Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 202 (2016)

Dag-like Communication and Its Applications

Dmitry Sokolov*

St. Petersburg Department of V.A. Steklov Institute of Mathematics of the Russian Academy of Sciences
27 Fontanka, St.Petersburg, 191023, Russia
sokolov.dmt@gmail.com

May 3, 2017

Abstract

In 1990 Karchmer and Widgerson considered the following communication problem `Bit`: Alice and Bob know a function $f : \{0,1\}^n \to \{0,1\}$, Alice receives a point $x \in f^{-1}(1)$, Bob receives $y \in f^{-1}(0)$, and their goal is to find a position i such that $x_i \neq y_i$. Karchmer and Wigderson proved that the minimal size of a boolean formula for the function f equals the size of the smallest communication protocol for the `Bit` relation. In this paper we consider a model of dag-like communication complexity (instead of classical one where protocols correspond to trees). We prove an analogue of Karchmer-Wigderson Theorem for this model and boolean circuits. We also consider a relation between this model and communication PLS games proposed by Razborov in 1995 and simplify the proof of Razborov's analogue of Karchmer-Wigderson Theorem for PLS games.

We also consider a dag-like analogue of real-valued communication protocols and adapt a lower bound technique for monotone real circuits to prove a lower bound for these protocols.

In 1997 Krajíček suggested an interpolation technique that allows to prove lower bounds on the lengths of resolution proofs and Cutting Plane proofs with small coefficients (CP*). Also in 2016 Krajíček adapted this technique to "random resolution". The base of this technique is an application of Razborov's theorem. We use real-valued dag-like communication protocols to generalize the ideas of this technique, which helps us to prove a lower bound on the Cutting Plane proof system (CP) and adapt it to "random CP".

Our notion of dag-like communication games allows us to use a Raz-McKenzie transformation [RM99, GP14], which yields a lower bound on the real monotone circuit size for the `CSP-SAT` problem.

*logic.pdmi.ras.ru/~sokolov

ISSN 1433-8092

1 Introduction

In 1990 Karchmer and Wigderson [KW90] introduced the following communication problem `Bit`: Alice receives a point u from a set $U \subseteq \{0,1\}^n$, Bob receives a point v from a set $V \subseteq \{0,1\}^n$, $U \cap V = \emptyset$, and their goal is to find a position i such that $u_i \neq v_i$. There is also a monotone version of this communication problem, called `MonBit`, in this case the goal of Alice and Bob is to find a position i such that $u_i = 1$ and $v_i = 0$. In [KW90] Karchmer and Wigderson proved the following Theorem: for every function f, there is a (monotone) boolean formula of size S iff there is a communication protocol of size S for the problem `Bit` (`MonBit`), where $U = f^{-1}(1)$ and $V = f^{-1}(0)$. Since then, a lot of results about the formula complexity of functions has been obtained by using this theorem, for example, a lower bound $2^{\Omega(\frac{n}{\log n})}$ on the monotone formula complexity for an explicit function [GP14], and a lower bound $n^{3-o(1)}$ on the formula complexity in de Morgan basis for an explicit function [DM16]. Karchmer-Wigderson Theorem gives a characterization of boolean formulas in terms of communication complexity, however, it does not work in the context of boolean circuits.

In 1995 Razborov [Raz95] introduced a model of communication *Polynomial Local Search* games (PLS). He gave a generalization of Karchmer-Wigderson Theorem replacing classical communication protocols by PLS games, and boolean formulas by boolean circuits. In this paper we consider a simplification of communication PLS games that is called boolean communication games (an analogue of this definition was also studied in [Pud10]). We show that for any communication problem there is a boolean communication game of size S iff there is a PLS game of size $\Theta(S)$ for the same communication problem. We also show a simple proof of a generalization of Karchmer-Wigderson result in the case of using boolean communication games and boolean circuits.

Razborov's result about the connection between PLS games and boolean circuits was used in 1997 by Krajíček [Kra97], who introduced a so-called "interpolation technique" for proving lower bounds on the size of propositional proof systems. In order to describe the essence of this technique let us consider a monotone function $f : \{0,1\}^n \to \{0,1\}$ from the class **NP** such that there is a lower bound $T(n)$ on the monotone circuit complexity of f. For example, one can use a function from [AB87]: let formula $\texttt{Zero}(x,r)$ encode with additional variables r, the fact that $x \in f^{-1}(0)$, and let formula $\texttt{One}(x,q)$ encode with additional variables q, the fact that $x \in f^{-1}(1)$. Krajíček has shown that if a proof system operates with clauses such that the communication complexity of evaluating these clauses (Alice knows the values of a part of variables, and Bob knows the values of the other part of variables) is bounded by parameter t, and in this proof system there is a proof of size S of the unsatisfiable formula $\texttt{Zero}(x,r) \wedge \texttt{One}(x,q)$, then one can create a PLS game of size $S \cdot 2^t$ for the Karchmer-Wigderson problem for function f. If the formulas `Zero` and `One` satisfy certain natural properties then this PLS game also solves a monotone version of the Karchmer-Wigderson problem for the function f. Thus we have a lower bound $S \geq \frac{T(n)}{2^t}$. There are proof systems for which lower bounds can be obtained by using this technique, for example, resolution, CP*, subsystems of LK, OBDD($\exists$, weakening) [Kra08]. However, if we cannot bound the parameter t then this technique does not give us any bounds, in particular we cannot use this technique for the CP proof system (without restrictions on the size of coefficients).

The second important communication problem is a canonical search problem $\texttt{Search}_\phi$

for an unsatisfiable formula $\phi(x, y)$ in CNF [BPS07]: Alice receives values for the variables x, Bob receives values for the variables y, and their goal is to find a clause of ϕ such that it is unsatisfied by this substitution. In the paper [BPS07], the authors present a technique of constructing communication protocols of size $\text{poly}(S)$ (in various classical communication models) for the $\texttt{Search}_\phi$ problem, where S is the size of a tree-like proof of ϕ in the proof system $\text{Th}(k)$ for fixed k that operates with polynomial inequalities of degree at most k over integer numbers. These proof systems cover a huge class of known proof systems (for example, CP is a special case of Th(1)). In [BPS07, HN12, GP14] the authors prove lower bounds on the communication complexity of the $\texttt{Search}_\phi$ problem and, as a corollary, a lower bound on the size of tree-like proofs in $\text{Th}(k)$. This technique allows to prove lower bounds only for tree-like versions of proof systems; general lower bounds are still unknown even for Th(2). Also in [GP14] the authors demonstrate a version of Raz-McKenzie transformation [RM99] that reduces the problem $\texttt{Search}_\phi$ to the problem $\texttt{MonBit}$ for a certain function $\texttt{SAT}_G$ (see Definition 7.1). As a corollary the authors obtain a lower bound on the monotone formula complexity of the function $\texttt{SAT}_G$.

