feige [13] has proved that Minimum Set Cover cannot be approximated to within a factor of $(1-\epsilon)\ln n$ unless $NP \subseteq DTIME(n^{O(\log\log n)})$.

In fact, we can get stronger results than the above reduction of Minimum Set Cover to Monotone Minimum Satisfying Assignment. There are two ways to see this: firstly, we can use a construction due to S. Arora [private communication] to reduce Monotone Minimum Satisfying Assignment to the Minimum Label Cover problem, or alternatively we can use a "self-improvement" property of the Monotone Minimum Satisfying Assignment problem to directly prove better non-approximation results. Both approaches prove that Monotone Minimum Satisfying Assignment cannot be approximated to within a factor of $2^{(\log n)^{1-\epsilon}}$ unless $NP \subseteq QP$. The advantage of the first approach is that it gives a sharper result, namely, a reduction of Minimum Label Cover to Monotone Minimum Satisfying Assignment for Π_4-formula. The second approach is more direct in that it avoids the use of Label Cover. We present both approaches next.

Minimum Label Cover: (see [2])

Instance: The input consists of: (i) a regular bipartite graph $G = (U, V, E)$, (ii) an integer N in unary, and (iii) for each edge $e \in E$, a partial function $\Pi_e : \{1, \ldots, N\} \to \{1, \ldots, N\}$ such that 1 is in the range of Π_e.

The integers in $\{1, \ldots, N\}$ are called *labels*. A *labeling* associates a nonempty set of labels with every vertex in U and V. A labeling *covers* an edge $e = (u, v)$ (where $u \in U$, $v \in V$) iff for every label ℓ assigned to v, there is some label t assigned to u such that $\Pi_e(t) = \ell$.

Solution: A labeling which covers all edges.

Objective function: The number of all labels assigned to vertices in U and V.

A Π_4-*formula* is a propositional formula which is written as an AND of OR's of AND's of OR's.

Theorem 1 (S. Arora) *There is an A-reduction from Minimum Label Cover to Monotone Minimum Satisfying Assignment such that the instances of Label Cover are mapped to* Π_4 *formulas.*

It was proved in [1] that Minimum Label Cover is not approximable within a $2^{\log^{(1-\epsilon)} n}$ factor unless $NP \subseteq QP$. An immediate corollary of Theorem 1 is that Monotone Minimum Satisfying Assignment enjoys the same hardness of approximation, even when restricted to Π_4-formulas.

Proof. Suppose we have an instance of Label Cover $(G, N, \{\Pi_e\})$. Let $m = |U|$ and $k = |V|$. For $1 \leq i \leq m$ and $1 \leq \ell \leq N$, let u_i^ℓ be a propositional variable with the intended meaning that $u_i^\ell = \top$ iff ℓ is one of the labels assigned to vertex $i \in U$. Likewise, for $1 \leq j \leq m$ and $1 \leq \ell \leq N$ let v_j^ℓ denote the condition that ℓ is one of the labels assigned to vertex $j \in V$. We shall construct a formula φ involving the variables u_i^ℓ and v_j^ℓ so that the minimal satisfying assignments for φ are precisely those truth assignments which correspond to minimal weight labelings which cover all edges. We define φ to be

$$\bigwedge_{j=1}^{k} \bigvee_{\ell=1}^{N} \left(v_j^\ell \wedge \bigwedge_{i|(i,j)\in E} \; \bigvee_{t|\Pi_e(t)=\ell} u_i^t \right)$$

The formula φ clearly is a monotone Π_4-formula and has size polynomial in N, m, k. It is easy to verify that any minimum satisfying assignment for φ corresponds to a labeling which covers all edges and which has a minimum number of labels. $\square$

We next present a second proof that Monotone Minimum Satisfying Assignment is hard to approximate to within a $2^{\log^{(1-\varepsilon)} n}$ factor; although it does not yield the hardness of the Π_4-formula case, it may be independent interest since because of its use of "self-improvement" and of non-constant depth propositional formulas.

Lemma 2 (Self-improvement property) *For any formulas φ_1, φ_2 there is a formula $\varphi_1 * \varphi_2$ such that $\rho(\varphi_1 * \varphi_2) = \rho(\varphi_1) \cdot \rho(\varphi_2)$ and $|\varphi_1 * \varphi_2| \le |\varphi_1| \cdot |\varphi_2|$.*

Proof. Let $\varphi_1 = \varphi_1(x_1, \dots, x_k)$, $\varphi_2 = \varphi_2(x_1, \dots, x_\ell)$. Let $(\varphi_1 * \varphi_2)$ be a composition of φ_1 and φ_2:

$$(\varphi_1 * \varphi_2)(x_1^1, \dots, x_1^\ell, \dots, x_k^1, \dots, x_k^\ell) = \varphi_1(\varphi_2(x_1^1, \dots, x_1^\ell), \dots, \varphi_2(x_k^1, \dots, x_k^\ell))$$

The proof follows. $\square$

Theorem 3 *There is no polynomial time algorithm which approximates Monotone Minimum Satisfying Assignment within a factor of $2^{(\log n)^{1-\epsilon}}$ for any $\epsilon > 0$ unless $NP \subseteq QP$.*

Proof. Suppose we have a formula ψ which is an instance of Satisfiability. Let $n = |\psi|$. By the reduction to Minimum Hitting Set, there exists a polynomial time reduction from Satisfiability to approximation of Monotone Minimum Satisfying Assignment within factor 2 [4], i.e., a function $f : \psi \mapsto \varphi, |\varphi| = n^{O(1)}$ such that if ψ is satisfiable then $\rho(f(\psi)) < C_1(n)$ and otherwise $\rho(f(\psi)) > C_2(n)$ with gap $g(n) = \frac{C_2(n)}{C_1(n)} > 2$.

Applying self-improvement k-times to φ, we increase the gap g to 2^k. This gives the formula $F(\psi) = f(\psi) * f(\psi) * \dots * f(\psi)$ (k-times), which has length $N = |F(\psi)| < |\varphi|^k$. If ψ is satisfiable than $\rho(F(\psi)) < C_1^k(n)$ otherwise $\rho(F(\psi)) > C_2^k(n)$, the new gap is at least 2^k.

If we take $k(n) = (\log n)^c$, we get

$$N = |\varphi|^{k(n)} \le n^{(\log n)^c}, \quad \text{and} \quad \log n \ge (\log N)^{\frac{1}{c+1}},$$

so,

$$g \ge 2^{k(n)} \ge 2^{(\log N)^{\frac{c}{c+1}}}.$$

Suppose we have an algorithm A which approximates Monotone Minimum Satisfying Assignment to within the factor $2^{(\log n)^{1-\epsilon}}$. Take c large enough so that that $c/(c+1) > 1 - \epsilon$. The function F is computable in QP, and applying algorithm A to the formula $F(\psi)$ solves determines the satisfiability of ψ.

That completes the proof of Theorem 3. $\square$

To summarize the above material, we have the following theorem:

Theorem 4

(a) *If $P \neq NP$, then there is no polynomial time algorithm which can approximate Monotone Minimum Satisfying Assignment (and hence Monotone Minimum Circuit Satisfying Assignment) to within a constant factor.*

(b) *If $NP \not\subseteq DTIME(n^{O(\log\log n)})$, then Monotone Minimum Satisfying Assignment (and Monotone Minimum Circuit Satisfying Assignment) cannot be approximated to within a factor of $(1-\epsilon)\ln n$ where n equals the number of distinct variables.*

(c) *If $NP \not\subseteq QP$, then there is no polynomial time algorithm which can approximate Monotone Minimum Satisfying Assignment (or Monotone Minimum Circuit Satisfying Assignment) to within a factor of $2^{\log^{(1-\epsilon)} n}$.*

In the next sections we reduce the Monotone Minimum Circuit Satisfying Assignment problem first to the Minimum Length Frege Proof problem and then to other problems on minimum proof length. This will establish the same hardness of approximation results for the Minimum Length Frege Proof problem and similarly for the other proof length optimization problems.

Open question. Is it possible to improve the non-approximation factor for Monotone Minimum Satisfying Assignment or Monotone Minimum Circuit Satisfying Assignment, or prove their hardness using just $P \neq NP$ as a complexity theory hypothesis?

In fact the known NP-hardness of Monotone Minimum Satisfying Assignment concerns Π_2 (CNF) formulae and Quasi NP-hardness uses Π_4 formulae. But in general the formula or circuit can have unbounded depth and thus it a priori has richer expressive abilities. Hence there could be some chance to prove its hardness by some other way without improving the corresponding factor of Label Cover, perhaps by using some extension of the self-improvement property.

