Graph expansion, Tseitin formulas and resolution proofs for CSP

Dmitry Itsykson[1⋆] and Vsevolod Oparin[2⋆⋆]

[1] Steklov Institute of Mathematics at St.Petersburg
dmitrits@pdmi.ras.ru
[2] St.Petersburg Academic University of Russian Academy of Sciences
oparin.vsevolod@gmail.com

Abstract. We study the resolution complexity of Tseitin formulas over arbitrary rings in terms of combinatorial properties of graphs. We give some evidence that an expansion of a graph is a good characterization of the resolution complexity of Tseitin formulas. We extend the method of Ben-Sasson and Wigderson of proving lower bound for the size of resolution proofs to constraint satisfaction problems under an arbitrary finite alphabet. For Tseitin formulas under the alphabet of cardinality d we provide a lower bound $d^{e(G)-k}$ for the tree-like resolution complexity that is stronger than the one that can be obtained by the Ben-Sasson and Wigderson method. Here k is an upper bound on the degree of the graph and $e(G)$ is the graph expansion that is equal to the minimal cut such that both its parts are no more than twice bigger than each other. We give a formal argument why a large graph expansion is necessary for lower bounds. Let $G = \langle V, E\rangle$ be the dependency graph of the CSP, vertices of G correspond to constraints; two constraints are connected by an edge for every common variable. We prove that the tree-like resolution complexity of the CSP is at most $d^{e(H)\cdot\log_{\frac{3}{2}}|V|}$ for some subgraph H of G.

1 Introduction

Backtracking algorithms are the most popular approach to solving NP-hard problems. The running of backtracking algorithms for SAT on unsatisfiable formulas is closely connected with the tree-like resolution proof system. Lower bounds on the complexity of resolution proofs imply the same lower bounds on the running time of backtracking algorithms. First superpolynomial lower bound for resolutions was proved by Tseitin [Tse68]; Tseitin used formulas that coded the following simple fact: in every graph the number of vertices with odd

⋆ partially supported by the grants MK-4108.2012.1 from the President of RF, by RFBR grant 12-01-31239-mol-a, by the Programme of Fundamental Research of RAS and by The Ministry of education and science of Russian Federation, project 8216.

⋆⋆ partially supported by RFBR grant 12-01-31239-mol-a, by Réseau STIC franco-russe and ANR NAFIT 008-01.

degree is even. First exponential lower bound was proved by Urqhart [Urq87]. The strongest known lower bounds were proved using the methods introduced by Ben-Sasson and Wigderson in [BSW01]. From practical point of view it is more interesting to have lower bound for backtracking algorithms on satisfiable formulas; there are several lower bounds on satisfiable formulas [AHI05], [CEMT09], [Its10], [IS12] under various restrictions on heuristics that choose a variable for splitting and a value that would be investigated at first. However all known lower bounds on satisfiable formulas are proved by reduction to lower bounds on unsatisfiable ones.

Baker [Bak95] introduced very natural extension of resolution proof system for constraint satisfaction problems (CSP) and defined the system NG-RES. Baker studied different backtracking algorithms for CSP; Baker introduced the notion of width of CSP and proved that there exists resolution proofs of size exponential only in the width and polynomial in other parameters. Baker also gave a hard distribution for backtracking algorithms for CSP and proved a super polynomial lower bound for NG-RES. Mitchell [Mit02b] introduced the proof system C-RES that was more powerful than NG-RES and proved an exponential lower bound for random CSP in C-RES. Mitchell [Mit03] proved a superpolynomial separation between C-RES and NG-RES and Hwang [Hwa04] proved an exponential separation.

The paper [IPS97] proves that linear lower bound on the proof degree in Polynomial Calculus implies the exponential lower bound on the proof size in Polynomial Calculus under fields. The paper [BGIP99] presents a linear lower bound on the degree of proofs of Tseitin formulas in Polynomial Calculus under fields and rings. This lower bounds are proved only for alphabets of cardinality p^m for prime p; and this result is not claimed to be optimal.