Remark 1.1. Although in Krajíček's paper [Kra97] the problem $\texttt{Search}_\phi$ is not used, in fact all PLS games in that paper with little modification solve this problem. As a corollary, these games also solve the Karchmer-Wigderson problem.

In this paper we also consider real communication games that generalize boolean communication games (which are a dag-like analogue of real-valued classical communication protocols [Kra98]). We prove an analogue of Krajíček's Theorem: we show how to construct a real communication game of size S for the problem $\texttt{Search}_\phi$ from a proof of ϕ in the CP proof system (and, as a corollary, from a proof in any proof system used in Krajíček's paper). Instead of constructing a circuit from a game we directly give a lower bound for real communication protocols. This result generalizes Cook and Haken's result [HC99] for monotone real circuits. As a corollary of this result we apply a Raz-McKenzie transformation and obtain a lower bound on the monotone real circuit size of the function $\texttt{SAT}_G$.

In [BKT14] the authors introduce a *random resolution* proof system. A δ-random resolution proof distribution for a formula ϕ is a random distribution (π_s, Δ_s) such that Δ_s is a CNF formula, π_s is a resolution proof of $\phi \wedge \Delta_s$, and every fixed truth assignment of all variables satisfies Δ_s with probability at least $1 - \delta$. We can consider a natural generalization of this definition to other proof systems and look at lower bounds for it. The only known technique for proving lower bounds for the CP proof system is the reduction, due to Pudlák [Pud97], to lower bounds on the size of real monotone circuits; Hrubeš [Hru13] generalizes this technique for the semantic version of CP. The exponential lower bounds on these circuits are given in [Pud97] and [HC99]. The reduction of lower bounds on the CP proof size to lower bounds on the size of real monotone circuits uses substantially the structure of the initial formula, and so it is unclear how to generalize them for a *random* CP proof system. In this paper we show that lower bounds that are obtained by using real communication games can be generalized for *random* CP by using a technique that has been recently introduced by Krajíček in [Kra16]. Unfortunately, this technique gives us a lower bound only for small values of the parameter δ.

Organization of the paper. In Section 2 we give definitions of boolean and real communication games and prove basic properties of these games. In Section 3 we define PLS games and prove a relation between PLS games and boolean communication games, also we give a simplification of Razborov's Theorem. In Section 4 we consider a construction of communication games from semantic CP proofs. In Section 5 we give a lower bound on the size of real communication games. In Section 6 we prove a lower bound on *random* CP proof system. In Section 7 we give a lower bound on the real circuit complexity of the function $\mathtt{SAT}_G$.

Remark 1.2. Definition 2.1 was introduced independently by Pavel Pudlák and Pavel Hrubeš. Also Pavel Pudlák in a private communication announced a proof of the opposite direction of the statement of Lemma 2.2.

2 Preliminaries

2.1 Games

The following definition has been also independently introduced by Pavel Pudlák and Pavel Hrubeš.

Definition 2.1. Let $U, V \in \{0,1\}^n$ be two sets. Let us consider a triple (H, A, B), where H is a directed acyclic graph, $A : H \times U \to \mathbb{R}$ and $B : H \times V \to \mathbb{R}$. We say that a vertex $v \in H$ is valid for a pair (x, y) iff $A(v, x) > B(v, y)$. We call this triple a *real communication game* for the pair (U, V) and some relation $N : U \times V \times T \to \{0, 1\}$, where T is a finite set of "possible answers", if the following holds:

- H is an acyclic graph and the out-degree of all its vertices is at most 2;
- the leaves of H are marked by element of T;
- there is a *root* $s \in H$ with in-degree 0 and this vertex is valid for all pairs from $U \times V$;
- if $v \in H$ is valid for (x, y) and v is not a leaf then at least one child of v is valid for (x, y);
- if $v \in H$ is valid for (x, y), v is a leaf and v is marked by $t \in T$ then $N(x, y, t) = 1$.

The size of the game is the size of the graph H.

We call it a *boolean communication game* if $A : H \times U \to \{0, 1\}$ and $B : H \times V \to \{0, 1\}$. (An analogue of boolean communication games was studied in [Pud10]).

Remark 2.1. It is useful to think that if we have a boolean communication game (H, A, B) for sets U, V then we mark each vertex $h \in H$ by rectangle $R_h \in U \times V$ of *valid inputs*, where $(x, y) \in R_h$ iff $A(h, x) = 1$ and $B(y, h) = 0$. So, if s is the root then it is marked by the rectangle $U \times V$. If h has two children h' and h'', then $R_h \subseteq R_{h'} \cup R_{h''}$.

Lemma 2.1. Let $U, V \in \{0,1\}^n$. If Alice receives $x \in U$, Bob receives $y \in V$ and we have a classical communication protocol of size S for some relation N, then we have a boolean communication game (H, A, B) of size S for sets U, V and relation N. Moreover, H is a tree.

Proof. Let us consider a tree K that corresponds to a classical communication protocol. Vertices of this tree correspond to the values of transmitted bits. We consider a vertex $k \in K$ and mark it by rectangle $R_k \in U \times V$, where $(x, y) \in R_k$ iff we run protocol on inputs x, y and come to vertex k at some moment. This tree with rectangles defines a boolean communication game (see Remark 2.1), the root of this tree is the root of the game. All required properties follow from the definition of rectangles R_k. □

Definition 2.2. Let $\phi(x, y)$ be an unsatisfiable CNF formula, U be an arbitrary subset of assignments to variables x, and V be an arbitrary subset of assignments to variables y. A canonical search problem (relation) $\mathtt{Search}_\phi : U \times V \times C \to \{0, 1\}$, where C is the set of clauses of formula ϕ, contains all triples (u, v, c) such that $c(u, v) = 0$.

Definition 2.3. Let $U, V \subseteq \{0, 1\}^n$, $U \cap V = \emptyset$. Relation $\mathtt{Bit}_{U,V} : U \times V \times [n] \to \{0, 1\}$ contains all triples (u, v, i) such that $u_i \neq v_i$. If there is a function f such that $U = f^{-1}(1)$ and $V = f^{-1}(0)$ we write $\mathtt{Bit}_f$.

Let $U, V \subseteq \{0, 1\}^n$, $U \cap V = \emptyset$ and $\forall\ x \in U,\ y \in V\ \exists i\ x_i = 1 \wedge y_i = 0$. Relation $\mathtt{MonBit}_{U,V} : U \times V \times [n] \to \{0, 1\}$ contains all triples (u, v, i) such that $u_i = 1 \wedge v_i = 0$. If there is a monotone function f such that $U = f^{-1}(1)$ and $V = f^{-1}(0)$ we write $\mathtt{MonBit}_f$.