3 Main Results for Frege Systems

Our first main results are that it is hard to approximate the length of the shortest Frege proof of a given tautology. In sections 3.2 and 3.3, we'll treat the case where proof length is measured in terms of the number of symbols in the proof; we leave the case where proof length is measured in terms of number of lines to section 3.4.

Hardness Theorem 5

(a) *If $P \neq NP$, then there is no polynomial time algorithm which can approximate Minimum Length Frege Proof to within a constant factor.*

(b) *If $NP \not\subseteq QP$, then there is no polynomial time algorithm which can approximate Minimum Length Frege Proof to within a factor of* $2^{\log^{(1-\varepsilon)} n}$ *for any ε.*

This theorem holds whether Frege proofs are taken to be tree-like or dag-like.

The proof of Theorem 5 will give a reduction of the Monotone Minimum (Circuit) Satisfying Assignment problem to the Minimum Length Frege Proof problem. Thus any hardness results for the Monotone Minimum Satisfying Assignment or Monotone Minimum Circuit Satisfying Assignment problem immediately also apply to the Minimum Length Frege proof problem. Thus, for instance, Theorem 4(b) implies that for some constant $\delta > 0$, the Minimum Length Frege Proof problem cannot be approximated to within a factor of $\delta \log n$ unless $NP \subseteq DTIME(n^{O(\log\log n)})$. Similar comments apply to the rest of our theorems on the hardness of approximating proof length.

3.1 Preliminaries

Frege proof systems are proof systems for propositional logic. A Frege proof system $\mathcal{F}$ is specified by its language L and a finite set of inference and axiom schemes. The language L is a finite set of Boolean connectives, which is complete in the sense that any Boolean function can be represented by an L-formula. The permissible inferences are specified schematically as inferences

$$\frac{A_1 \quad A_2 \quad \cdots \quad A_k}{B}$$

which indicates that for any substitution σ of formulas for variables, $B\sigma$ may be inferred from the formulas $A_1\sigma, \ldots, A_k\sigma$. We allow $k = 0$ in the above scheme, which corresponds to axioms. Finally, the Frege proof system must be implicationally complete, i.e., if $A_1, \ldots, A_k \models B$ then there is a derivation of B from the assumptions $A_1, \ldots, A_k$ using the inferences of $\mathcal{F}$.

We define the size or length, $|C|$, of a formula C to equal the number of symbols in C, where each occurence of a variable or a connective is counted as a symbol. Likewise, if P is a Frege proof, then the symbol size of P, $|P|$, equals the number of symbols in P. If $\mathcal{F}_1$ is a Frege system, then $\mathcal{F}_1 \vdash \varphi$ means that there is an $\mathcal{F}_1$ proof of φ. The *symbol size* of a Frege proof is the total number of symbols in the proof. The *step-length* (or length) of a Frege proof is the number of lines in the proof. $\mathcal{F}_1 \overset{n}{\vdash} \varphi$ means that there exists an $\mathcal{F}_1$ proof P such that $|P| \leq n$.

Typical examples of Frege system includes the 'textbook systems' which use the language $\{\wedge, \vee, \neg, \rightarrow\}$ and have a finite set of axiom schemes and have modus ponens as their only other rule of inference. Of course there are many possible such textbook systems since there are many choices for the axiom schemes; however, they are all essentially equivalent in terms of proof length. Indeed the following holds:

Theorem 6 ([12, 19, 20]) *If $\mathcal{F}_1$ and $\mathcal{F}_2$ are Frege systems with the same language, then they linearly simulate each other; i.e., for all φ, if $\mathcal{F}_1 \overset{n}{\vdash} \varphi$ then $\mathcal{F}_2 \overset{O(n)}{\vdash} \varphi$, and vice-versa.*

For Frege proof systems in differing languages, it is known that any two Frege systems $\mathcal{F}_1$ and $\mathcal{F}_2$ *p*-simulate each other, i.e., that any $\mathcal{F}_1$-proof can be translated into an $\mathcal{F}_2$-proof in polynomial time and vice-versa; see [12, 19] for precise definitions and proofs of this.

Extended Frege proof systems are propositional proof systems which allow the introduction of abbreviations of formulas on the fly. It is conjectured that the minimal symbol size for extended Frege proofs can sometimes be exponentially smaller than the corresponding minimal Frege proof; however, this is still open.

We next define the notion of "active" formulas in a proof, which will be useful for proving lower bounds on the lengths of proofs. Recall that an inference in a proof must be a substitution instance of an axiom scheme, i.e., each inference must be of the form

$$\frac{A_1\sigma \quad \cdots \quad A_k\sigma}{A_{k+1}\sigma} \quad \text{(I)}$$

Consider a particular occurrence of a formula C as a subformula of a formula A_i in the inference (I). If the principal connective of C is present already in the formula A_i, then we say C is *active* w.r.t. the inference (I). Otherwise, C occurs as a (not necessarily proper) subformula of $x\sigma$ for some variable x, and C is not active w.r.t. inference (I).

If a formula C has some occurrence in a proof P which is active with respect to some inference of P, then C is said to be *active* in P. (The terminology is potentially confusing: it is important to note that an active formula of P may never occur as a formula in the proof P, but instead only as a subformula of formulas in P.)

The next theorem lets us obtain a lower bound on the length of P, $|P|$, in terms of the lengths of the active formulas of P.

Theorem 7 (see [10]) *Let $\mathcal{F}$ be a Frege proof system. There is a constant ϵ such that if P is a Frege proof and we let φ range over active formula-occurrences in P, then*

$$|P| \;\geq\; \epsilon \cdot \sum_{\varphi} |\varphi|$$

Proof. A formula can be viewed as a tree with nodes labeled by connectives from L. The *depth* of a formula is defined to equal the height of this tree, namely the maximum number of connectives along any branch of the tree. Let d equal the maximum depth of the depths of the formulas which occur in the inference schemes of $\mathcal{F}$, and set $\epsilon = (1/d)$. Clearly, any active occurrence of a formula in P must have its principal connective at distance at $d-1$ from the root of the formula's tree. Thus, any given single symbol a occurring in P can occur inside at most d active occurrences of subformulas. From this, we immediately get

$$d \cdot |P| \;\geq\; \sum_{\varphi} |\varphi|$$

and the theorem follows immediately. □

As mentioned earlier, the *step-length* of a proof P is equal to the number of steps or inferences (counting axioms as nullary inferences) in the proof. There is a linear relationship between the number of formulas active in P and the step length of P; namely,

Theorem 8 *Let $\mathcal{F}$ be a Frege proof system. Then there is a constant ϵ so that if P is a proof and m is the number of distinct active formulas in P, then the step-length of P is $\geq \epsilon m$.*

Proof. We let α equal the maximum number of subformulas that can be active in any given formula in P. For instance, if every inference scheme has depth bounded by d and r is the maximum arity of connectives in the language of F, then $\alpha = \sum_{i=0}^{d} r^i$ works. Clearly Theorem 8 is true with $\epsilon = 1/\alpha$. □

It is an interesting (albeit trivial) observation that if no formula is repeated in P, then the number of steps in P is also linearly upper bounded by the number of distinct formulas active in P.

3.2 Frege Systems with language $\{\wedge, \vee, \neg, \to\}$

We shall first prove our main theorem for Frege proof systems which have language $\wedge$, $\vee$, $\neg$ and $\to$. This will introduce and motivate the proof in the next section for general Frege proof systems.

Thus, consider an arbitrary Frege proof system $\mathcal{F}_1$ over the above set of connectives.

To prove Theorem 5 for $\mathcal{F}_1$, it will suffice to give a reduction of the Monotone Minimum Satisfying Assignment problem to the Minimum Length Frege Proof problem for $\mathcal{F}_1$. We will start with a construction from [9]:

Definition Let $p_0, p_1, \ldots$ be propositional variables. Let $\perp$ be the contradiction $p_0 \wedge \neg p_0$, and let τ_i^0 be the tautology $(\neg p_i \vee p_i)$. Define τ_i^ℓ to be $(\perp \vee (\perp \vee \cdots (\perp \vee \tau_i^0) \cdots))$, where there are ℓ occurrences of $\perp$. Note that $|\tau_i^l| = O(\ell)$.

The next two lemmas state that the proof of τ_i^ℓ is big for large ℓ, and that a derivation of τ_i^ℓ must essentially derive first all τ_i^k for all $k < \ell$.

Lemma 9 ([9]) *τ_i^ℓ has a tree-like $\mathcal{F}_1$ proof of length $O(\ell^2)$.*

This lemma is proved by noting that $\mathcal{F}_1$ can derive successively τ_i^0, τ_i^1, τ_i^2, etc., until τ_i^ℓ is derived.