In this paper we are interested in the precise complexity of backtracking algorithms (or tree-like resolution) on Tseitin formulas under an arbitrary finite alphabet. In the propositional case the strongest lower bound for Tseitin formulas follows from the paper of Ben-Sasson and Wigderson. Namely every tree-like resolution proof of Tseitin formula based on a graph with maximal degree at most k has the size at least $2^{e(G)-k}$, where $e(G)$ is an expansion of the graph that is equal to the size of minimal cut such that such that both its parts are no more than twice bigger than each other. Method of Ben-Sasson and Wigderson consists of the two steps: first stage is an establishing of a relationship between the proof size and the width of the proof; the second step is an establishing of a relationship between the width of the proof and the expansion of the CSP. Mitchell in [Mit02a] generalizes the relationship between size and width of the proof to a nonbinary case. The trivial case of such a generalization to the alphabet of size d implies the lower bound $2^{e_d(G)-k}$ for the tree-like resolution complexity of Tseitin formulas, where e_d is the size of the minimal cut such that both its parts are no more than d times bigger than each other. Generally speaking $e_d(G)$ may be much smaller than $e(G) = e_2(G)$.

For arbitrary CSP ϕ the results of [Mit02a] implies the following generalization of [BSW01]:

1. $S_T(\phi) \geq 2^{e(\phi)-k-1}$,
2. $S(\phi) \geq \exp\left(\Omega\left(\frac{(e(\phi)-k-1)^2}{n}\right)\right)$,

where $S_T(\phi)$ and $S(\phi)$ are tree-like and general resolution complexity of ϕ, $e(\phi)$ is the expansion of CSP ϕ.

The latter implies lower bound $2^{e(G)-k-1}$ on the tree-like resolution complexity of Tseitin formulas. By means of more a specific analysis for Tseitin formulas we improve this lower bound and get $d^{(e(G)-k)}$.

It is well known that lower bound proofs for Tseitin formulas use the graph with a high expansion. We study the question whether a high expansion is indeed necessary for lower bounds or not. We give the answer for arbitrary CSP: let $G = \langle V, E\rangle$ be the dependency graph of CSP; vertices of G correspond to constraints; two constraints are connected by an edge for every common variable. We prove that the tree-like resolution complexity of the CSP is at most $d^{e(H)\cdot\log_{\frac{3}{2}}|V|}$ for some subgraph H of G. Thus for Tseitin formula ϕ based on the graph $G = \langle V, E\rangle$ we have that there is a subgraph H of G such that $d^{e(H)-k} \leq S_T(\phi) \leq d^{e(H)\cdot\log_{\frac{3}{2}}|V|}$.

In Section 2 we give definitions of basic concepts. In Section 3 we provide the relationship between width of the proof and the expansion of the CSP. In Section 4 we prove the stronger lower bound for the tree-like resolution complexity of Tseitin formulas. In Section 5 we prove the upper bound on the tree-like resolution complexity of the CSP in terms of the expansion of the dependency graph.

2 Preliminaries

2.1 Constraint satisfaction problem (CSP)

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a finite set of variables that take values from a finite set D, and S be a set of constraints; every constraint defines a subset of variables X' and a set of possible values that variables of X' can take at the same time. A triplet $\langle X, D, S\rangle$ is called a constraint satisfaction problem (CSP). If every constraint restricts at most k variables than we call such problem k-CSP.

A partial substitution is a mapping $\rho : X \to D \cup \{*\}$, where '*' means an unspecified value; a support of a substitution is the set $\rho^{-1}(D)$. A substitution is complete if its support equals to X.

A partial substitution ρ satisfies a constraint $R \in S$ if after the substitution of values of variables from the support of ρ the constraint R is satisfied independently of values of the other variables. A substitution ρ satisfies CSP $\langle X, D, S\rangle$ if ρ satisfies every constraint $R \in S$. CSP ϕ is satisfiable if there exists at least one substitution that satisfies ϕ.

We call a constraint of the type $\neg(x_1 = a_1 \wedge \cdots \wedge x_k = a_k)$ the *nogood*, where $x_1, \ldots, x_k \in X, a_1, \ldots, a_k \in D$. The notion of a nogood is an extension of the notion of a clause in the propositional case ($D = \{0, 1\}$). For example the nogood $\neg(x_1 = 0 \wedge x_2 = 1)$ is equivalent to the clause $(x_1 \vee \bar{x}_2)$.

In what follows we consider only k-CSP and denote $|D| = d$. Every restriction in k-CSP may be written as a conjunction of at most d^k nogoods.

2.2 Backtracking algorithms

Now we define backtracking algorithms for CSP.