Lemma 2.2. Let $f : \{0, 1\}^n \to \{0, 1\}$ be a monotone function. If there is a monotone (boolean) real circuit for f of size S then there is a (boolean) real communication game of size S for sets $(f^{-1}(1), f^{-1}(0))$ and relation $\mathtt{MonBit}_f$.

Proof. A graph H of our real communication game is a graph of the minimal monotone real circuit for function f with inverted edges. $A(e, u)$ returns the value of the gate that corresponds to the vertex e on the input u. We define $B(e, v)$ in the same way. If a leaf $h \in H$ corresponds to an input variable x_i then mark this leaf by i.

Let us check all the required properties:

- H is an acyclic and all leaves are marked;
- the root $s \in H$ corresponds to the output gate of the circuit;
- note that $A(s, f^{-1}(1)) = 1$ and $B(s, f^{-1}(0)) = 0$, hence the root is valid for all pairs from $f^{-1}(1) \times f^{-1}(0)$;
- if $h \in H$ is an inner vertex and $A(h, u) > B(h, v)$, then it has a child h' such that $A(h', u) > B(h', v)$ since a gate that corresponds to h computes a monotone function;
- if $h \in H$ is a leaf with label i then $A(h, u) = u_i$ and $B(h, v) = v_i$. Hence if $A(h, u) > B(h, v)$ then $u_i = 1$ and $v_i = 0$.

□

2.2 Semantic Cutting Planes

We consider a semantic version of the Cutting Plane (CP) proof system.

Definition 2.4 (Hrubeš [Hru13])**.** A proof in semantic CP for CNF formula ϕ is a sequence of linear inequalities with real coefficients $C_1, C_2, \ldots, C_k$, such that C_k is the trivially unsatisfiable inequality $0 \geq 1$ and C_i can be obtained by one of the following rules:

- C_i is a linear inequality that encodes a clause of formula ϕ;
- C_i semantically follows on $\{0,1\}$ values from C_j, C_k where $j, k < i$.

The size of proof is the number of inequalities k. We say that we have a proof in CP* if coefficients in the proof are *integer* and bounded by a polynomial in the number of variables of ϕ.

2.3 Broken Mosquito Screen

Definition 2.5 (Cook, Haken [HC99])**.** An instance of the Broken Mosquito Screen (BMS) problem encodes a graph with m^2-2 vertices, where $m \geq 3$ is a convenient parameter for indexing. The graphs are represented in a standard way, as a string of bits that indicates for each pair of vertices whether there is an edge between them, with value 1 for the edge being present and value 0 for the edge being absent.

The graph is *good*, or accepted, if there is a partition of its vertices into $m-1$ sets of size m and one set of size $m-2$ such that each of these subsets forms a clique. A graph is *bad*, or rejected if there is a partition of its vertices into $m-1$ sets of size m and one set of size $m-2$ such that each of these subsets forms an anticlique.

Lemma 2.3 (Cook, Haken [HC99])**.** No instance of BMS can be good and bad simultaneously. Furthermore, each element in good set is not less (as a vector) than any element in bad set.

Definition 2.6 (Cook, Haken [HC99])**.** Let G_0 be a set of good instances of the BMS problem that are minimal: only the edges that are explicitly needed to meet the acceptance condition are present. Let B_0 be a set of bad instances of the BMS problem that are maximal: all edges are present except those that are explicitly required to be absent to meet the rejection condition.

Now we describe unsatisfiable formulas that are based on the BMS problem. $\texttt{BMS}(x,q,r) = Part(x,q) \wedge Part(\neg x, r)$, where $x \in \{0,1\}^{(m^2-2)(m^2-3)/2}$ are variables that correspond to a graph, $\neg x$ means that we substitute the negation of the respective literals, $q = \{q_{ijk} \mid i,j \in [m], k \in [m^2-2]\}, r = \{r_{ijk} \mid i,j \in [m], k \in [m^2-2]\}$. $Part(x,y)$ equals true iff x is a good instance of BMS problem, $y_{ijk} = 1$ iff we put a vertex k on the j-th place in the i-th component, and the formula $Part(x,y)$ consists of the following clauses:

- $\forall i,j \in [m], k_1, k_2 \in [m^2-2], k_1 \neq k_2 : \quad (\neg y_{ijk_1} \vee \neg y_{ijk_2})$;
- $\forall i < m, j \leq m : \quad \bigvee_{k \in [m^2-2]} y_{ijk}$;
- $\forall j \leq m-2 : \quad \bigvee_{k \in [m^2-2]} y_{mjk}$;
- $\forall j \in \{m-1, m\}, k \in [m^2-2] : \quad (\neg y_{mjk})$;

- $\forall i, j_1 < j_2, k_1 \neq k_2 : \quad (\neg y_{ij_1k_1} \vee \neg y_{ij_2k_2} \vee x_{k_1k_2})$.

We also need a variant of this formula in 3-CNF, denote it by $Part'$. It can be obtained by replacing long clauses by a standard procedure: if we have a clause C of the form $(a \vee b \vee c \vee D)$ then we replace it by two new clauses $(a \vee b \vee \ell)$ and $(\neg\ell \vee D)$, where ℓ is a new variable.

$\texttt{BMS}'(x, q, r, z) = Part'(x, q, z) \wedge Part'(\neg x, r, z)$.

3 Bit relation and circuits

In this section we prove a generalization of Kachmer-Wigderson Theorem. This Theorem relates the size of classic communication protocols for the relation Bit to the size of boolean formulas. We prove a similar result for boolean communication games and boolean circuits. We also consider a model of PLS communication games [Raz95] with a fixed graph and prove its equivalence to boolean communication games, hence we give a simple proof of Razborov's Theorem about the relation between communication PLS games and boolean circuits.

3.1 PLS games and boolean circuits

We start with a model of PLS games. We use a bit simpler notion of PLS games from Krajíček's paper [Kra97], where the graph of game is fixed.

Definition 3.1 ([Raz95, Kra97])**.** Let $U, V \in \{0,1\}^n$ be two sets and let $N : U \times V \times T \to \{0,1\}$ be a relation, where T is a finite set of "possible answers". A communication PLS game for sets U, V and relation N is a labelled directed graph G satisfying the following four conditions:

- G is acyclic and has a root (the in-degree 0 node) denoted $\emptyset$;
- each leaf is labelled by some $t \in T$;
- there is a function $S(g, x, y)$ (the strategy) that given a node $g \in G$ and a pair $x \in U, y \in V$, outputs the end of an edge leaving the node g;
- for every $x \in U, y \in V$, there is a set $F(x, y) \in G$ such that:
  - $\emptyset \in F(x, y)$;
  - if $g \in F(x, y)$ is not a leaf then $S(g, x, y) \in F(x, y)$;
  - if $g \in F(x, y)$ is a leaf and it is marked by $t \in T$ then $N(x, y, t) = 1$.