Definition Let ψ be a formula and consider a particular τ_i^ℓ. We define $\psi/(\tau_i^\ell)$ to be the formula obtained from ψ by replacing every occurrence of τ_i^ℓ as a subformula of ψ with the formula $\perp$. Note, that if $\ell_1 < \ell_2$ then $\tau_i^{\ell_2}/(\tau_i^{\ell_1})$ is not a tautology anymore.

If P is a proof, then we define $P/(\tau_i^\ell)$ be a sequence of formulas obtained by replacing every ψ in P with $\psi/(\tau_i^\ell)$. Note that $P/(\tau_i^\ell)$ will not, in general, be a valid proof.

Lemma 10 *Let P be a proof of ψ and suppose that the formula τ_i^ℓ is not active in P. Then, $\psi/(\tau_i^\ell)$ is a tautology.*

The proof of Lemma 10 is immediate by the fact that if τ_i^ℓ is not active in P, then $P/(\tau_i^\ell)$ is identical to P, except that it may use a different substitution of formulas for variables, and hence it is still a valid proof. $\Box$

Finally we need a lemma that we will use to upper bound the lengths of Frege proofs which are monotone Boolean combinations of formulas of the form τ_i^ℓ.

Lemma 11 *Let $\varphi(x_1, \ldots, x_k)$ be a formula of size $|\varphi| = n$ over the basis $\{\vee, \wedge\}$. Suppose $\langle v_j \rangle$ is a satisfying assignment and that I is the index set of all v_i such that $v_i = \top$. Then, for every set of formulas $\psi_1, \psi_2, \ldots, \psi_k$ we have a tree-like (and hence dag-like) proof of*

$$\left(\bigwedge_{i \in I} \psi_i\right) \rightarrow \varphi(\psi_1, \ldots, \psi_k)$$

of length $O(M \cdot k^3 + M \cdot n^2) = O(M \cdot n^3)$,

Proof. By a straightforward induction on the size of φ. For the base case note that $\bigwedge_i \psi_i \rightarrow \psi_{i_0}$ has a proof of length $O(M \cdot k^2)$; for the induction step observe that in any Frege system over the language $\{\wedge, \vee, \neg, \rightarrow\}$, we have that $\{x \rightarrow y, x \rightarrow z\} \stackrel{O(1)}{\vdash} (x \rightarrow (y \wedge z))$, and $\{x \rightarrow y\} \stackrel{O(1)}{\vdash} (x \rightarrow (y \vee z))$. $\Box$

To prove Theorem 5 for $\mathcal{F}_1$, it suffices to prove:

Theorem 12 *There is an A-reduction from the Monotone Minimum Satisfying Assignment problem to the Minimum Length Frege Proof problem for $\mathcal{F}_1$. This reduction works for both the dag-like and the tree-like forms of $\mathcal{F}_1$.*

Proof. Recall that an A-reduction is a polynomial time Karp-reduction which preserves the non-approximating ratio to within a constant factor.

Suppose we have a formula $\varphi(x_1, \ldots, x_k)$ of length $n = |\varphi|$ over the basis $\{\vee, \wedge\}$, which is an instance of Monotone Minimum Satisfying Assignment. Take m sufficiently large compared to n, say $m = n^4$, and consider the following tautology:

$$\psi \;=\; \varphi(\tau_1^m, \tau_2^m, \ldots, \tau_k^m).$$

(a) Suppose that the optimal solution to the Monotone Minimum Satisfying Assignment problem for φ has $\rho(\varphi) \leq p_1$. We construct a tree-like (and hence dag-like) $\mathcal{F}_1$-proof of ψ of length $O(p_1 \cdot m^2) + O(m \cdot n^2) = O(p_1 \cdot m^2)$

as follows: First construct a tree-like proof of $\bigwedge_{i\in I}\tau_i^m \to \varphi$ of length $O(m\cdot n^3)$ using Lemma 11, then prove each τ_i^m for $i \in I$ with a proof of length $O(p_1\cdot m^2)$ by Lemma 9, and then prove the conjunction $\bigwedge_{i\in I}\tau_i^m$ of these formulas and applies modus ponens - which adds $O(mn)$ symbols to the proof.

(b) Suppose that $\rho(\varphi) \geq p_2$. Let P be some $\mathcal{F}_1$-proof of ψ. Let I be the index set of all i such that τ_i^ℓ is active in P for all $0 \leq \ell \leq m$. For $j \notin I$, choose j_r so that $\tau_j^{j_r}$ is not active in P. By Lemma 10 we have that, after replacing all $\tau_j^{j_r}$ with $\perp$ for all $j \notin I$, the formula ψ remains a tautology. Hence the formula φ is satisfied by the truth assignment corresponding to the characteristic function of I, hence $|I| \geq p_2$ by choice of p_2. Thus $|P| = \Omega(p_2 \cdot m^2)$, By Theorem 7 and the fact that the total length of the formulas τ_i^ℓ, for $i \in I,\ 0 \leq \ell \leq m$, is $\Omega(p_2 \cdot m^2)$.

(a) and (b) establish that $\varphi \mapsto \psi$ is an A-reduction; namely, it is easy to see that the constructed reduction transforms a sufficiently large constant gap g between hard and easy instances of Monotone Minimum Satisfying Assignment into a constant gap g_ε in Minimum Length Frege Proof, hence if Minimum Length Frege Proof is approximable with some constant then Monotone Minimum Satisfying Assignment is approximable too. This proves Theorem 12 as well as Theorem 5 for the case of Frege systems with language $\{\neg, \wedge, \vee, \to\}$. □

The following theorem shows that Frege proof systems, even tree-like ones, can deal in a certain sense even with monotone *circuits*.

Theorem 13 *There is an A-reduction from the Monotone Minimum Circuit Satisfying Assignment problem to the Minimum Length Frege Proof problem for $\mathcal{F}_1$. This is true for both tree-like and dag-like Frege systems.*

Proof. We present a reduction which works for both dag-like and tree-like Frege systems.

Suppose we have circuit C with inputs $x_1, \ldots, x_k$ given as a set of instructions $x_i := x_{j_{i,1}} OP x_{j_{i,2}}$ for $i \in \{k+1, \ldots, n\}$, where OP is $\wedge$ or $\vee$, where $j_{i,1}, j_{i,2} < i$ and where x_n is the output node. Construct a tautology

$$\psi = \left(\bigwedge_{i=k+1}^{n} ((x_{j_{i,1}} OP x_{j_{i,2}}) \to x_i) \right) \to x_n,$$

where $x_1, \ldots, x_k$ are replaced with $\tau_1^m, \ldots, \tau_k^m$. As before it will be sufficient to take m large enough, ie $m = n^4$.

The lower bound for the proof of ψ is absolutely identical to that of Theorem 12, namely every proof of ψ has length at least $\rho \cdot m^2$ where $\rho = \rho(C)$. Now we estimate an upper bound, by constructing a tree-like (and hence dag-like) proof of length $O(\rho m^2 + mn^3) = O(\rho m^2)$.

Suppose $C(v_1, \ldots, v_k) = \top$, and let $I = \{i : v_i = \top\}$. Suppose $\rho = |I|$ variables have value $\top$. We need to find a tree-like proof of ψ of length $O(\rho m^2 + m \cdot n^3)$. Let $r_1, r_2, \ldots, r_s = n$ be the increasing sequence of indices such that $x_{r_1}, x_{r_2}, \ldots, x_{r_s}$ are made true by the truth assignment $\vec{v}$. Then, it is straightforward to construct a tree-like Frege proof of the formulas

$$\bigwedge_{i \in I} x_i \quad \to \quad \left(\left(\bigwedge_{i=k+1}^{n} ((x_{j_{i,1}} OP x_{j_{i,2}}) \to x_i) \right) \to \bigwedge_{i=1}^{\ell} x_{r_i} \right)$$

which proceeds by proving these successively with $\ell = 1, 2, 3, \ldots, s$. Since the conjunctions all have $O(n)$ inputs and since the formulas have symbol-length $O(m \cdot n)$, the proof has symbol-length $O(m \cdot n^3)$. (If one is careful and uses balanced conjunctions [7], the proof can have length $O(m \cdot n^2 \log n)$.) Finally there is a tree-like proof of $\bigwedge_{i \in I} x_i$ of length $O(\rho \cdot m^2)$ since I has cardinality ρ. Thus, with a final modus ponens we get $(\bigwedge_{i=k+1}^{n} ((x_{j_{i,1}} OP x_{j_{i,2}}) \to x_i)) \to \bigwedge_{i=1}^{\ell} x_{r_i}$, from which we get ψ with additional $O(m \cdot n^2)$ symbols. In total, we get a tree-like proof of ψ of length $O(\rho m^2 + m \cdot n^3)$. □

3.3 General Frege proof systems

We now take up the proof of Theorem 5 for general Frege proof systems which use an arbitrary (complete) finite language of propositional connectives. Fix some such proof system $\mathcal{F}$, with language L.