A backtracking algorithm is parametrized by two heuristics B and C and a simplification procedure S. A heuristic B takes CSP ϕ and returns a variable x for splitting. A heuristic C takes a pair (ϕ, x) and returns an order on D (this is an order in which an algorithm substitutes values from D for the variable x).

The simplification procedure $S(\phi, x := a)$ removes from the $\phi[x := a]$ all constraints that have been already satisfied.

A backtracking algorithm $A(\phi)$ is defined as follows

- If ϕ does not contain constraints, return SATISFIABLE.
- If ϕ contains already falsified constraint, return UNSATISFIABLE.
- Pick a variable $x := B(\phi)$. According to the order given by $C(\phi, x)$, for all $a \in D$ make a recursive call $A(S(\phi, x := a))$. If one of recursive call returns SATISFIABLE, immediately return SATISFIABLE, otherwise return UNSATISFIABLE.

The running time of the backtracking algorithm is the size of the recursion tree. We ignore the computational complexity of heuristics B and C.

2.3 Resolution proof system

We consider only unsatisfiable instances of CSP.

We define a resolution proof system that generalizes a well known system in the propositional case. This definition is due to [Bak95].

The resolution proof system is a way to show that a given CSP is unsatisfiable. We assume that all constraints are represented as a set of nogoods.

Let $\{N_a\}_{a \in D}$ be a set of nogoods such that $N_a = \neg(x = a \wedge \alpha_a)$ for every $a \in D$. A nogood $\neg(\bigwedge_{a \in D} \alpha_a)$ is a resolvent of $\{N_a\}_{a \in D}$.

Definition 1. *A sequence of nogoods $\pi = \{N_i\}$ is a resolution proof for CSP ϕ if*

- *every nogood N_i is either a nogood of ϕ or a resolvent of d nogoods that precede N_i: $N_{j_1}, \ldots, N_{j_d}$, where $j_1, \ldots, j_d < i$;*
- *the last nogood in the π is an empty nogood $\neg()$ (i.e. contradiction).*

Every resolution proof may be represented as a directed acyclic graph with nogoods as vertices, there is an arc between N_i and N_j if N_i is in the premise of the resolution rule that produced N_j. The proof is called tree-like if its graph is a tree. A tree-like resolution proof system accepts only tree-like proofs.

Similarly to the propositional case, the running of backtracking algorithms on unsatisfiable CSPs and tree-like resolution proofs are equivalent.

Proposition 1 ([Bak95][Hwa04]). *The size of the smallest tree-like resolution refutation is exactly the same as the size of the minimal recursion tree of the backtracking algorithm.*

Thus upper and lower bounds on the size of tree-like resolution proofs provide the same upper and lower bounds on the running time of backtracking algorithms.

2.4 Tseitin formulas and expansion

The paper [BGIP99] generalizes Tseitin formulas [Tse68] to the nonbinary case. Consider a graph $G = \langle V, E\rangle$ and a function $f : V \to \mathbb{Z}_d$. We associate every edge $e \in E$ with a variable x_e. For every vertex u we have a constraint of type

$$\sum_{(u,v)} \gamma_{(u,v)} \cdot x_{(u,v)\in E} = f(u) \bmod d$$

where $\gamma_{(u,v)} \in \{+1, -1\}$. Every edge (u, v) corresponds to a variable $x_{(u,v)}$ and two values $\gamma_{(u,v)}$ and $\gamma_{(v,u)}$ that satisfy $\gamma_{(u,v)} + \gamma_{(v,u)} = 0$. Note that $x_{(u,v)}$ and $x_{(v,u)}$ denote the same variable.

The following lemma is very similar to the propositional case.

Lemma 1. *Tseitin formula $\phi(G, f)$ based on a connected graph G is satisfiable if and only if $\sum_v f(v) = 0$.*

Definition 2. *The expansion of a graph $G = \langle V, E\rangle$ is $e(G) = \min_{A\subseteq V,\, \frac{1}{3}|V|\le|A|\le\frac{2}{3}|V|} |E(A, \bar{A})|$.*

Further we will see the relationship between the expansion of a graph and the sizes of resolution proofs of Tseitin formulas.

3 Resolution width and expansion

The paper [BSW01] introduced a technique of proving lower bounds in the propositional resolution proof system, that are quite strong. We generalize that result to CSP.