The communication complexity of G is the minimal number t such that for every $g \in G$ the players (one knowing x and g, the other one y and g) decide whether $g \in F(x, y)$ and compute $S(g, x, y)$ with at most t bits exchanged in the worst case. The size of the game is defined as $|G|$.

Remark 3.1. We remove the cost function from the original definition in [Raz95] since if a graph is fixed then the cost function can be replaced by the topology number of vertex.

Theorem 3.1. Let $U, V \subseteq \{0,1\}^n$, and $N : U \times V \times \mathbb{N} \to \{0,1\}$ be a relation.

1. If there is a communication PLS game of size L and communication complexity t for sets U, V and a relation N then there is a boolean communication game of size at most $L \cdot 2^{3t}$ for the same sets and relation.

2. If there is a boolean communication game of size L for sets U, V and a relation N then there is a communication PLS game of size L and communication complexity two for the same sets and relation.

Proof. We start with the easy direction. If there is a boolean communication game (H, A, B) then one define a PLS game as follows:

- $G = H$ and $\emptyset$ equals the root of the communication game;
- $v \in F(x, y)$ iff $A(v, x) > B(v, y)$;
- if v is not a leaf of H then $S(v, x, y)$ returns a child v' of v in H such that $A(v', x) > B(v', y)$;
- if v is a leaf of H then $S(v, x, y)$ returns v and it is marked by the same element as in boolean communication game.

The required properties straightforwardly follow from similar properties of the communication game. The communication complexity of F, S is bounded by 2 since it is enough to send $A(v, x)$ and $B(v, y)$ to check whether v in $F(x, y)$ and it is enough to calculate which child of v in $F(x, y)$ to calculate S. So the size of PLS game is at most $|H|$ and communication complexity is at most 2.

Now we assume that we have a PLS game G of communication complexity t. We create a graph H for our boolean communication game and mark every vertex $h \in H$ by some rectangle $R_h \subseteq U \times V$ (see Remark 2.1). If $R \subseteq U \times V$ and there is a vertex $h \in H$ that is marked by R then we write that $(R) \in H$.

Consider sets $R_g = \{(x, y) \in U \times V \mid g \in F(x, y)\}$. A classical communication protocol that computes $F(x, y)$ defines at most 2^t rectangles $R_{g,i} \subseteq U \times V$ such that $R_g = \bigcup_i R_{g,i}$. We add to our graph H vertices that correspond to such rectangles for all g and i, so we have at most $|G| \cdot 2^t$ vertices. Since for all $x \in U, y \in V$ it holds that $\emptyset \in F(x, y)$, there is a vertex that is marked by rectangle $(U \times V)$, we say that it is a root of our graph H. Also note, that if $g \in G$ is a leaf then we say that all vertices $(R_{g,i}) \in H$ are leaves of H and mark it by the same label that g is marked.

Now our goal is to establish a connection between these vertices. Let us consider a rectangle $R_{g,i} = U_{g,i} \times V_{g,i}$ for some internal node $g \in G$ and some number i and a function $\texttt{Next}_{g,i}$ that takes a point $(x, y) \in R_{g,i}$ and returns a rectangle $R_{g',j}$ such that $g' = S(g, x, y)$ and $(x, y) \in R_{g',j}$. We have a classical communication protocol that can find a vertex $g' \in G$, and if we know g' we can use a protocol that decides whether $g' \in F(x, y)$ to find a specific rectangle $R_{g',j}$. Hence we have a classical communication protocol that can find "next rectangle" in at most $2t$ rounds. By Lemma 2.1 we have a boolean communication game $(H_{g,i}, A_{g,i}, B_{g,i})$ for sets $U_{g,i}$, $V_{g,i}$ and function $\texttt{Next}_{g,i}$ of size 2^{2t}. For all g and i we add the vertices of $H_{g,i}$ to H. Since the root of $H_{g,i}$ is marked

by $R_{g,i}$ we can identify it with the vertex $(R_{g,i}) \in H$, and since in leaf $t \in H_{g,i}$ we know the value of function $\texttt{Next}_{g,i}$ it is marked by some rectangles $R_t \subseteq \texttt{Next}_{g,i}(x,y) = R_{g',j}$ for some (x,y) and we can identify this leaf with $(R_{g',j})$. At this step we add at most $|G| \cdot 2^{3t}$ vertices.

Let us check the properties:

- the root is marked by $U \times V$ hence it is valid for all points $(x,y) \in U \times V$;
- all leaves are marked by answers of N;
- if $(R_{g,i})$ is a leaf and $(x,y) \in R_{g,i}$ then $(R_{g,i})$ is marked by a correct answer since g is a leaf of the PLS game and $g \in F(x,y)$;
- if $h \in H$ is not a leaf and $(x,y) \in R_h$ then there is at least one child h' of h such that $(x,y) \in R'_v$ since all games for $\texttt{Next}$ are correct.

□

3.2 Games and circuits

The proof of the following theorem generalizes a result from [KW90] and uses a similar proof strategy. The sketch of circuits construction from protocols was given in [Pud10] (Lemma 1), for protocols construction from circuits we will use combination of Razborov's Theorem [Raz95] and Theorem 3.1.

Theorem 3.2. Let $f : \{0,1\}^n \to \{0,1\}$ be a function. There is a boolean communication game for $\texttt{Bit}_f$ of size S iff there is a circuit for f of size $O(S)$. Moreover there is a boolean communication game for $\texttt{MonBit}_f$ of size S iff there is a monotone circuit for f of size S.

Proof. Consider a boolean communication game (H, A, B) for $\texttt{Bit}_f$ of size S. Each vertex $h \in H$ is marked by a rectangle $R_h = U_h \times V_h$ (see Remark 2.1). We define circuits $f_h : \{0,1\}^n \to \{0,1\}$ such that $f_h(U_h) = 1$ and $f_h(V_h) = 0$ by induction from leaves to root. Since the root $h_0 \in H$ is marked by $R_{h_0} = (f^{-1}(1), f^{-1}(0))$, the function f_{h_0} equals f.

If $h \in H$ is a leaf then it is marked by $i \in [n]$ such that for all $x \in U_h$ and $y \in V_h$ it holds that $x_i \neq y_i$, moreover, since $U_h \times V_h$ forms a rectangle, a specific value of x_i corresponds to h. So if $x_i = 1$ then $f_h(z) = z_i$ else $f_h(z) = \neg z_i$.