We wish to construct L-formulas $\tau_i^{\ell,L}$ which are analogues of the formulas τ_i^{ℓ}. To do this with similar size bounds, we need the following lemma:

Lemma 14 (Reckhow [19]) *There are L-formulas $NOT(x,z)$, $AND(x,y,z)$, $OR(x,y,z)$, and $IMP(x,y,z)$ such that*

(1) *$NOT(x,z)$ contains one occurrence of x, and $AND(x,y,z)$, $OR(x,y,z)$, and $IMP(x,y,z)$ contain exactly one occurrence of each of x and y.*

(2) *The four formulas represent the Boolean functions $\neg x$, $(x \wedge y)$, $(x \vee y)$ and $(x \to y)$; in particular, the truth values of the formulas are independent of the truth value of z.*

Proof. The side variable z acts merely as a placeholder whose truth value is irrelevant: in fact, if the language L contains a constant symbol, then the use of the variable z is unnecessary. We shall assume that the constant symbols $\top$ and $\bot$ are included in L; this may be assumed without loss of generality since the symbols $\top$ and $\bot$ may be replaced everywhere by L-formulas equivalent to the formulas $(z \vee \neg z)$ and $(z \wedge \neg z)$, respectively.

Let N be an L-formula containing only the variable x which represents the propositional function $\neg x$; N exists since L is a complete set of connectives. If N contains n occurrences of x, we write $N = N(x, x, \ldots, x)$ with each occurrence of x separately indicated. Let $\top^i$ denote a vector of i occurrences of $\top$, and define $\bot^i$ similarly. By choice of N, $N(\top^n)$ has value *False* and $N(\bot^n)$ has value *True*. Therefore, there is some $0 \leq i < n$ such that $N(\top^i, \bot^{n-i})$ has value *True* and $N(\top^{i+1}, \bot^{n-i-1})$ has value *False*. Thus, we can take $NOT(x)$ to be the formula $N(\top^i, x, \bot^{n-i-1})$.

To prove the remainder of the lemma, it will suffice to find an L-formula $X(x, y)$ which has one occurrence of each of x and y, and which has appearing in its truth table either three values *True* and one value *False*, or three values *False* and one value *True*. (Note that conjunction, disjunction and implication are three of the eight propositional functions whose truth table has this property.) This will suffice since *AND*, *OR* and *IMP* can be readily defined from such a formula X and from *NOT*.

Let A be an L-formula containing only the variables x and y which represents the propositional function $(x \wedge y)$. Assume $A = A(x, \ldots, x, y, \ldots, y)$ has m occurrences of x and n occurrences of y, each occurrence separately indicated. Define $A_{i,\top}$ to be the formula $A(\top^i, \bot^{m-i}, \top^n)$ and $A_{i,\bot}$ to be $A(\top^i, \bot^{m-i}, \bot^n)$. Now $A_{m,\top}$ and $A_{m,\bot}$ have different truth values, and $A_{0,\top}$ and $A_{0,\bot}$ have the same truth value. Clearly, there is a value $0 \leq i < m$ so that $A_{i,\top}$ and $A_{i,\bot}$ have the same truth value, but so that $A_{i+1,\top}$ and $A_{i+1,\bot}$ have different truth values. Fix such an i and let $B = B(x, y, \ldots, y)$ be the formula $A(\top^i, x, \bot^{m-i-1}, y, \ldots, y)$. Note that B has one occurrence of x and n occurrences of y, each indicated separately. Let $B_i(x)$ be the formula $B(x, \top^i, \bot^{n-i})$. From the definition of B, the multiset of the four truth values of $B_0(\top)$, $B_0(\bot)$, $B_n(\top)$ and $B_n(\bot)$ contains either three *Trues* and one *False*, or one *True* and three *Falses*. Therefore, there is some value $0 \leq j < n$ so that the multiset of the truth values of $B_j(\top)$, $B_j(\bot)$, $B_{j+1}(\top)$ and $B_{j+1}(\bot)$ enjoys the same property. Letting $X(x, y)$ be the formula $B(x, \top^j, y, \bot^{n-j-1})$ gives the desired formula. $\square$

For simplicity of notation, we shall henceforth suppress mentioning the occurrences of the side variable z.

Definition If φ is a formula over the basis $\{\neg, \wedge, \vee, \rightarrow\}$, then its *$L$-translation*, φ^L, is the L-formula obtained by replacing the connectives $\neg$, $\wedge$, $\vee$, and $\rightarrow$ with the formulas *NOT*, *AND*, *OR* and *IMP* in the obvious way. Because of the condition that x and y occur at most once in the formulas *NOT*, *AND*, *OR* and *IMP*, the size $|\varphi^L|$ of φ^L is $O(|\varphi|)$.

We write $\tau_i^{\ell,L}$ to denote $(\tau_i^\ell)^L$; thus $|\tau_i^{\ell,L}| = O(\ell)$.

The next lemma will be used to give upper bounds on the lengths of $\mathcal{F}$-proofs and is the analogue of Lemmas 9 and 11.

Lemma 15
(1) *$\tau_i^{\ell,L}$ has an $\mathcal{F}$-proof of length $O(\ell^2)$.*
(2) *Let $\varphi(x_1, \ldots, x_k)$ be the L-translation of an $\{\wedge, \vee\}$-formula. Suppose $\langle v_j \rangle$ is a satisfying assignment and that $I = \{i : v_i = \top\}$. Then, for every set of formulas $\psi_1, \psi_2, \ldots, \psi_k$, there is a tree-like $\mathcal{F}$-proof of*

$$\left(\bigwedge_{i\in I} \psi_i\right)^L \rightarrow \varphi(\psi_1, \ldots, \psi_k)$$

of length $O(M \cdot k^2 \cdot n + M \cdot n^2) = O(M \cdot n^3)$, where $n = |\varphi|$ and M is the maximal size of ψ_i.

Theorem 16 *There is an A-reduction from the Monotone Minimum Satisfying Assignment problem to the Minimum Length Frege Proof problem for $\mathcal{F}$. This reduction works both for tree-like $\mathcal{F}$-proofs and dag-like $\mathcal{F}$-proofs.*

Proof. The proof is similar to the proof of Theorem 12. Suppose we have formula $\varphi(x_1, \ldots, x_k)$ of size $n = |\varphi|$ over the basis $\{\vee, \wedge\}$, which is an instance of Monotone Minimum Satisfying Assignment. Take m sufficiently large compared to n, say $m = n^4$, and consider the tautology

$$\psi \;=\; \varphi^L(\tau_1^m, \tau_2^m, \ldots, \tau_k^m).$$

(a) Suppose that the optimal solution to the Monotone Minimum Satisfying Assignment problem for φ has $\rho(\varphi) \leq p_1$. We have $|\varphi^L| = O(|\varphi|)$ so, by Lemmas 9 and 15, ψ has a (tree-like) $\mathcal{F}$-proof of length $O(p_1 \cdot m^2) + O(m \cdot n^3) = O(p_1 \cdot m^2)$.

(b) Suppose that $\rho(\varphi) \geq p_2$. Let P be some $\mathcal{F}$-proof of ψ. Let I be the index set of all i such that $\tau_i^{\ell,L}$ is active in P for all $0 \leq \ell \leq m$. For $j \notin I$, choose j_r so that $\tau_j^{j_r,L}$ is not active in P. By Lemma 10 we have that, after replacing all $\tau_j^{j_r,L}$ with $\perp$ for all $j \notin I$, the formula ψ remains a tautology. Hence the formula φ is satisfied by the truth assignment corresponding to the characteristic function of I, hence $|I| \geq p_2$. The total of the lengths of the active formulas $\tau_i^{\ell,L}$, $i \in I$, is greater than $p_2 \cdot m^2$; therefore, by Theorem 7, $|P| = \Omega(p_2 \cdot m^2)$.

(a) and (b) establish that $\varphi \mapsto \psi$ is an A-reduction. This proves Theorems 16 and 5. □

One can show that this reduction can be applied to monotone circuits as well:

Theorem 17 *There is an A-reduction from the Monotone Minimum Circuit Satisfying Assignment problem to the Minimum Length Frege Proof problem for $\mathcal{F}$. This is true for both the tree-like and dag-like versions of $\mathcal{F}$.*

The proof of this theorem is analogous to that of Theorem 13.

3.4 Hardness of approximating step-length

The previous section discussed the problem of minimizing the number of *symbols* in a Frege proof. We now prove the hardness of approximating the minimum step-length of Frege proofs. We write $\mathcal{F} \mathrel{\vdash_{\overline{k}}} \varphi$ to denote the condition that φ has an $\mathcal{F}$-proof of $\leq k$ steps.

