PROCEEDINGS OF THE YEREVAN STATE UNIVERSITY

Physical and Mathematical Sciences 2011, № 3, p. 3–8

Mathematics

POLYNOMIAL LENGTH PROOFS FOR SOME CLASS OF TSEITIN FORMULAS

A. G. ABAJYAN*

Chair of Discrete Mathematics and Theoretical Computer Science YSU, Armenia

In this paper the notion of quasi-hard determinative formulas is introduced and the proof complexities of such formulas are investigated. For some class of quasi-hard determinative formulas the same order lower and upper bounds for the length of proofs are obtained in several proof systems, basing on disjunctive normal forms (conjunctive normal forms).

Keywords**:** proof complexity, Split Tree, resolution system, resolution over linear equations, determinative conjunct, quasi-hard determinative formula.

1. Introduction. One of the starting points of propositional proof complexity is the seminal paper of Cook and Reckhow [1], where they formalized propositional proof systems as polynomial-time computable functions, which have as their range the set of all propositional tautologies. In that paper Cook and Reckhow also observed a fundamental connection between lengths of proofs and the separation of complexity classes: they showed that there exists a propositional proof system, which has polynomial-size proofs for all tautologies (a polynomially bounded proof system, which is called super system), iff the class *NP* is closed under complementation. From this observation the so called *Cook-Reckhow programme* was derived, which serves as one of the major motivations for propositional proof complexity: to separate *NP* from co*NP* (and hence *P* from *NP*) it suffices to show super-polynomial lower bounds to the size of proofs in all propositional proof systems.

Although the first super-polynomial lower bound to the lengths of proofs had already been shown by Tseitin in the late 60's for the resolution [2], and, therefore, the resolution system is not a super system, but resolution system and some other weak systems are vital to applications as the design of efficient *SAT* (satisfaction of Boolean functions) solvers, and hence proving lower and upper bounds of proof complexity in such systems is a very important field of logic.

In [3] the notion of hard-determinative formulas was introduced and it was shown that such kind of formula hardness is enough to receive a super-polynomial lower bounds to the lengths of proofs in weak systems. In this paper the notion of quasi-hard determinative formulas is introduced and it is shown that such quasi-

* E-mail: ashotabajian@rambler.ru

hardness can also be useful to obtain "good" bounds. It is shown that some class of Tseitin formulas is quasi-hard determinative, and for this class the same order lower and upper bounds for proof complexities are obtained in several weak systems.

2. Main Notions and Notations. To prove our main results, we recall some notions and notations. We will use the current concept of the unit Boolean cube (E^n), a propositional formula, a tautology, a proof system for Classical Propositional Logic (CPL) and proof complexity.

By $|\varphi|$ we denote the size of formula φ, defined as the number of all variable entries. It is obvious that the full length of a formula, which is understood to be the number of all symbols and the number of all entries of logical signs, are bounded by some linear function in $|\varphi|$. A tautology φ is called minimal, if φ is not an instance of a shorter tautology.

Following the usual terminology we call the variables and negated variables literals. The conjunct K can be simply represented as a set of literals (no conjunct contains a variable and its negation at the same time).

In [3] the following notions were introduced.

We call a replacement rule each of the following trivial identities for a propositional formula φ:

$0 \& \psi = 0,\ \ \psi \& 0 = 0,\ \ 1 \& \psi = \psi,\ \ \psi \& 1 = \psi,\ \ \psi \& \psi = \psi,\ \ \psi \& \overline{\psi} = 0,\ \ \overline{\psi} \& \psi = 0,$
$0 \vee \psi = \psi,\ \ \psi \vee 0 = \psi,\ \ 1 \vee \psi = 1,\ \ \psi \vee 1 = 1,\ \ \psi \vee \psi = \psi,\ \ \psi \vee \overline{\psi} = 1,\ \ \overline{\psi} \vee \psi = 1,$
$0 \supset \psi = 1,\ \ \psi \supset 0 = \psi,\ \ 1 \supset \psi = \psi,\ \ \psi \supset 1 = 1,\ \ \psi \supset \psi = 1,\ \ \psi \supset \overline{\psi} = \overline{\psi},\ \ \overline{\psi} \supset \psi = \psi,$
$\overline{0} = 1,\ \ \overline{1} = 0,\ \ \overline{\overline{\psi}} = \psi$.

Application of a replacement rule to some word consists in replacing some of its subwords, having the form of the left-hand side of one of the above identities by the corresponding right-hand side.

Let φ be a propositional formula, $X = \{x_1, \ldots, x_n\}$ be the set of all variables of φ and $X' = \{x_{i_1}, \ldots, x_{i_m}\}$ $(1 \le m \le n)$ be some subset of X.