If $h \in H$ is not a leaf and h', h'' are children of h then $R_h \subseteq R_{h'} \cup R_{h''}$; moreover, one of the following cases holds:

- $U_h \subseteq U_{h'} \cap U_{h''}$;
- $V_h \subseteq V_{h'} \cap V_{h''}$;
- $R_h \subseteq R_{h'}$ or $R_h \subseteq R_{h''}$

since $R_h, R_{h'}, R_{h''}$ are rectangles (if not then for example $U_h \nsubseteq U_{h'}$ and $V_h \nsubseteq V_{h''}$ and we can pick some $u \in U_h \setminus U_{h'}$ and $v \in V_h \setminus V_{h''}$, so the point (u,v) is not covered, the other cases can be considered in a similar way). Let us consider the first case $U_h \subseteq U_{h'} \cap U_{h''}$ and define $f_h = f_{h'} \wedge f_{h''}$. If $x \in U_h$ then $f_{h'}(x) = f_{h''}(x) = 1$, if $x \in V_h$ then $f_{h'}(x) = 0$

or $f_{h''}(x) = 0$. In the second case $V_h \subseteq V_{h'} \cap V_{h''}$ define $f_h = f_{h'} \vee f_{h''}$. If $x \in V_h$ then $f_{h'}(x) = f_{h''}(x) = 0$, if $x \in U_h$ then $f_{h'}(x) = 1$ or $f_{h''}(x) = 1$. In the last case we have already had a function that is correct on R_h.

The circuit that we thus constructed for the root computes f and consists of at most $|H|$ and/or gates, all negations are in its inputs. Note that we use negations only if we have an answer $i \in [n]$ such that $x_i = 0 \wedge y_i = 1$ hence we do not need negations if we have a game for $\texttt{MonBit}_f$.

A proof in the other direction in the monotone case follows from Lemma 2.2. In the nonmonotone case we create a PLS game with a graph of size S and communication complexity 2. A graph G of our boolean communication game is a graph of the minimal monotone real circuit for function f with inverted edges; $g \in F(x, y)$ iff the corresponding gate returns different values of x and y. If $g \in F(x, y)$ then $S(g, x, y)$ chooses a child of g that is in $F(x, y)$, there should be at least one because the gate g returns different answers on x and y. A leaf in $F(x, y)$ corresponds to a variable that has different values on the inputs x and y. By Theorem 3.1 there is a boolean communication game of size $O(S)$. □

Corollary 3.1 ([Raz95])**.** Let $f : \{0,1\}^n \to \{0,1\}$ be a function. If there is a PLS communication game for $(\texttt{MonBit}_f)$ $\texttt{Bit}_f$ of size S and communication complexity t then there is a (monotone) circuit for f of size $S \cdot 2^{O(t)}$. If there is a (monotone) circuit for f of size S then there is a PLS communication game for $(\texttt{MonBit}_f)$ $\texttt{Bit}_f$ of size S and communication complexity two.

Proof. Follows from Theorem 3.1 and Theorem 3.2. □

4 From Proofs to Games

In this section we relate real communication games to proofs in the semantic CP proof system.

At first we consider a connection between MonBit relation and $\texttt{Search}_\phi$ problem.

Lemma 4.1. Let $U, V \subseteq \{0,1\}^n$, $U \cap V = \emptyset$ and $\forall\, x \in U, y \in V\ \exists i\ x_i = 1 \wedge y_i = 0$. Let $Q(z, q)$ be a boolean CNF formula such that $x \in U$ iff the formula $\exists q\ Q(x, q)$ is true. Let $R(z, r)$ be a boolean CNF formula that satisfies the following properties:

- there is at most one variable z in each clause;
- all variables z occur with negative signs;
- $y \in V$ iff the formula $\exists r\ R(y, r)$ is true.

For each $x \in U$ one can fix arbitrary q^x such that $Q(x, q^x) = 1$, and for each $y \in V$ fix arbitrary r^y such that $R(y, r^y) = 1$. Let $L = \{(x, q^x) \mid x \in U\}$, and $L' = \{(y, r^y) \mid y \in V\}$. If there is a real (boolean) communication game for sets L, L' and $\texttt{Search}_{Q(z,q)\wedge R(z,r)}$ of size S then there is a real (boolean) communication game for sets U, V and $\texttt{MonBit}_{U,V}$ of size S.

Proof. We have a real communication game (H, A, B) of size S for sets of substitution (x, q^x) and (r^y). Note, that function A depends only on a vertex of graph H and variable x, and function B depends only on vertex of graph H and variable y, thus we consider this games as a game for sets U, V. If we remove all vertices that is not valid for any pair $(x, y) \in U \times V$ then the game will remain correct. Leaves of this game are marked by clauses that are falsified by a substitution (x, q^x, r^y) for some (x, y), but this clause cannot be from formula Q by the choose of q^x, hence it is a clause from formula R that is not satisfied by r^y, thus this clause contains a variable z_i and $y_i = 0 \wedge x_i = 1$ because z_i has a negative sign and it is not satisfied by x_i, but it satisfied by y_i by the choice of r^y. □

Lemma 4.2. Let $\phi(x, y)$ be an unsatisfiable CNF formula, U be an arbitrary subset of substitutions to variables x and V is an arbitrary subset of substitutions to variables y. If there is a semantic CP proof of this formula of size S then there is a real communication game of size S for the sets (U, V) and the canonical search problem $\texttt{Search}_\phi$.

Proof. Let H be the graph of the semantic CP proof of the formula ϕ with inverted edges. There is a correspondence between vertices and inequalities of the proof. Consider a vertex $h \in H$, this vertex corresponds to inequalities $f(x) + \ell(y) \geq c$, define the functions A, B in the following way $A(h, u) = -f(u)$ and $B(h, v) = \ell(v) - c$. Note that a vertex is valid for pair (u, v) iff $A(h, u) > B(h, v)$, hence $f(u) + \ell(v) < c$, i.e. in this case the inequality is falsified by the substitution (x, y).

The root of our game corresponds to the trivially false inequality $0 \geq 1$, hence the root is valid for any pair $(u, v) \in U \times V$. If a substitution satisfies all inequalities in the children of some vertex $h \in H$ then this substitution satisfies the inequality in h. Thus, if h is valid for some pair then at least one child of h is valid for this pair.

If a leaf h is valid for the pair (u, v) then the inequality in h is falsified by the substitution (u, v). □

Lemma 4.3. Let $U, V \subseteq \{0, 1\}^n$, $U \cap V = \emptyset$ and $\forall\, x \in U, y \in V\ \exists i\ x_i = 1 \wedge y_i = 0$. Let $Q(z, q)$ be a boolean CNF formula such that $x \in U$ iff the formula $\exists q\ Q(x, q)$ is true. Let $R(z, r)$ be a boolean CNF formula that satisfies the following properties:

- there is at most one variable z in each clause;
- all variables z occur with negative signs;
- $y \in V$ iff the formula $\exists r\ R(y, r)$ is true.