Hardness Theorem 18

(a) *If $P \neq NP$, then there is no polynomial time algorithm which can approximate Minimum Step-Length Frege Proof to within a constant factor.*

(b) *If $NP \not\subseteq QP$, then there is no polynomial time algorithm which can approximate Minimum Step-Length Frege Proof to within a factor of $2^{\log^{(1-\varepsilon)} n}$ for any ε.*

This theorem holds whether Frege proofs are taken to be tree-like or dag-like.

To prove Theorem 18, fix a Frege proof system $\mathcal{F}$ with language L. For proving upper bounds on the step-length of $\mathcal{F}$-proofs we need the following lemma:

Lemma 19

(1) $\tau_i^{\ell,L}$ *has an* $\mathcal{F}$*-proof of step-length* $O(\ell)$.

(2) *Let* $\varphi(x_1,\ldots,x_k)$ *be the* L*-translation of an* $\{\wedge,\vee\}$*-formula. Suppose* $\langle v_j \rangle$ *is a satisfying assignment and that* $I = \{i : v_i = \top\}$. *Then, for every set of formulas* $\psi_1,\psi_2,\ldots,\psi_k$, *there is a tree-like* $\mathcal{F}$*-proof of*

$$\left(\bigwedge_{i\in I}\psi_i\right) \to \varphi(\psi_1,\ldots,\psi_k)$$

of step-length $O(k^2+n) = O(n^2)$.

Proof. (1) Since $\mathcal{F}$ is a schematic system there is a constant upper bound on the number of steps necessary to prove $(\bot \vee \varphi)^L$ from φ^L. Therefore a $\mathcal{F}$-proof of $\tau_i^{\ell,L}$ with $O(\ell)$ steps can be given by proving successively $\varphi_i^{k,L}$ for $k = 0,1,\ldots,\ell$. (2) is proved similarly to Lemma 11. $\square$

To finish the proof of Theorem 18 suffices to prove:

Theorem 20 *There is an A-reduction from the Monotone Minimum Satisfying Assignment problem to the Minimum Step-Length Frege Proof problem for* $\mathcal{F}$. *This reduction works both for tree-like* $\mathcal{F}$*-proofs and dag-like* $\mathcal{F}$*-proofs.*

Proof. Suppose we have formula $\varphi(x_1,\ldots,x_k)$ of size $n = |\varphi|$ over the basis $\{\vee,\wedge\}$, which is an instance of Monotone Minimum Satisfying Assignment. Take m sufficiently large compared to n, say $m = n^3$, and consider the tautology

$$\psi \;=\; \varphi^L(\tau_1^m,\tau_2^m,\ldots,\tau_k^m).$$

(a) Suppose that the optimal solution to the Monotone Minimum Satisfying Assignment problem for φ has $\rho(\varphi) \le p_1$. We have $|\varphi^L| = O(|\varphi|)$, so by Lemma 19 ψ has a tree-like $\mathcal{F}$-proof of length $O(p_1\cdot m+n^2) = O(p_1\cdot m)$.

(b) Suppose that $\rho(\varphi) \ge p_2$. Let P be some $\mathcal{F}$-proof of ψ. Let I be the index set of all i such that $\tau_i^{\ell,L}$ is active in P for all $0 \le \ell \le m$. For $j \notin I$, choose j_r so that $\tau_j^{j_r,L}$ is not active in P. By Lemma 10 we have that, after replacing all $\tau_j^{j_r,L}$ with $\bot$ for all $j \notin I$, the formula ψ remains a tautology. Hence the formula φ is satisfied by the truth assignment corresponding to the characteristic function of I, hence $|I| \ge p_2$. The total number of active formulas $\tau_i^{\ell,L}$, $i \in I$, is greater than or equal to $p_2 \cdot m$; therefore, by Theorem 8, $|P| = \Omega(p_2\cdot m)$.

(a) and (b) establish that $\varphi \mapsto \psi$ is an A-reduction. This proves Theorems 20 and 18. $\square$

One can show that this reduction can be applied to monotone circuits as well:

Theorem 21 *There is an A-reduction from the Monotone Minimum Circuit Satisfying Assignment problem to the Minimum Step-Length Frege Proof problem for $\mathcal{F}$. This is true for both the tree-like and dag-like versions of $\mathcal{F}$.*

Proof. The proof of this theorem is similar to that of Theorem 13. In fact, the identical reduction works and the same $\mathcal{F}$-proofs establish the upper and lower bounds; however, the actual bounds change since we are measuring proof length in terms of steps instead of symbols. For the lower bound, it is easy to see that Lemma 10 implies that any proof of ψ^L must have step-length $\Theta(\rho \cdot m)$. For the upper bound, it is straightforward to verify that the (tree-like) proof of ψ^L derived from the proof of ψ constructed in the proof of Lemma 13 has length $O(\rho \cdot m + n^2)$. □

Remark. All of our hardness results for approximating step-length and symbol-length of Frege proofs also apply to extended Frege systems. To see this, it suffices to note that all the upper and lower bounds on the length of Frege proofs which were obtained in the proofs of Theorems 12, 13, 16 and 20 also apply to extended Frege proofs. Of course it is obvious that the upper bounds apply since every Frege proof is an extended Frege proof. The lower bounds also apply, since Theorems 7 and 8 are also true for extended Frege systems ([10]).

4 The Hardness of Resolution

Resolution is a well-known proof system and its extensions are widely used as a foundation for many theorem proving systems. Thus it is of particular interest that it is difficult to find approximately shortest resolution refutations.

We shall be concerned exclusively with the propositional resolution systems. Recall that a *literal* is either a variable p or the negation of a variable $\overline{p}$. A *clause* is a finite set of literals and is interpreted as the disjunction of its members. A *set of clauses* Γ is interpreted as the conjunction of its member clauses; thus a set of clauses can be identified with a formula in conjunctive normal form.

The resolution rule allows an inference of the form

$$\frac{C \cup \{p\} \qquad D \cup \{\overline{p}\}}{C \cup D}$$

It is well-known that resolution is sound and complete as a refutation system; namely, a set Γ is unsatisfiable if and only if the empty clause can be derived using only resolution inferences from the clauses in Γ.

A *Horn* clause is a clause which contains at most one positive literal.

A resolution refutation consists of a sequence of clauses, ending with the empty clause. We can measure the length or size of a refutation in terms of either its step-length or its symbol-length. The *step-length* of the refutation is just equal to the number of clauses in the refutation. The *symbol-length* is defined to equal the sum of the cardinalities of the clauses appearing in the refutation. (Unlike the situation for Frege proofs where proof length is traditionally defined in terms of symbol-length, there seems to be no fixed convention on how measure the length of resolution refutations: thus, we shall always explicitly include one of the modifiers 'step-' or 'symbol-'.)

A refutation can be either tree-like or dag-like; unless it is explicitly stated otherwise, refutations are considered to be dag-like. We shall obtain the best possible results in that our upper bounds on length will apply to tree-like refutations and our lower bounds on length will apply to dag-like refutations. In the introduction, we introduced the Minimum Length Resolution Refutation problem: henceforth we'll be more precise and talk about the Minimum Step-Length Resolution Refutation and the the Minimum Symbol-Length Resolution Refutation problems.

Hardness Theorem 22

(a) *If $P \neq NP$, then there is no polynomial time algorithm which can approximate Minimum Step-Length (Symbol-Length) Resolution Refutation to within a constant factor.*

(b) *If $NP \not\subseteq QP$, then there is no polynomial time algorithm which can approximate Minimum Step-Length (Symbol-Length) Resolution Refutation to within a factor of $2^{\log^{(1-\varepsilon)} n}$ for any ε.*

These hardness results apply to both dag-like and tree-like resolution. In addition, the hardness results apply to Horn resolution, where all the input clauses are Horn clauses.

To prove the Hardness Theorem, we shall reduce the Monotone Minimum Circuit Satisfying Assignment problem to Minimum Length Resolution Refutation problems. Let φ be an instance of Monotone Minimum Circuit Satisfying Assignment; we define a set of clauses Γ_φ which will be an A-reduction to the Minimum Length Resolution Refutation problems.

Enumerate the subcircuits of φ as $\varphi_1, \ldots, \varphi_\ell$, where the input variables are first in the enumeration and where each φ_i is listed only after all of its own subcircuits are enumerated, and thus φ_ℓ is φ. Obviously the number ℓ of subcircuits is less than the number of symbols n in φ. We introduce new propositional variables $y_1, \ldots, y_\ell$, and define the set Γ_φ to contain the following clauses:

a. The clause $\{\overline{y_\ell}\}$ is in Γ_φ.

b. For each $i \leq \ell$, if φ_i is $(\varphi_j \wedge \varphi_k)$, then the clause $\{\overline{y_j}, \overline{y_k}, y_i\}$ is in Γ_φ.

c. For each $i \leq \ell$, if φ_i is $(\varphi_j \vee \varphi_k)$, then the clauses $\{\overline{y_j}, y_i\}$ and $\{\overline{y_k}, y_i\}$ are in Γ_φ.