Let us consider a k-CSP $\phi = \langle X, D, S\rangle$ that is represented by a set of nogoods.

A width of a nogood is the number of variables that appear in it. If π is a resolution proof of ϕ, then a width of π is the maximal width of a nogood in π; we denote it by $W(\pi)$. A width of refutation of CSP ϕ is the minimal width of all resolution proofs of ϕ; we denote it by $W(\phi \vdash 0)$.

Theorem 1 ([Mit02a]). *For every k-CSP ϕ the following inequalities are satisfied*

$$S_T(\phi) \ge 2^{W(\phi\vdash 0)-k},$$

$$S(\phi) \ge \exp\left(\Omega\left(\frac{(W(\phi \vdash 0) - k)^2}{n}\right)\right),$$

where $S_T(\phi)$ is the minimal size of a tree-like resolution proof of ϕ and S_ϕ is the minimal size of a resolution proof of ϕ.

Let's consider CSP ϕ; let S be the set of constraints of ϕ (it is not necessary that all constraints are nogoods). Let F be some subset of the set of constraints S; we denote by ∂F the set of variables x such that there is exactly one constraint in F that depends on x. The expansion of ϕ is defined as follows

$$e(\phi) = \min_F |\partial F|,$$

where the minimum is over all $F \subseteq S$ such that $\frac{1}{3}|S| \leq |F| \leq \frac{2}{3}|S|$.

Definition 3. *Let ϕ be an unsatisfiable CSP. We say that ϕ is* minimally unsatisfiable *if ϕ becomes satisfiable after removing any of its constraints.*

Theorem 2. *Let ϕ be a minimally unsatisfiable CSP and S be a constraint set of ϕ. Let ϕ satisfy the following property:*

- *for every constraint $f \in S$ every two substitutions that violate f differ in at least two variables.*

Then $W(\phi \vdash 0) \geq e(\phi) - 1$.

Proof. We say that a nogood N is semantically implied from $F \subseteq S$, if every substitution that satisfies F also satisfies N. We denote this implication by $F \models N$. We define Ben-Sasson-Wigderson measure on the set of all nogoods. For a nogood N we define $\mu(N) = \min\{|F| \mid F \subseteq S, F \models N\}$. The following properties are straightforward:

- $\mu(N) \leq 1$ for every nogood N from ϕ;
- $\mu(\neg()) = |S|$;
- If N is the resolvent of $\{N_a\}_{a \in D}$, then $\mu(N) \leq \sum_{a \in D} \mu(N_a)$.

Lemma 2. *Let F be a minimal set of constraints that semantically implies N. Then the size of N is at least $|\partial F|$.*

Proof. Note that for every constraint $f \in F$ there is the substitution ρ_f that refutes N and f, but ρ_f satisfies every other constraint $g \in F$. Otherwise we may remove such constraint from F and this contradicts to the minimality of F.

For $x \in \partial F$ let $f \in F$ be the constraint depended on x. Then there exists such $a \in D$ that changing a value of variable x in ρ_f to a makes it satisfy f and therefore satisfy N. Thus N depends on x. □

In the propositional case we may finish the proof since the properties of a measure μ imply that every resolution proof contains a nogood N with a measure in $[\frac{1}{3}|S|, \frac{2}{3}|S|]$. Lemma 2 implies that N contains at least $e(\phi)$ variables. However for arbitrary d we can't guarantee that such nogood N exists. We choose another way.

Any resolution proof of the formula ϕ contains the nogood N such that it is the resolvent of nogoods N_a on a variable x, $a \in D$, $\mu(N) > \frac{1}{3}|S|$ and for every premise N_a the inequality $\mu(N_a) \leq \frac{1}{3}|S|$ holds.