If there is a proof of formula $Q(z, q) \wedge R(z, r)$ in semantic CP of size S then there is a real communication game for (U, V) and relation $\texttt{MonBit}_{U,V}$ of size S.

Proof. Follows from Lemmas 4.2 and 4.1. □

5 Lower bound

We remind that G_0 is the set of minimal good instances of BMS and B_0 is the set of maximal bad instances of BMS.

Lemma 5.1 ([HC99], Section 4.4)**.** $|G_0| = |B_0| = \frac{(m^2-2)!}{(m!)^{m-1}(m-2)!(m-1)!}$.

For the rest of the section we fix some subsets $U_0 \subseteq G_0, V_0 \subseteq B_0$ of size at least $\frac{|G_0|}{2}$, w.l.o.g. $|U_0| = |V_0|$.

Theorem 5.1. The size of any real communication game for pair U_0, V_0 and relation $\texttt{MonBit}_{U_0,V_0}$ is at least $\frac{1.8^{\sqrt{m/8}}}{4}$.

Before we prove this theorem we need to present a notion of fences [HC99]. For the rest of this section we fix some real communication game (H, A, B) for pair (U_0, V_0) and relation $\texttt{MonBit}_{U_0,V_0}$. Our goal is to construct a partial map $\mu : (U_0 \cup V_0) \to H$ such that the domain of μ is big enough and the size of preimage of any element of H is small. We create this map step by step. At the step $i \in 0, 1, \ldots$ (we say that i is the current time) we consider the sets $U_i \subseteq U_0$, $V_i \subseteq V_0$ and pick some element $g \in U_i \cup V_i$ and put it to some vertex from H, after that we increase the time and proceed with sets $U_{i+1} = U_i \setminus \{g\}$ and $V_{i+1} = V_i \setminus \{g\}$. Note that either $U_{i+1} = U_i$ or $V_{i+1} = V_i$.

Definition 5.1. Let h be a vertex in a real communication game (H, A, B) and let $g \in U_i$. A *fence* around graph g in the vertex h at time i is a conjunction $C = z_1 \wedge \cdots \wedge z_q$ where $z_1, \ldots, z_q$ are bits of the input of BMS problem. Furthermore, $C(g) = 1$, and if h is a valid vertex for pair (g, g') for some $g' \in V_i$ then $C(g') = 0$. The length of fence is the number of variables q. A minimal fence around g in h at time i is a fence of minimal length around g in h at time i.

Dually, a fence around $g \in V_i$ in h at time i is a disjunction $D = z_1 \vee \cdots \vee z_q$, where $z_1, \ldots, z_q$ bits of the input of BMS problem. Furthermore, $D(g) = 0$ and if h is a valid vertex for pair (g', g) for some $g' \in U_i$ then $D(g') = 1$.

Proposition 5.1. The length of minimal fence around $g \in U_0 \cup B_0$ in h is not increasing in time.

Proof. Follows from the definition of fence. □

Definition 5.2 ([HC99])**.** Let $k = \frac{m}{2}$. We call a fence long if it is longer than $\frac{k}{2}$, otherwise we call it short.

5.1 Construction of a mapping μ

Definition 5.3 ([HC99])**.** Let us fix some topological sorting of the graph H so that the children of some vertex $h \in H$ have bigger numbers than h. At time i let $h_i \in H$ be the vertex with the maximum topological number such that there is a graph $d_i \in G_i \cup B_i$ such that d_i requires a long fence at h_i at time i. Define $\mu(d_i) = h_i$ and delete d_i from $G_i \cup B_i$ to get $G_{i+1} \cup B_{i+1}$ (if there is more than one such d_i then we choose some from G_i first). This process stops when the remaining graphs have short fences at all gates.

The following lemmas are proved by analogy with the paper [HC99].

Lemma 5.2 ([HC99], Lemma 2)**.** The size of the domain of μ is at least $|U_0|$.

Proof. Let us consider two cases.

Case 1. If all the graphs in U_0 or all the graphs in V_0 are mapped by μ then the Lemma holds.

Case 2. Let $U_i \cup V_i$ be a set of unmapped graphs at the time i when the definition of μ no longer can be continued. Let us consider the root of the game s and $b \in V_i$. There exists a short fence D around b at the vertex s at time i.

Let D be $x_1 \vee x_2 \vee \cdots \vee x_\ell$, where $\ell \leq \frac{k}{2}$. Therefore all graphs in U_i contain at least one edge from this disjunction. The fraction of graphs in G_0 that contain the edge x_1 is less than $\frac{1}{m}$. When a good graph is known to contain edge x_1 it means the two endpoints of the edge are in the same subset of the partition. If two vertices are chosen randomly, the chance of the second vertex being in the same subset as the first is the number of other vertices in that subset divided by the number of other vertices in the graph. That fraction is $\frac{m-1}{m^2-3}$ or less, which is less than $\frac{1}{m}$ (for $m > 3$). So the fraction of G_0 that contains any of the ℓ literals in D must be less than $\frac{\ell}{m}$, which is less than $\frac{1}{4}$. Therefore, the size of U_i is less than $\frac{|G_0|}{4} \leq \frac{|U_0|}{2}$.

A dual argument says that $|V_i| < \frac{|V_0|}{2}$, so the size of the mapped set is at least $\frac{|U_0|}{2} + \frac{|V_0|}{2} = |U_0|$. □

Lemma 5.3. Let $h, h_1, \ldots, h_\ell \in H$ and $h_1, \ldots, h_\ell$ be children of the vertex h, and at time i some graph $d \in U_i \cup V_i$ has a fence of length s_j at h_j then d has a fence of length at most $\sum s_j$ at h at time i.

Proof. If $d \in U_i$ we can consider a conjunction of all fences at vertices h_j. This conjunction equals 1 on d, and if a pair (d, d') is valid for h then it is valid for some h_i, hence this conjunction equals to 0 on d'. Analogously if $d \in V_i$ we can consider a disjunction of all fences at vertices h_j. □

The next lemma is an analogue of Lemma 4 from [HC99].

Lemma 5.4. The number of graphs from $U_0 \cup V_0$ that can be mapped by μ to any single $h \in H$ is at most

$$2\frac{(km)^{r/2}(m^2-m)^{r/2}(m^2-2-r)!}{(m!)^{m-1}(m-2)!(m-1)!},$$

where r is the greatest even number that is less or equal to $\sqrt{\frac{m}{2}}$.

Proof. This argument gives an upper bound on how many good graphs are mapped to h. By symmetry, the number of bad graphs mapped to h satisfies the same bound. Thus the initial factor 2 in the formula.