The above clauses describe the evaluation of φ; however, note that they say nothing about the truth of the input variables $x_1, \ldots, x_p$ of φ. For each variable x_i of φ, we introduce new variables $x_{i,j}$ for $j = 1, 2, \ldots, m$, and further include in Γ_φ the following clauses:

d. For each $i \leq p$, the clauses $\{x_{i,1}\}$ and $\{\overline{x_{i,m}}, y_i\}$ and

$$\{\overline{x_{i,j}}, x_{i,j+1}\} \quad \text{for } j = 1, \ldots, m-1$$

are included in Γ_φ. These clauses are said to be *associated with* x_i.

That completes the definition of Γ_φ. Informally, Γ_φ asserts that there exists a truth-evaluation to all subcircuits of ϕ such that: the truth evaluation given to the output circuit is 0, the truth evaluation given to all input variables is 1, and the truth evaluation is consistent with all intermediate gate values. Clearly, this is an unsatisfiable formula since ϕ is monotone.

Lemma 23 *Let φ be an instance of Monotone Minimum Circuit Satisfying Assignment and let ρ equal the cardinality of the minimum satisfying assignment for φ.*

(1) *Γ_φ has a dag-like refutation with symbol-length (and hence step-length) equal to $O(\rho m + n)$.*

(2) *Γ_φ has a tree-like refutation of symbol-length $O(\rho m + n^2)$ and step-length $O(\rho m + n)$.*

Proof. (a) Let $I \subseteq \{x_1, \ldots, x_p\}$ specify a satisfying assignment for φ of cardinality ρ. The dag-like proof proceeds by first deriving the clause $\{y_i\}$ for each $x_i \in I$. Each $\{y_i\}$ is derived in m steps using the clauses associated with x_i; this part of the refutation takes ρm steps. The refutation then derives clauses $\{y_i\}$ starting with smaller values of i and ending with $\{y_\ell\}$ (e.g., the subcircuits of φ are processed in a bottom-up order). This takes $O(n)$ steps. One further resolution with the input clause $\{\overline{y_\ell}\}$ completes the refutation. Each clause in the refutation contains a constant number of (in fact, at most three) literals. Hence the symbol-length of the refutation is also $O(\rho m + n)$.

(b) The above proof is clearly not tree-like. To form a tree-like refutation, we use a top-down procedure to generate the refutation. The first phase of the refutation starts with the clause $\{\overline{y_\ell}\}$ and derives successively clauses of the form $\{\overline{y_{k_1}}, \overline{y_{k_2}}, \ldots, \overline{y_{k_r}}\}$ with $k_1 > k_2 > \cdots > k_r$. Such a clause is resolved with one of the (at most two) clauses that contain y_{k_1} positively. This continues until we have a clause which contains only literals $\overline{y_i}$ corresponding to input x_i of φ. It is possible to do this so that the remaining clause is just $\{\overline{y_i} : x_i \in I\}$. For the second phase of the refutation, derive the clauses $\{y_i\}$, for $x_i \in I$, with ρm steps, and for the third phases, use ρ resolutions to derive the empty clause.

There are obviously $O(n)$ steps in the first and third phases of the derivation, so the whole refutation has $O(\rho m + n)$ steps. Furthermore, each clause in the first and third phase contains at most n literals. The second phase contains ρm clauses each with a constant number of literals. Therefore, the symbol-length of the tree-like refutation is $O(\rho m + n^2)$. □

Lemma 24 *Let φ and ρ be as above. Then any resolution refutation (dag-like or tree-like) must have step-length, and hence symbol-length, of at least ρm.*

Proof. Let R be a resolution refutation. An input variable x_i is defined to be *R-analyzed* if every one of the $(m+1)$-clauses associated with x_i is used in the refutation R. Obviously it will suffice to prove that at least ρ input variables are R-analyzed. In fact, if I is defined to equal the set of R-analyzed variables, then I implies a satisfying assignment for φ.

This last fact is almost immediate. To prove it formally, we define a truth assignment τ as follows: (1) τ assigns truth values to variables y_i according to the value I assigns to φ_i (2) τ assigns *True* to $x_{i,k}$ iff each clause $\{x_{i,1}\}$ and $\{\overline{x_{i,j}}, x_{i,j+1}\}$ for $1 \leq j < k$ is used in R. If I doesn't satisfy φ, then τ would satisfy all the clauses used in the refutation R, which is impossible. Therefore, I is a satisfying assignment for φ. □

As an immediate corollary to these two lemmas we get:

Theorem 25 *There is an A-reduction from the Monotone Minimum Circuit Satisfying Assignment problem to the Minimum Length Resolution Refutation problems. This reduction works for both tree-like and dag-like refutations and for both step-length and symbol-length. Furthermore, the reduction produces only sets of Horn clauses.*

This immediately implies Theorem 22.

5 The propositional sequent calculus

This section covers the hardness results for the propositional sequent calculus: somewhat surprisingly, the hardness results apply equally to the sequent calculus with cuts and to the cut-free sequent calculus. We presume the reader is familiar with the sequent calculus: any of the usual variants of the sequent calculus may be used with the proviso that initial sequents are of the form $A \to A$ where A may be *any* formula (not necessarily atomic).

The propositional sequent calculus (with cuts allowed) and Frege systems are very close in strength and are known to p-simulate each other (actually they simulate each other to within a factor of $O(\log n)$, see [7]).

Theorem 26 *There is an A-reduction from the Monotone Minimum Circuit Satisfying Assignment problem to the Minimum Length Propositional Sequent Calculus Proof problem.*

As usual, this theorem holds for proof length measured in terms of either number of symbols or number of steps. In addition, it holds for the tree-like and the dag-like versions of the sequent calculus. Since the proof of Theorem 26 is quite similar to the proofs of section 3, we shall omit it.

As an immediate corollary, the propositional sequent calculus enjoys the same hardness results as we have obtained for the other proof systems.

We now turn to the cut-free propositional sequent calculus. Our main theorem implies that all of our hardness results apply also to this proof system:

Theorem 27 *There is an A-reduction from the Monotone Minimum Circuit Satisfying Assignment problem to the Minimum Length Cut-Free Propositional Sequent Calculus Proof problem.*

For the proof of this theorem, we will presume that the sequent calculus includes the connectives $\land$, $\lor$ and $\to$ (although the results hold even if the only connectives are $\land$ and $\lor$). We use $\Rightarrow$ for the sequent connective (which should not be confused with the implication sign $\to$!) We remind the reader of the Kreisel-Takeuti trick of replacing cuts with $\to$:left inferences:

$$\frac{\Gamma \Rightarrow \Delta, A \quad A, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta} \qquad \Rightarrow \qquad \frac{\Gamma \Rightarrow \Delta, A \quad A, \Gamma \Rightarrow \Delta}{A \to A, \Gamma \Rightarrow \Delta}$$

Proof. We will first handle the reduction of the Monotone Minimum Satisfying Assignment problem to the Minimum Length Cut-Free Sequent Calculus Problem, and then discuss how to modify it to get the desired reduction to the Monotone Minimum Circuit Satisfying Assignment problem.

Let $\varphi(x_1, \ldots, x_k)$ be an $\{\land, \lor\}$ formula. Define the formulas τ_i^m as in section 3.2. Let ψ be the formula

$$\bigwedge_{i=1}^{k} (\tau_i^m \to \tau_i^m) \;\to\; \varphi(\tau_1^m, \ldots, \tau_k^m).$$

We shall prove that, on one hand, $\Rightarrow \psi$ has a cut-free tree-like proof which has step-length $O(\rho \cdot m + n^2)$ and symbol-length $O(\rho \cdot m^2 + m \cdot n^3)$; and, on the other hand, that any dag-like proof of $\Rightarrow \psi$ requires $\Omega(\rho \cdot m)$ steps and $\Omega(\rho \cdot m^2)$ symbols. Taking $m = n^4$ then proves that the mapping $\varphi \mapsto \psi$ is an A-reduction of the Monotone Minimum Satisfying Assignment problem to the Minimum Length Cut-Free Sequent Calculus Problem.

The lower bounds are almost immediate, since the notion of "active formula" makes sense for the sequent calculus and thus Theorems 7 and 8 apply to the sequent calculus. (Side remark: this even gives lower bounds on the length of proofs of ψ which use cuts!)