Definition 1. Given $\sigma = \{\sigma_1, ..., \sigma_m\} \in E^m$, the conjunct $K^\sigma = \{x_{i_1}^{\sigma_1}, x_{i_2}^{\sigma_2}, ..., x_{i_m}^{\sigma_m}\}$ is called φ*-determinative*, if assigning σ_j $(1 \le j \le m)$ to each x_{i_j} and successively using replacement-rules we obtain the value of φ (0 or 1) independently of the values of the remaining variables.

Definition 2. We call the minimal possible number of variables in a φ-determinative conjunct the *determinative size* of φ and denote it by $d(\varphi)$.

Obviously, $d(\varphi) < |\varphi|$ for every formula φ, and the smaller is the difference between these quantities, the "harder" can be considered the formula under study.

Definition 3. Let φ_n $(n \ge 1)$ be a sequence of minimal tautologies. If for some n_0, $\forall n \ge n_0$, $d(\varphi_n) < d(\varphi_{n+1})$, then the formulas $\varphi_{n_0}, \varphi_{n_0+1,} \varphi_{n_0+2}, ...$ are called *quasi-hard determinative.*

Examples.

1. The determinative conjuncts of the formulas $\varphi_k = x_1 \supset (x_2 \supset (\cdots \supset (x_{k-1} \supset x_k)\cdots))$ $(k \ge 3)$ are in particular $\{x_k\}, \{\overline{x}_1\}, \{\overline{x}_{k-1}\}$, $\{x_{k-1}, x_k\}$, $\{x_1, x_2, \ldots, x_k\}$, therefore, $d(\varphi_k) = 1$ for all $k \ge 3$.

2. For the well-known tautologies $PHP_n = \underset{i=1}{\overset{n+1}{\&}} \underset{j=1}{\overset{n}{\vee}} x_{ij} \supset \underset{1\le i<k\le n+1}{\vee} \; \underset{1\le j\le n}{\vee} (x_{ij} \,\&\, x_{kj})$ $(n \ge 1)$ presenting the Pigeonhole Principle, the determinative conjuncts are in particular $\{x_{11}, x_{21}\}, \{\overline{x}_{11}, \overline{x}_{12}, \dots, \overline{x}_{1n}\}$, therefore, $d(PHP_n) = 2$ for all $n \ge 1$.

3. For example, the following tautologies $u_n = x_n \equiv x_{n-1} \equiv \cdots \equiv x_1 \equiv x_n \equiv$ $\equiv x_{n-1} \equiv \cdots \equiv x_1$ are quasi-hard determinative. Indeed, $d(u_n) = n$ and for $n \ge 1$ $d(u_{n+1}) > d(u_n)$. Other sequences of quasi-hard tautologies can be constructed on the base of graphs.

Proposition. Any disjunctive normal form (DNF) for tautology φ contains at least $2^{d(\varphi)}$ conjuncts.

Proof. Let's assume tautology φ has n distinct variables. Each K_j conjunct from DNF will cover at most $2^{n-d(\varphi)}$ vertices from E^n. In that case we need at least $2^n / 2^{n-d(\varphi)} = 2^{d(\varphi)}$ conjuncts to cover all E^n. □

Let us recall the definition of Tseitin graph formulas [2]. Let G be a connected and finite graph with no loops, and assume distinct literals are attached to its edges.

Definition 4. Graph is called *marked*, if each vertex is marked by 0 or 1 and one assigned literal is chosen for each edge.

Let $x_1, \dots, x_n$ be distinct literals, $\varepsilon \in \{0,1\}$. $[x_1, \dots, x_n]^{\varepsilon}$ denotes the set of disjunctions that consists of literals $x_1, \dots, x_n$ and satisfy the following conditions:

1. For each i $(1 \le i \le n)$ either x_i or $\overline{x}_i$ belongs to the disjunction.

2. If ε is odd, then the number of negated literals is even, and if ε is even, the number of negated literals is odd.

Let G be a marked graph. Let's construct the set of $[x_1, \dots, x_n]^{\varepsilon}$ disjunctions for each vertex, where ε is the value assigned to the given vertex and $x_1, \dots, x_n$ are variables assigned to the incident edges. The set of disjunctions constructed for all vertices of graph G is denoted by $\alpha(G)$ and the sum of values assigned to vertices of the graph by modulo 2 is denoted by $\sigma(G)$. In [2] it is proved that $\alpha(G)$ is unsatisfiable, iff $\sigma(G) = 1$.

It is obvious, that if Tseitin graph formulas are constructed on the base of graphs, minimal degree of which is of the same order as the number of vertices, then such formulas are quasi-hard determinative.

Let us recall the definition of some proof systems of CPL.

3.1 . Split Tree System. Let us give the notion of Split Tree (ST) system following [4]. This proof system is the analogue of the Analytic Tableaux system.