1. Let g be the first good graph mapped to h and g is mapped at time i. Let $d_1, \ldots, d_s$ be a complete list of the graphs in V_i, listed in the order of their value of the function $B(h, \cdot)$ so that $B(h, d_1) \leq B(h, d_2) \leq \cdots \leq B(h, d_s)$.

2. Each of these bad graphs d_j has a short fence at each child of h at time i, else d_j could be mapped to vertex of H with bigger topological number than h. Since any vertex has at most two children, by Lemma 5.3 d_j has a fence of size at most k at h. Let $D_j = (z_{j,1} \vee \cdots \vee z_{j,k})$, where literals might be repeated in D_j if there are less than k distinct ones.

3. Let the graph $g' \in U_0$ be also mapped to h, so it evaluates to 1 all those fences D_j such that $B(h, d_j) < A(h, g')$. Suppose this condition is satisfied for the first t graphs d_j, so $B(h, d_{t+1}) \geq A(h, g')$. Select one literal from each fence D_j for j from 1 to t which represents an edge in g'. The conjunction of this set of literals is a fence for g' at time i (and hence at any bigger time) and therefore must include more than $\frac{k}{2}$ distinct literals.

4. Note that a list of more than $\frac{k}{2}$ distinct edges must contain r different endpoints where $r > \sqrt{\frac{m}{2}}$. It is convenient for r to be even, so subtract 1 if r is odd.

5. The denominator of the formula in the statement of the theorem is the number of the orderings of the equally sized subsets and the ordering of the vertices within a subset is immaterial for the partition. The calculation counts ordered partitions, so it overcounts unordered partitions by a factor of $(m!)^{m-1}(m-2)!(m-1)!$.

6. To count how many ways are there to choose a graph g so that $\mu(g) = h$, proceed by choosing edges, and thereby vertices, so that the fences $D_1, D_2, \ldots$ are satisfied until r vertices have been chosen.

7. In the case of D_1, one of at most k different edges $z_{1,1}$ to $z_{1,k}$ can be chosen. That choice dictates that the two endpoints are in the same subset of the partition. There are m of these subsets, and to justify dividing the formula by the denominator, any of the subsets must be possible for the two vertices. Furthermore, within the subset, any of the $m(m-1)$ ordering positions for the two chosen vertices must be possible. So, for the first two vertices chosen there are only $km(m^2 - m)$ choices.

8. When satisfying fence D_j, several things can happen: if there are already two vertices v_1, v_2 chosen to be in the same subset of the partition, and the literal representing the edge from v_1 to v_2 is one of the disjuncts in D_j, then no vertices are added to the partition and the procedure moves on to D_{j+1}. Otherwise, one of the edges $z_{j,1}$ to $z_{j,k}$ must be chosen. It might be impossible to make such a choice if all the edges in D_j run between vertices that are already assigned to different subsets. In that case, the partition is abandoned as an instance of overcounting the graphs that can be mapped to h.

9. In case an edge from D_j can be chosen, the choice gives one or two "new" vertices that need to fit into the partition. If only one of the endpoints is new, the new vertex must go into the same subset as the other endpoint. To justify the denominator that converts ordered to unordered partition counting, any of at most $m-1$ places in the subset must be possible for the new vertex. So at most $k(m-1)$ choices are made to get one more vertex. If both vertices are new, there are m choices for which subset of the partition they go into, and at most $m^2 - m$ choices of position for the two vertices within the subset. So at most $km(m^2 - m)$ choices are made to get two more vertices.

10. Once r vertices have been chosen and partitioned to satisfy fences, the partition is completed by choosing the remaining $m^2 - 2 - r$ vertices.

11. The numerator of the formula from the statement is an overestimate of the product of the number of choices possible while choosing r vertices to satisfy fences, times $(m^2-2-r)!$ choices made out of the urn. When two vertices are chosen at once from a fence, the factors are "(km)" and "(m^2-m)". When only one vertex is chosen, the factor is "$k(m-1)$". The term "$k(m-1)$" is less than "km" and less than "(m^2-m)", so to be safe, assume all vertices are chosen in pairs, yielding $\frac{r}{2}$ factors "km" and $\frac{r}{2}$ factors "(m^2-m)".

□

Proof of Theorem 5.1. The size of real communication game is at least the size of the domain of μ divided by the maximum size of preimage of the elements in the image of μ, hence from Lemmas 5.2 and 5.4 we conclude that the size is at least

$$2\frac{|U_0|(m!)^{m-1}(m-2)!(m-1)!}{(km)^{r/2}(m^2-m)^{r/2}(m^2-2-r)!} \geq \frac{|G_0|(m!)^{m-1}(m-2)!(m-1)!}{(km)^{r/2}(m^2-m)^{r/2}(m^2-2-r)!} \geq \frac{1.8^{\sqrt{m/8}}}{4}.$$

The last inequality follows from [HC99], Section 4.6. □

Corollary 5.1. Let $Q(z,q)$ be a boolean CNF formula that $x \in G_0$ iff the formula $\exists q\ Q(x,q)$ is true. Let $R(z,r)$ be a boolean CNF formula that satisfy the following properties:

- there is at most one variable z in each clause;
- all variables z occur with negative signs;
- $y \in B_0$ iff the formula $\exists r\ R(y,r)$ is true.

Let L be a set of substitution to variable z,q and L' be a set of substitution to variable r. The size of any real communication game for the pair L, L' and the relation $\mathtt{Search}_{Q(z,y)\wedge R(z,r)}$ is at least $\frac{1.8^{\sqrt{m/8}}}{4}$.

Proof. Follows from Theorem 5.1 and Lemma 4.1. □

6 Random Cutting Planes

Definition 6.1. A δ-random CP proof distribution of formula ϕ is a random distribution (π_s, Δ_s) such that Δ_s is a CNF formula, π_s is a CP proof of $\phi \wedge \Delta_s$, and every fixed truth assignments of all variables satisfies the formula Δ_s with probability at least $1-\delta$.

The size of distribution is the maximum size of π_s.

Theorem 6.1. Let (π_s, Δ_s) be a δ-random CP proof distribution of the formula $\mathtt{BMS}$ for a convenient parameter m. Let d be the maximum number of clauses in formulas Δ_s. If $d\sqrt{\delta} \leq \frac{1}{2}$ then the size of this distribution is at least $(1-d\sqrt{\delta})\frac{1.8^{\sqrt{m/8}}}{4}$.

For $(g,h) \in G_0 \times B_0$ define $w(g,h) = (g, q^g, r^h)$ such that $Part(g,q^g) = 1$ and $Part(\neg h, r^h) = 1$. Let us assume that w is an injective map (since G_0 and B_0 are extremal instances we can choose w in such a way).