We now argue the upper bound: let Γ^m denote the cedent

$$\tau_1^m \to \tau_1^m, \tau_2^m \to \tau_2^m, \ldots, \tau_k^m \to \tau_k^m.$$

To prove the upper bound, it will suffice to describe a proof of the sequent $\Gamma^m \Rightarrow \varphi(\tau_1^m, \ldots, \tau_k^m)$, since the formula ψ is easily proven from this sequent with only $O(n^2)$ additional steps and $O(m \cdot n^3)$ additional symbols. Let v be a minimum satisfying assignment for φ and $I = \{i : v(x_i) = \top\}$. So $|I| = \rho$. We let T_I^m be the cedent

$$\tau_{i_1}^m, \tau_{i_2}^m, \ldots, \tau_{i_\rho}^m,$$

where $I = \{i_1, \ldots, i_\rho\}$. Proceeding by induction on the complexity of φ and using the fact that I corresponds to a satisfying assignment for φ, it is easy to check that the sequent $T_I^m \Rightarrow \varphi(\tau_1^m, \ldots, \tau_k^m)$ has a cut-free proof of step-length $O(n^2)$ and symbol-length $O(m \cdot n^3)$. Now there are also tree-like, cut-free sequent calculus proofs of the sequents $\Rightarrow \tau_{i_r}^m$ which have symbol-length $O(m^2)$ and step-length $O(m)$. Using ρ of these derivations, followed by ρ uses of the $\to$:left rule, we derive the following sequent (this is based on the trick of Kreisel-Takeuti):

$$\tau_{i_1}^m \to \tau_{i_1}^m, \ldots, \tau_{i_\rho}^m \to \tau_{i_\rho}^m \Rightarrow \varphi(\tau_1^m, \ldots, \tau_k^m).$$

Using $k - \rho$ weakening inferences, we derive the desired sequent $\Gamma^m \Rightarrow \varphi(\tau_1^m, \ldots, \tau_k^m)$. The proof overall has $O(\rho \cdot m + n^2)$ steps and $O(\rho \cdot m^2 + m \cdot n^3)$ symbols.

We next prove there is a reduction of the Monotone Minimum Circuit Satisfying Assignment problem to the Minimum Length Cut-Free Sequent Calculus Problem. The proof method is a blend of the above reduction and the proof of Theorem 13. As before, assume we have a circuit C with inputs $x_1, \ldots, x_k$ and with internal gates $x_{k+1}, \ldots, x_n$ which is specified as a set of instructions $x_i := x_{j_{i,1}} OP_i x_{j_{i,2}}$ where each OP_i is $\land$ or $\lor$ and where $j_{i,1}, j_{i,2} < i$. The gate x_n is the output of C. Let Δ be the cedent containing the following formulas:

a. If OP_i is $\land$, then Δ contains the formula $x_{j_{i,1}} \land x_{j_{i,2}} \to x_i$.

b. If OP_i is $\lor$, then Δ contains the two formulas $x_{j_{i,1}} \to x_i$ and $x_{j_{i,2}} \to x_i$.

Let the formula $\chi(x_1, \ldots, x_n)$ be

$$\left(\bigwedge_{i=1}^{n} (x_i \to x_i) \land \bigwedge \Delta \right) \to x_n.$$

Let $\psi = \psi(x_{k+1}, \ldots, x_n)$ be the formula obtained from χ by replacing each variable x_i with $i \leq k$ with τ_i^m.

Using the 'active formulas' theorems (Theorems 7 and 8), any proof of ψ must have step-length at least $\rho \cdot m$ and must have symbol-length at least $\rho \cdot m^2$. As usual, this upper bound applies to either dag-like or tree-like proofs.

Now we prove there is a tree-like cut-free proof of ψ which has step-length $O(\rho \cdot m + n^2)$ and symbol-length $O(\rho \cdot m^2 + m \cdot n^3)$. Taking $m = n^4$, this will complete the proof of Theorem 27. As before, let v be a minimum satisfying assignment and let I and T_I^m be as above. Let $x_{r_1}, \ldots, x_{r_s}$ be the gates of C made true by the assignment v, with $\{r_i\}_i$ an increasing sequence, so $r_s = n$. Let t be such that $r_t \le k < r_{t+1}$. Let Λ be the cedent containing the formulas $x_{r_\ell} \to x_{r_\ell}$ for $\ell > t$. We claim there is a cut-free tree-like proof P_1 of the sequent

$$\Lambda, \Delta, x_{r_1}, \ldots, x_{r_t} \Rightarrow x_{r_s}$$

which has step-length $O(n^2)$ and symbol-length $O(n^3)$. This proof is obtained as follows: First prove

$$\Lambda, \Delta, x_{r_1}, \ldots, x_{r_s} \Rightarrow x_{r_s}$$

from the initial sequent $x_{r_s} \Rightarrow x_{r_s}$ using weakenings. Then derive successively the sequents S_ℓ:

$$\Lambda, \Delta, x_{r_1}, \ldots, x_{r_\ell} \Rightarrow x_{r_s}$$

for $\ell = s-1, \ldots, t$. The derivation of $S_{\ell-1}$ from S_ℓ proceeds as follows: (a) if OP_i is $\vee$, use an $\to$:left inference with one of the sequents

$$x_{j_{r_\ell,u}} \to x_{r_\ell}, x_{j_{r_\ell,u}} \Rightarrow x_{r_\ell}$$

(with $u \in \{1,2\}$) and then contract the new duplicate formulas in the antecedent; and (b) if OP_i is $\wedge$, do the same thing but with the sequent

$$(x_{j_{r_\ell,1}} \wedge x_{j_{r_\ell,2}}) \to x_{r_\ell}, x_{j_{r_\ell,1}}, x_{j_{r_\ell,2}} \Rightarrow x_{r_\ell}.$$

Examination of the proof P_1 shows that it has step-length $O(n^2)$ and symbol-length $O(n^3)$, where the extra factor of n allows for lots of exchanges at each stage of the proof.

Now replace every occurrence of variables x_i with $i \le t$ in P_1 with the formulas τ_i^m. This gives a proof P_2 of the sequent

$$\Lambda, \Delta, T_I^m \Rightarrow x_n.$$

The step-length of P_2 is still $O(n^2)$ and its symbol-length is now $O(m \cdot n^3)$. Now derive the ρ sequents $\Rightarrow \tau_i^m$ for $i \in I$. Combine these with P_2 using $\to$:left inferences and weakenings to get a proof P_3 of the sequent

$$\Gamma^m, \Lambda, \Delta \Rightarrow x_n.$$

The proof P_3 has step-length $O(\rho \cdot m + n^2)$ and symbol-length $O(\rho \cdot m^2 + m \cdot n^3)$. Finally, the sequent $\Rightarrow \psi$ is easily proved by adding only a small number of inferences to the end of P_3: this increases the lengths of P_3 by at most a constant factor. $\square$

The cut-free proofs constructed in the proof of Theorem 27 included initial sequents of the forms $\tau_i^m \Rightarrow \tau_i^m$, so therefore our arguments only work for variations of the sequent calculus which allow arbitrary formulas A in initial sequents $A \Rightarrow A$. If we worked with dag-like sequent calculus proofs, we could get by with proving initial sequents $\tau_i^m \Rightarrow \tau_i^m$ only once for each $i \in I$ and our upper bounds would still hold. However, our proof methods do not work for tree-like, cut-free sequent calculi which allow only atomic formulas in initial sequents.

6 Hardness results for long proofs

In the previous sections we proved that it is NP-hard to approximate the minimal propositional proof length by any constant factor, and that if NP is not in QP then minimum proof-length cannot be approximated (in polynomial time) within a $2^{\log^{(1-\epsilon)} n}$ factor. The formulas used in obtaining these results had "short" proofs (or refutations); that is, proofs whose length is at most linear in the size of the formula. However, if $NP \neq coNP$, then for any proof system $\mathcal{S}$, there are tautologies whose shortest $\mathcal{S}$-proof is of super-polynomial length. It is therefore interesting to ask whether better non approximability results can be achieved when the proof lengths are not bounded. We provide below a simple argument that implies that, assuming $NP \not\subseteq P/poly$, no polynomial time circuit can approximate the minimum refutation of a given unsatisfiable CNF formula,[2] for every proof system where there are unsatisfiable formulas requiring super-polynomial size refutations, and which satisfy certain other natural properties.

Definition For a proof system $\mathcal{S}$ and an unsatisfiable formula φ, $\mu_{\mathcal{S}}(\varphi)$ is the minimum length of a refutation of φ in $\mathcal{S}$. For an integer n, $MAX_{\mathcal{S}}(n) = \max\{\mu_{\mathcal{S}}(\varphi)\}$, where φ ranges over all unsatisfiable formulas of length $\leq n$.