Let F_a be the minimal subset of constraints such that $F_a \models N_a$. Since $|F_a| \leq \frac{1}{3} \cdot |S|$, we can choose $D' \subseteq D$ in such a way that for F' defined as $\bigcup_{a \in D'} F_a$ we have $\frac{1}{3} \cdot |S| \leq |F'| \leq \frac{2}{3} \cdot |S|$. Thus $|\partial F'| \geq e(\phi)$, and by Lemma 2 for every variable $y \in \partial F'$ there exists a nogood N_a ($a \in D'$) that depends on y. Therefore $(\partial F' \backslash \{x\}) \subseteq \mathrm{Vars}(N)$, hence $|\mathrm{Vars}(N)| \geq e(A) - 1$, where $\mathrm{Vars}(N)$ is a set of variables from the nogood N. □

Corollary 1. *If a Tseitin formula $\phi(G, f)$ is unsatisfiable, then $W(\phi(G, f) \vdash 0) \geq e(G) - 1$*

Proof. Follows from Theorem 2 and Lemma 1. □

Finally if the degree of all vertices in a graph G is at most k and Tseitin formula $\phi(G, f)$ is unsatisfiable, then Corollary 1 and Theorem 1 imply the following lower bounds:

1. $S_T(\phi) \geq 2^{e(G)-k-1}$,
2. $S(\phi) \geq \exp\left(\Omega\left(\frac{(e(G)-k-1)^2}{n}\right)\right)$,

Note that we have 2 in the base of the exponent in the tree-like case as it was for a binary alphabet. But it is more natural to have number d in the base of the exponent since every node of the tree has d children. In the next section we give more accurate analysis for Tseitin formulas and prove a lower bound $d^{e(G)-k}$ for tree-like resolution.

4 Lower bound for Tseitin formulas

In this section we prove the lower bound for the size of tree-like resolution proofs of Tseitin formulas that is stronger than the lower bound from the previous section. Consider a graph $G = \langle V, E \rangle$ and the unsatisfiable Tseitin formula ϕ based on it. Let the maximal degree of G be at most k. We assume that the domain D equals $\mathbb{Z}_d$. We prove that $S_T(\phi) \geq d^{e(G)-k}$, where S_T is the size of the minimal tree-like resolution proof of ϕ.

4.1 Reduced splitting tree

Let $G = \langle V, E \rangle$ be a connected graph with the maximal degree of vertices at most k. We consider a protocol of a backtracking algorithm and define the notion of the complexity of the graph G. It equals the minimal size of resolution proofs of $\phi(G, f)$. For a connected graph G we define

$$C(G) = \begin{cases} 1, \text{ if } |V| = 1 \\ \min_{e \in E} T(G \setminus e) + 1, \text{ otherwise,} \end{cases}$$

where $T(G)$ is defined for all G with at most two connected components in the following way:

$$T(G) = \begin{cases} d \cdot C(G), \text{ if } G \text{ is connected;} \\ (d-1) \cdot C(G_1) + C(G_2), \text{ otherwise,} \end{cases}$$

where G_1 and G_2 are two connected components of graph G and $C(G_1) \leq C(G_2)$.

Lemma 3. *The minimal running time of a backtracking algorithm on an unsatisfiable Tseitin formula $\phi(G, f)$ based on a connected graph G does not depend on the function f and equals $C(G)$.*

Proof. We prove it by induction on the number of edges. The base of induction is trivial. Consider now an arbitrary graph $G = \langle V, E \rangle$ and a function $f : V \to \mathbb{Z}_d$.

Let optimal backtracking algorithm start with a splitting on a variable x_e. In the first case $G \setminus e$ is connected. Then we have to solve d subproblems of the type $\phi(G \setminus e, f'_a)$, where the function f'_a differs from f at the ends of the edge e. By the induction hypothesis the minimal running time of a backtracking algorithm, on the formula $\phi(G \setminus e, f'_a)$ is equal to $C(G \setminus e)$. Therefore the total number of steps of the optimal backtracking algorithm is $d \cdot C(G \setminus e)$.

In the second case the edge e is the bridge of the graph G. Let G_1 and G_2 be the two connected components of $G \setminus e$. After the substitution $x_e := a$, the formula $\phi(G, f)$ splits into two independent subformulas ϕ_1 and ϕ_2, that correspond to graphs G_1, G_2 and to functions $f_{1,a}$, $f_{2,a}$ respectively.

We show that there is exactly one value of x_e that makes the formula ϕ_i satisfiable for $i = 1, 2$. The inductive hypothesis implies that the minimal complexity of a backtracking algorithm is $(d-1) \cdot C(G_1) + C(G_2) + 1$.