DNF is represented as a set of conjuncts, and conjunct is represented as a set of literals. Two formulas are different, if the corresponding sets are different.

Definition 5. Split results by variable x of formula F are called two formulas $F[x]$ and $F[\overline{x}]$, which are being obtained from F by substituting inverse values of x ($x = 1$ and $x = 0$ accordingly).

In the rest of the paper, we will assume that F consists of n variables, and X is the set of its variables $(\|X\| = n)$, $Y \subseteq X$, $Y \ne \varnothing$.

Definition 6. ST for formula F on set Y is called a marked binary tree T, each node v of which is assigned by a formula F_v, and the following statements are satisfied:

1. Formula F is assigned to the root.

2. Assignments on the leaves contain no variables from the set Y (particularly, 0 or 1 can be assigned).

3. The formulas assigned to the children of node v are obtained from F splitting it by variable y, that is, $F_v[y]$ and $F_v[\overline{y}]$, where $y \in Y$, and formula F_v contains y.

When Y coincides with X, we deal with the ST for formula F. The leaves of such a tree can be only 0 or 1. A ST is called closed, if 1 is the only assignment to the leaves. Formula F is a tautology, iff it possesses an closed ST.

3.2 . Resolution System. Let us describe the resolution refutation system (R) following [5]. A *clause* is a disjunction of literals. A conjunctive normal form (CNF) *formula* is a conjunction of clauses.

Resolution is a complete and sound proof system for unsatisfiable CNF formulas. Let C and D be two clauses containing neither x_i nor $\overline{x}_i$. The *resolution rule* allows one to derive $C \vee D$ from $C \vee x_i$ and $D \vee \overline{x}_i$.

Definition 7 (Resolution). A *resolution proof of the clause D from a CNF formula K* is a sequence of clauses $D_1, D_2, \ldots, D_l$ such that:

1. Each clause D_j is either a clause of K or can be obtained from two previous clauses in the sequence using the resolution rule.

2. The last clause $D_l = D$.

A *resolution refutation* of a CNF formula K is a resolution proof of the empty clause from K (the empty clause stands for $FALSE$, that is, no value satisfies to the empty clause.

3.3 . Resolution Over Linear Equations. Now, let us describe $R(lin)$ system following [5]. $R(lin)$ is an extension of resolution, which operates with disjunction of linear equations with integer coefficients. A disjunction of linear equation is of the following form

$$(a_1^{(1)}x_1 + \ldots + a_n^{(1)}x_n = a_0^{(1)}) \vee \ldots \vee (a_1^{(t)}x_1 + \ldots + a_n^{(t)}x_n = a_0^{(t)}),$$

where $t \geq 0$ and the coefficients $a_i^{(j)}$ are integers (for all $0 \leq i \leq n$, $1 \leq j \leq t$). We discard duplicate linear equations from a disjunction of linear equations. Any CNF formula can be translated to a collection of disjunctions of linear equations directly: every clause $\underset{i \in I}{\vee} x_i \vee \underset{j \in J}{\vee} \neg x_j$ (where I and J are sets of indices of variables) involved in the CNF is translated into the disjunction $\underset{i \in I}{\vee}(x_i = 1) \vee \underset{j \in J}{\vee}(x_j = 0)$. For a clause D we denote by $\tilde{D}$ its translation into a disjunction of linear equations. It is easy to verify that any Boolean assignment to the variables $x_1, \ldots, x_n$ satisfies a clause D, iff it satisfies $\tilde{D}$.

As we wish to deal with Boolean values we augment the system with axioms, called *Boolean axioms*: $(x_i = 0) \vee (x_i = 1)$ for all $i \in [n]$.

Axioms are not fixed: for any formula φ we obtain $\neg\varphi$, then we obtain $R(lin)$ translation of CNF of $\neg\varphi$. We also add Boolean axioms for each variable.

Definition 8 $(R(lin))$. Let $K=\{K_1,...,K_m\}$ be a collection of disjunctions of linear equations. An $R(lin)$*-proof from* K *of a disjunction of linear equations* D is a finite sequence $\pi=(D_1,...,D_l)$ of disjunctions of linear equations such that $D_l=D$ and for every $i\in[l]$, either $D_i=K_j$ for some $j\in[m]$, or D_i is a Boolean axiom $(x_h=0)\vee(x_h=1)$ for some $h\in[n]$, or D_i was deduced by one of the following $R(lin)$-inference rules, using D_j, D_k for some $j,k<i$:

Resolution. Let A, B be two disjunctions of linear equations (possibly the empty disjunctions), and let L_1, L_2 be two linear equations. From $A\vee L_1$ and $B\vee L_2$ it is derived $A\vee B\vee(L_1+L_2)$ or $A\vee B\vee(L_1-L_2)$.