We say that a non-decreasing function f has *super-polynomial growth* if for every polynomial r, $f(n) > r(n)$ for almost all positive integers n. f has a *smooth* super-polynomial growth if in addition there is a constant D such that for each n there is $1 < d < D$ such that $f(n^d) > f^d(n)$. [If we write $f(n) = n^{e(n)}$, then the first condition states that $e(n)$ is not bounded

[2]we present the results in terms of finding short refutations of unsatisfiable formulas, but equivalent definitions and results are easily obtained for finding short proofs of tautologies.

from above, and the second condition states that for each n there is m, $n < m < n^D$, such that $e(m) > e(n)$.]

For simplicity we assume that $\wedge$ is a connective in $\mathcal{S}$. Formulas ψ and η are said to be *disjoint* if their underlying sets of variables are disjoint.

Theorem 28 *Assume that $NP \not\subseteq P/poly$, and let $\mathcal{S}$ be a proof system which satisfies:*

1. *For every pair of disjoint formulas ψ and η, where η is unsatisfiable, the following holds:*
   (a) *If ψ is unsatisfiable, then $\mu_{\mathcal{S}}(\psi \wedge \eta) < \mu_{\mathcal{S}}(\psi) + r(|\psi| + |\eta|)$ for some (fixed) polynomial r.*
   (b) *If ψ is satisfiable, than $\mu_{\mathcal{S}}(\psi \wedge \eta) \geq \mu_{\mathcal{S}}(\eta)$;*
2. *$MAX_{\mathcal{S}}(n)$ has a smooth super-polynomial growth.*

Then for any polynomial q, there is no polynomial time q-approximation algorithm for the minimum length proof in $\mathcal{S}$.

Observe that property 1 above holds trivially for all proof systems mentioned in this paper. Property 2 is known to hold for resolution,, since in this case $MAX_{\mathcal{S}}(n) < 3^n$ for all n, and by [5], for each n there is an e, $1 < e < 3$ s.t. $MAX_{\mathcal{S}}(n^e) > 2^{\frac{n^e}{40}}$, thus property 2 holds for $D = 3$ for all large enough n's. We conjecture it to be valid for any known proof system in which the proof lengths are not polynomially bounded.

Proof. We show that the existence of a polynomial time q-approximation algorithm, AL, for $\mathcal{S}$, implies polynomial time circuits for solving SAT.

Let j be such that $q(n) < n^j$ for almost all n, and let D be the constant guaranteed by the smooth super-polynomial growth of $MAX_{\mathcal{S}}$. Since $MAX_{\mathcal{S}}$ has super-polynomial growth, for all large enough n it holds that $r(n + n^{2jD}) < MAX_{\mathcal{S}}(n)$. Fix an integer n_0 for which this inequality holds. Since the super-polynomial growth of $MAX_{\mathcal{S}}$ is smooth, there is a number d, $2j \leq d \leq 2jD$, such that $[MAX_{\mathcal{S}}(n_0)]^d < MAX_{\mathcal{S}}(m)$, where $m = {n_0}^d$. Let η_m be a formula of size $\leq m$ such that $\mu_{\mathcal{S}}(\eta_m) = MAX_{\mathcal{S}}(m)$. An input formula ψ of size n_0 is reduced to $\varphi = \psi \wedge \eta_m$, where the variables of η_m are disjoint from these of ψ (note that φ is unsatisfiable and its size is polynomial in that of ψ). We claim that ψ is unsatisfiable if and only if AL on input φ will output a number $k < MAX_{\mathcal{S}}(m)$. To see this, observe that if ψ is unsatisfiable, then by property (1a) above, $\mu_{\mathcal{S}}(\varphi) \le \mu_{\mathcal{S}}(\psi) + r(|\psi| + |\eta_m|) < 2MAX_{\mathcal{S}}(n_0)$. Hence, by the assumption on AL, AL must produce an output $k < (2MAX_{\mathcal{S}}(n_0))^j < MAX_{\mathcal{S}}(m) = \mu_{\mathcal{S}}(\eta_m)$. On the other hand, if ψ is satisfiable, then, by property (1b), $\mu_{\mathcal{S}}(\varphi) \ge \mu_{\mathcal{S}}(\eta_m) = MAX_{\mathcal{S}}(m)$. □

7 Acknowledgments

We are grateful to A.A. Razborov for extremely helpful discussions. We also would like to thank S. Arora for pointing out that Minimum Label Cover can be reduced to Monotone Minimum Satisfying Assignment.

References

[1] S. Arora, L. Babai, J. Stern, and Z. Sweedyk, *The hardness of approximate optima in lattices, codes, and systems of linear equations*, Journal of Computer and System Sciences, 54 (1997), pp. 317–331. Earlier version in *Proc. 34th Symp. Found. of Comp. Sci.*, 1993, pp.724-733.

[2] S. Arora and C. Lund, *Hardness of approximations*, in Approximation Algorithms for NP-hard Problems, D. S. Hochbaum, ed., PWS Publishing Co., Boston, 1996, pp. 399-446.

[3] M. Bellare, *Proof checking and approximation: Towards tight results*, SIGACT News, 27 (1996), pp. 2–13. Revised version at `http://www-cse.ucsd.edu/users/mihir`.

[4] M. Bellare, S. Goldwasser, C. Lund, and A. Russell, *Efficient probabalistically checkable proofs and applications to approximation*, in Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing, Association for Computing Machinery, 1993, pp. 294–304.

[5] P. Beame and T. Pitassi, *Simplified and Improved Resolution Lower Bound*, in Proceedings of the 37th Annual symposium on Foundations of Computer Science, Burlington, Vermont, 1996, IEEE Computer Society, Society, pp. 274–282.

[6] M. L. Bonet, T. Pitassi, and R. Raz, *No feasible interpolation for TC^0-Frege proofs*, in Proceedings of the 38th Annual Symposium on

Foundations of Computer Science, Piscataway, New Jersey, 1997, IEEE Computer Society, pp. 264–263.

[7] M. L. Bonet, *The Lengths of Propositional Proofs and the Deduction Rule*, PhD thesis, U.C. Berkeley, 1991.

[8] M. L. Bonet, T. Pitassi, and R. Raz, *No feasible interpolation for TC^0-Frege proofs*, in Proceedings of the 38th Annual Symposium on Foundations of Computer Science, Piscataway, New Jersey, 1997, IEEE Computer Society, pp. 264–263.

[9] S. R. Buss, *On Gödel's theorems on lengths of proofs II: Lower bounds for recognizing k symbol provability*, in Feasible Mathematics II, P. Clote and J. Remmel, eds., Birkhäauser-Boston, 1995, pp. 57–90.

[10] ——, *Some remarks on lengths of propositional proofs*, Archive for Mathematical Logic, 34 (1995), pp. 377–394.

[11] M. Clegg, J. Edmonds, and R. Impagliazzo, *Using the Groebner basis algorithm to find proofs of unsatisfiability*, in Proceedings of the Twenty-eighth Annual ACM Symposium on the Theory of Computing, Association for Computing Machinery, 1996, pp. 174–183.

[12] S. A. Cook and R. A. Reckhow, *The relative efficiency of propositional proof systems*, Journal of Symbolic Logic, 44 (1979), pp. 36–50.

[13] U. Feige, *A threshold of* $\ln n$ *for approximating set cover*, in Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, Association for Computing Machinery, 1996, pp. 314–318.

[14] K. Iwama, *Complexity of finding short resolution proofs*, in Mathematical Foundations of Computer Science 1997, I. Prívara and P. Ruzicka, eds., Lecture Notes in Computer Science #1295, Springer-Verlag, 1997, pp. 309–318.

[15] K. Iwama and E. Miyano, *Intractibility of read-once resolution*, in Proceedings of the Tenth Annual Conference on Structure in Complexity Theory, Los Alamitos, California, 1995, IEEE Computer Society, pp. 29–36.

[16] S. Khanna, M. Sudan, and L. Trevisan, *Constraint satisfaction: The approximability of minimization problems*, in Twelfth Annual

Conference on Computational Complexity, IEEE Computer Society, 1997, pp. 282–296.

[17] J. Krajíček and P. Pudlák, *Some consequences of cryptographic conjectures for S_2^1 and EF*, in Logic and Computational Complexity, D. Leivant, ed., Berlin, 1995, Springer-Verlag, pp. 210–220.

[18] C. Lund and M. Yannakakis, *On the hardness of approximating minimization problems*, Journal of the Association for Computing Machinery, 41 (1994), pp. 960–981.

[19] R. A. Reckhow, *On the Lengths of Proofs in the Propositional Calculus*, PhD thesis, Department of Computer Science, University of Toronto, 1976. Technical Report #87.

[20] R. Statman, *Complexity of derivations from quantifier-free Horn formulae, mechanical introduction of explicit definitions, and refinement of completeness theorems*, in Logic Colloquium '76, R. Gandy and M. Hyland, eds., Amsterdam, 1977, North-Holland, pp. 505–517.