Let an edge e connect vertices u and v and the vertex v belong to G_1. Note that values of functions $f_{1,a}$ and f on the vertices of the graph G_1 can differ only at vertex v. Lemma 1 implies that if we fix $f_{1,a}$-values for all vertices in G_1 except v, then there exists exactly one value of $f_{1,a}(v)$ that makes ϕ_1 satisfiable. □

Lemma 3 allows to present a protocol of a backtracking algorithm in an economical way. We define a rooted tree; nodes of this tree are marked with connected graphs. For the Tseitin formula $\phi(G, f)$ our tree T looks as follows:

- The root of the tree is marked by G.
- Every leaf of the tree is marked by a graph with one vertex.
- Every node of the tree has either one or two children.
- Let node v be marked by a graph G_v. If v has only one child then it is marked by $G_v \setminus e$ for some edge e. If v has two children then each of them is marked by the corresponding connected component of $G_v \setminus e$ for some bridge e in G_v.

We call such a tree a reduced splitting tree.

We define a function f on the nodes of a reduced splitting tree.

$$f(v) = \begin{cases} 1, \text{ if } v \text{ is a leaf;} \\ d \cdot f(u) + 1, \text{ if } u \text{ is a unique child of } v; \\ (d-1) \cdot f(u_1) + f(u_2) + 1, \text{ where } u_1, u_2 \text{ are children of } v \text{ and } f(u_1) \leq f(u_2); \end{cases}$$

For a reduced splitting tree T we define $F(T) = f(r)$, where r is a root of T. It is easy to see that

$$C(G) = \min_T F(T),$$

where the minimum is over all reduced splitting trees for a given graph G.

4.2 Lower bound

We define the notion of the width of the reduced splitting tree.

Let $G = \langle V, E \rangle$ be a connected graph and ϕ be an unsatisfiable CSP based on G. Let T be a reduced splitting tree for ϕ. We consider a node v marked with $G_v = \langle V_v, E_v \rangle$. Let $E_{ext} = \{(x, y) \in E | x \in V \vee y \in V_v\}$ be a number of edges that has at least one end in the set V_v. We define a value $w(v) = |E_{ext} \setminus E_v|$ that is the number of removed edges that are incident to some vertices from V_v. A width of the tree is $W(T) = \max_v w(v)$, where the maximum is over all nodes of T.

Lemma 4. *For every connected graph $G = \langle V, E \rangle$ with the expansion $e(G)$ and for every reduced splitting tree of an unsatisfiable formula $\phi(G, f)$ the inequality $W(T) \geq e(G)$ holds.*

Proof. Let T be a reduced splitting tree. T contains a node v marked by $G_v = \langle V_v, E_v \rangle$ such that

- $|V_v| > \frac{2}{3} \cdot |V|$;
- v has two children;
- if u is a child of v, then $|V_u| \leq \frac{2}{3} \cdot |V|$.

There exists the node u, that is a child of v and $|V_u|$ is between $\frac{1}{3}|V|$ and $\frac{2}{3}|V|$. Thus by the definition of the expansion $w(u) \geq e(G)$. □

Lemma 5. *Let T be the reduced splitting tree for Tseitin formula $\phi(G)$, then there exists a reduced splitting tree T' for $\phi(G)$ such that $W(T') \leq k + \log_d F(T)$.*

Proof. By induction on the number of nodes in the tree T we show that if $F(T) \leq d^b$, then there exists a tree T' for $\phi(G)$ that $W(T') \leq k + b$. The base of induction is obvious.

Let the root r of T have only the one child v. Let T_v be a subtree of T with the root v. If $F(T) \leq d^b$, then $F(T_v) \leq d^{b-1}$. By the induction hypothesis we have a tree T'_v such that $W(T') \leq b - 1 + k$. We attach the tree T'_v to r and get a tree T' such that $W(T') \leq (b - 1 + k) + 1 = b + k$.

Let r have two children v_1 and v_2. Let T_1 and T_2 be subtrees with roots in v_1 and v_2 respectively; G_1 and G_2 are labels of v_1 and v_2 respectively. By the definition of F

$$F(T) = (d-1) \cdot F(T_1) + F(T_2) + 1,$$

We know that $d \cdot F(T_1) < F(T)$. Thus if $F(T) \leq d^b$, then $F(T_1) \leq d^{b-1}$ and $F(T_2) \leq d^b$. Therefore by the induction hypothesis there exist reduced splitting trees T_1' and T_2' for G_1 and G_2 respectively, such that $W(T_1') \leq k + b - 1$ and $W(T_2') \leq k + b$. We show that T_1' and T_2' may be used in the construction of such a reduced splitting tree T' for $\phi(G)$ that $W(T) \leq k + b$.