Weakening. From a disjunction of linear equations A one derives $A\vee L$, where L is an arbitrary linear equation over X.

Simplification. From $A\vee(0=k)$ derive A, where A is a disjunction of linear equations and $k\neq 0$.

An $R(lin)$ *refutation* of a collection of disjunctions of linear equations K is a proof of the empty disjunction from K. Raz and Tzameret showed, that $R(lin)$ is a sound and complete Cook-Reckhow refutation system for unsatisfiable CNF formulas (translated into unsatisfiable collection of disjunctions of linear equations).

3.4. Proof Complexity, Polynomial Simulation. In the theory of proof complexity one of the characteristics of the proof is t-complexity, defined as the number of proof steps. Let Φ be a proof system and φ be a tautology. We denote by t_φ^Φ the minimal possible value of t-complexity of all the proofs of a tautology φ in Φ.

Let Φ_1 and Φ_2 be two different proof systems. Following [1], we recall

Definition 9. Φ_2 polynomially simulates Φ_1, if there exists a polynomial $p()$ such that for every formula φ derivable both in Φ_1 and in Φ_2, $t_\varphi^{\Phi_2}\leq p(t_\varphi^{\Phi_1})$.

Definition 10. The systems Φ_1 and Φ_2 are polynomially equivalent, iff Φ_1 polynomially simulates Φ_2 and Φ_2 polynomially simulates Φ_1.

4. Main Results. Let us denote by $Tsgf_n$ $(n\geq 2)$ the Tseitin graph formulas, which are constructed on the base of complete n-vertex graphs, only one of vertices of which is marked with 1.

As pointed above, every formula $Tsgf_n$ is unsatisfiable, therefore, has some refutation in R (hence, in $R(lin)$ also), and $\neg Tsgf_n$ is tautology and can be derived in ST.

Theorem.

1. $t_{Tsgf_n}^R\geq 2^{n-1}$.
2. $t_{Tsgf_n}^R\leq t_{Tsgf_n}^{R(lin)}\leq p(2^n)$ for some polynomial $p()$.

Proof.

1. Taking into consideration that $d(\varphi_{Tsgf_n}(n)) = n-1$ and proposition from the third paragraph, we will get $t_{\varphi}^{R}(n) \geq 2^{n-1}$, since the number of axioms in R is more than 2^{n-1}.

2. Let's consider a proof of $\neg Tsgf_n$ in ST. To pass from a complete graph with n vertices to complete graph with $n-1$ vertices, we take one vertex labeled 0 and do successively splits for all the variables labeled to the edges incident to that vertex. As a result, we will have a binary tree with $n-1$ depth, half of which leaves will be assigned by 1, and another half will belong to the same isomorph class, therefore, they will also be counted once. Denote by $\Psi(i)$ the derivation complexity of the complete graph with i vertices. So, $\Psi(1) = 2$ and $\Psi(n) = \Psi(n-1) + \sum_{i=1}^{n-2} 2^i + 1$. It is not difficult to verify that $\Psi(n) = 2^n - n - 1 \leq 2^n$. Taking into consideration that R system polynomially simulates ST system [6], we obtain that there is polynomial $f()$ such that $t_{Tsgf_n}^{R}(n) \leq f(2^n)$.

Taking into consideration that fact, that $R(lin)$ polynomially simulates R system [5], we obtain that $t_{Tsgf_n}^{R(lin)}(n) \leq p(2^n)$ for some polynomial $p()$. □

Corollary 1. $Tsgf_n$ have polymonially-size R, $R(lin)$ refutation.

Indeed, since $|Tsgf_n| = n2^n$, then the proof follows from the Theorem.

Corollary 2. There are lower and upper bounds of the same order for proof complexities in every system, which is polynomially equivalent to R, $R(lin)$ or ST systems.

Received 21.03.2011

REFERENCES

1. **Cook S.A. and Reckhow A.R.** J. Symbolic Logic, 1979, v. 44, p. 36–50.
2. **Tseitin G.S.** Zap. Nauchn. Semin. LOMI. Leningrad: Nauka, 1968, v. 8, p. 234–259 (in Russian).
3. **Aleksanyan S.R and Chubaryan A.A.** Siberian Mathematical Journal, 2009, v. 50, № 2, p. 243–249.
4. **Dantsin E.Ya.** Zap. Nauchn. Sem. LOMI. Leningrad: Nauka, 1981, v. 105, p. 24–44 (in Russian).
5. **Ran Raz and Iddo Tzameret** Ann. Pure Appl. Logic, 2008, v. 155, № 3, p. 194–224.
6. **Pudak P.** Lengths of Proofs. Handbook of Proof Theory. North-Holland, 1998, p. 547–637.