Let the root r of the tree be marked by G and children v_1 and v_2 be marked by connected components of $G \setminus e$. Let an edge e connect a vertex z from G_1 with a vertex y from G_2. We construct T' as follows.

We modify the tree T_2': to every label that contains the vertex y we attach a copy of the graph G_1 to y by the edge e. The original tree T_2' contains a leaf that is marked with the graph with only one vertex y; after the modification this leaf is marked with y with G_1 attached by means of the edge e. We make a splitting in this leaf on the variable x_e. We get a leaf that is marked with z and a leaf w that is marked with G_1. We attach a tree T_1' to w. So we get a reduced splitting tree T' for $\phi(G)$ such that $W(T') \leq \max\{W(T_2'), W(T_1') + 1, k\} \leq k + b$. □

Corollary 2. $e(G) \leq k + \log_d C(G)$.

Finally we prove the following theorem:

Theorem 3. *If degrees of all vertices of a graph G are at most k, then the size of the tree-like resolution proof of unsatisfiable Tseitin formula $\phi(G, f)$ is at least $d^{e(G)-k}$.*

5 Upper bound for CSP

We consider an arbitrary unsatisfiable CSP $\phi = \langle X, D, S \rangle$. Let $|D| = d$. For every constraint $C \in S$ we denote by $\mathrm{Vars}(C)$ the set of variables x such that C depends on x.

We construct a dependency graph $G = \langle V, E \rangle$ of CSP ϕ. Vertices of this graph correspond to constraints from S. Two constraints C_i and C_j are connected with $|\mathrm{Vars}(C_i) \cap \mathrm{Vars}(C_j)|$ edges, every edge is labelled with a common variable of C_i and C_j.

Note that a dependency graph of a Tseitin formula based on a graph H is isomorphic to H.

Theorem 4. *In the dependency graph $G = \langle V, E \rangle$ of an unsatisfiable CSP ϕ there is a subgraph H with the expansion $e(H) \geq \frac{\log_d S_T(\phi)}{\log \frac{3}{2} |V|}$, where $S_T(\phi)$ is a size of a minimal tree-like resolution refutation of ϕ.*

Proof. We consider the following backtracking algorithm $A(\phi)$

– It constructs a dependency graph $G = \langle V, E \rangle$ of ϕ.
– It finds a minimal cut $U \subseteq V$ such that $\frac{1}{3} \cdot |V| \leq |U| \leq \frac{2}{3} \cdot |V|$.
– For all variables that correspond to edges that connect U with its complement, algorithm A chooses them for splitting one by one.
– Now graph contains several connected components. The algorithm chooses an unsatisfiable component and makes a recursive call on it

Let $Time$ be a running time of the algorithm A; it equals to the size of some tree-like resolution proof of ϕ.

The execution protocol of A may be represented by a tree T with weighted edges (edges correspond to cuts and weights correspond to sizes of cuts). Vertices of the tree T are labelled with CSPs, that are passed in the recursive calls. Let the vertex v contain a formula ϕ and let the algorithm A find a cut U in the dependency graph of ϕ.

Let $X_{\phi,U}$ be the set of variables corresponding to edges in this cut. The weight of the edge that corresponds to this cut is $|X_{\phi,U}|$. A weighted height of the T is the maximal weight of the path from the root of T to a leaf. Let us denote the weighted height of T by h. Note that Time $\leq d^h$.

The number of vertices in the dependency graph of CSP in the child of T is at least $\frac{3}{2}$ times smaller than the number of vertices in the parent. Let a vertex u be the parent of a vertex v. Then the number of vertices in the dependency graph of the CSP in the vertex u is at least $\frac{3}{2}$ times the number of vertices in the dependency graph of the CSP in the vertex v

Let us denote the unweighted height of T by h_u; then h_u is at most $\log_{\frac{3}{2}} |V|$. Hence there exists an edge (v, u) with the weight at least $\frac{h}{\log_{\frac{3}{2}} |V|} \geq \frac{\log_d \text{Time}}{\log_{\frac{3}{2}} |V|}$.

Let CSP in v correspond to a dependency graph H. Therefore: $e(H) \geq \frac{\log_d \text{Time}}{\log_{\frac{3}{2}} |V|}$. □

Corollary 3. *Time* $\leq d^{e(H) \cdot \log_{\frac{3}{2}} |V|}$.

Corollary 4. *For an unsatisfiable Tseitin formula ϕ over the domain $D = \mathbb{Z}_d$ based on a graph $G = \langle V, E \rangle$ with degrees of its vertices at most k, there exists a subgraph H of G such that $S_T(\phi) \leq d^{e(H) \cdot \log_{\frac{3}{2}} |V|}$.*

Proof. The dependency graph of ϕ is isomorphic to G. Therefore by the Theorem 4 there is a subgraph H in G such that $S_T(\phi) \leq d^{e(H) \cdot \log_{\frac{3}{2}} |V|}$. □

Thus the minimal running time of a backtracking algorithm on a Tseitin formula based on a graph G satisfies inequalities $d^{e(G)-k} \leq \text{Time} \leq d^{e(H) \cdot \log_{\frac{3}{2}} |V|}$ for some subgraph H of G.

6 Open questions

– To prove (or refute) that there exists $c > 1$ such that the size of a dag-like resolution proof of a Tseitin formula based on a graph G is at least $c^{e(G)}$.

Such a lower bound exists if $e(G) = \Omega(n)$. This is also true for doubled graphs where every edge has a parallel copy.
- To reduce the gap between the upper and lower bounds.

Acknowledgements

The authors are grateful to Alexander Shen for drawing attention to the case of arbitrary alphabets and to anonymous referees for comments that improved the readability of the paper.

References

[AHI05] Michael Alekhnovich, Edward A. Hirsch, and Dmitry Itsykson. Exponential lower bounds for the running time of DPLL algorithms on satisfiable formulas. *J. Autom. Reason.*, 35(1-3):51–72, 2005.

[Bak95] Andrew B. Baker. Intelligent backtracking on constraint satisfaction problems: Experimental and theoretical results, 1995.

[BGIP99] Sam Buss, Dima Grigoriev, Russell Impagliazzo, and Toniann Pitassi. Linear gaps between degrees for the polynomial calculus modulo distinct primes. In *Journal of Computer and System Sciences*, pages 547–556, 1999.

[BSW01] E. Ben-Sasson and A. Wigderson. Short proofs are narrow — resolution made simple. *Journal of ACM*, 48(2):149–169, 2001.

[CEMT09] James Cook, Omid Etesami, Rachel Miller, and Luca Trevisan. Goldreich's one-way function candidate and myopic backtracking algorithms. In *Proceedings of TCC*, pages 521–538. Springer-Verlag, 2009.

[Hwa04] Cho Yee Joey Hwang. A Theoretical Comparison of Resolution Proof Systems for CSP Algorithms. Master's thesis, Simon Fraser University, 2004.

[IPS97] Russell Impagliazzo, Pavel Pudlak, and Jiri Sgall. Lower bounds for the polynomial calculus and the grobner basis algorithm. *Computational Complexity*, 8:127–144, 1997.

[IS12] D. Itsykson and D. Sokolov. The complexity of inversion of explicit Goldreichs function by DPLL algorithms. *Zapiski nauchnyh seminarov POMI*, 399:88–109, 2012.

[Its10] D. Itsykson. Lower bound on average-case complexity of inversion of Goldreich function by drunken backtracking algorithms. In *Proceedings of III International Computer Science Symposium in Russia*, volume 6072 of *Lecture Notes in Computer Science*, pages 204–215. Springer, 2010.

[Mit02a] David G. Mitchell. The resolution complexity of constraint satisfaction, 2002.

[Mit02b] David G. Mitchell. Resolution complexity of random constraints. In *Proceedings of the 8th International Conference on Principles and Practice of Constraint Programming*, CP '02, pages 295–309, London, UK, UK, 2002. Springer-Verlag.

[Mit03] David G. Mitchell. Resolution and constraint satisfaction. In *In Proceedings CP03*, pages 555–569. Springer, 2003.

[Tse68] G. S. Tseitin. On the complexity of derivation in the propositional calculus. *Zapiski nauchnykh seminarov LOMI*, 8:234–259, 1968. English translation of this volume: Consultants Bureau, N.Y., 1970, pp. 115–125.

[Urq87] A. Urquhart. Hard examples for resolution. *JACM*, 34(1):209–219, 1987.