Let (π_s, Δ_s) be an arbitrary δ-random CP proof. Denote the size of π_s by k. For a sample s define a set $Bad_s \subseteq G_0 \times B_0$ to be the set of all pairs (g,h) such that $w(g,h)$ falsifies Δ_s.

Lemma 6.1 ([Kra16], Lemma 2.1)**.** There exists a sample s such that $|Bad_s| \leq \delta |G_0 \times B_0|$.

Let us fix s from this Lemma. Let d be the number of clauses in Δ_s.

Lemma 6.2 ([Kra16], Lemma 2.2)**.** There exist subsets $U \subseteq G_0$ and $V \subseteq B_0$ such that

- $U \times V \cap Bad_s = \emptyset$;
- $|U| \geq (1 - d\sqrt{\delta})|G_0|$;
- $|V| \geq (1 - d\sqrt{\delta})|B_0|$.

Lemma 6.3. Consider a pair (U, V) from Lemma 6. There is a real communication game for (U, V) and relation $\texttt{MonBit}_{U,V}$ of size that equals the size of π_s.

Proof. For each $x \in U$ one can fix some q^x such that $Q(x, q^x) = 1$, and for each $y \in V$ fix some r^y such that $R(y, r^y) = 1$.

Let us consider a proof of size S of the formula $Q(z, q) \wedge R(z, r)$. By Lemma 4.2 we can create a real communication game (H, A, B) of size S for sets of substitution (x, q^x) and (r^y). Note that function A depends only on a vertex of graph H and variables x, and function B depends only on a vertex of graph H and variables y, thus we consider this game as a game for the sets U, V. If we remove all vertices that are not valid for any pair $(x, y) \in U \times V$ then the game will remain correct. The leaves of this game are marked by clauses that are falsified by substitution (x, q^x, r^y) for some (x, y), this clause cannot be from formula Δ_s by the choice of (U, V), and this clause cannot be from formula Q by the choice of q^x, hence it is a clause from the formula R that is not satisfied by r^y, thus this clause contains a variable z_i and $y_i = 0$, but $x_i = 1$ and we can mark this leaf by i. □

Proof of Theorem 6.1. Consider s from Lemma 6.1. By Lemma 6.3 and Lemma we have a real communication game for sets (U, V) and relation $\texttt{MonBit}_{U,V}$ of size that equals the size of π_s where $U \subseteq G_0$, $V \subseteq B_0$ and $|U| \geq \frac{|G_0|}{2}$, $|V| \geq \frac{|B_0|}{2}$. Hence the statement of the Theorem follows from Theorem 5.1. □

7 Monotone CSP-SAT

In this section we consider a monotone function called CSP-SAT. This function was defined in [Oli15, GP14]; in [RP16] the authors gave a fully exponential lower bound on the size of monotone boolean formulas for this function. We prove that this function requires an exponential monotone real circuit size.

Definition 7.1 ([GP14])**.** The function CSP-SAT is defined relative to some finite alphabet Σ and a fixed constraint topology given by a bipartite graph G with left vertices V (variable nodes) and right vertices U (constraint nodes). We think of each $v \in V$ as a variable taking on values from Σ, an edge $(v, u) \in E(G)$ indicates that variable v is involved in constraint node u. Let d be the maximum degree of a node in U. We define $\texttt{SAT} = \texttt{SAT}_{G,\Sigma} : \{0,1\}^N \to \{0,1\}$ on $N \leq |U| \cdot |\Sigma|^d$ bits as follows. An input $\alpha \in \{0,1\}^N$ describes a CSP instance by specifying, for each constraint node $u \in U$, its truth table: a list of at most $|\Sigma|^d$ bits that record which assignments to the variables involved in

u satisfy u. Then $\texttt{SAT}(\alpha) := 1$ iff the $\texttt{CSP}$ instance described by α is satisfiable. This encoding of $\texttt{CSP}$ satisfiability is indeed monotone: if we flip any 0 in a truth table of a constraint into a 1, we are only making the constraint easier to satisfy.

The proof of the following theorem use a simplification of analogy of reduction from [GP14, RM99].

Theorem 7.1. Let Φ be an unsatisfiable d-CNF formula on n variables and m clauses with the variables splitted into sets X, Y. Let G be a constraint topology of Φ. If there is a real (boolean) communication game of size S for sets $\texttt{SAT}^{-1}_{G,\{0,1\}}(1), \texttt{SAT}^{-1}_{G,\{0,1\}}(0)$ and $\texttt{MonBit}_{\texttt{SAT}_{G,\{0,1\}}}$ relation then there is a real (boolean) communication game of size S for sets $\{0,1\}^{|X|}, \{0,1\}^{|Y|}$ (sets of all possible substitution to variables X and Y) and $\texttt{Search}_\phi$ relation.

Proof. We want to create a mapping an instance of $\texttt{Search}_\Phi$ to instance of $\texttt{MonBit}_{\texttt{SAT}_{G,\{0,1\}}}$. Let (x, y) be an instance of $\texttt{Search}_\Phi$. Consider a set $P = x \times \{0,1\}^{|Y|}$, which is a set of substitutions to formula Φ, and define a "positive" instance of $\texttt{SAT}_{G,\{0,1\}}$ as follows: a constraint that corresponds to a vertex $u \in U$ is satisfied only by substitutions from set P. To create a "negative" instance ℓ we consider a vertex $u \in U$ and clause C_u from Φ that corresponds to u. We say that a constraint of our instance is satisfied by a substitution ρ iff $\rho = (a, y|_{vars(C_u)})$, where a is a restriction of ρ to variables from X and $y|_{vars(C_u)}$ is a restriction of y to variables of clause C_u, and C_u is satisfied by ρ. The satisfiability of $\texttt{SAT}_{G,\{0,1\}}(\ell)$ is equivalent to satisfiability of formula Φ, hence it is a "negative" instance.

If we know an answer for $\texttt{MonBit}_{\texttt{SAT}_{G,\{0,1\}}}$ on our instance then we know a constraint and an element from P that satisfies this constraint and we know that this constraint is not satisfied by any extension of y, therefore a clause that corresponds to this constraint is not satisfied by (x, y). Hence we have a real (boolean) communication game for $\texttt{MonBit}_{\texttt{SAT}_{G,\{0,1\}}}$, so we can use it for $\texttt{Search}_\phi$ by using the above reduction. □

Corollary 7.1. Let G be a constraint topology of $\texttt{BMS}'$. The size of any monotone real circuit that computes $\texttt{SAT}_{G,\{0,1\}} : \{0,1\}^N \to \{0,1\}$ is at least $2^{\Omega(N^{1/8})}$.

Proof. Follows from Theorem 7.1 and Corollary 5.1. □

Acknowledgements

This research is supported by Russian Science Foundation (project 16-11-10123).

The author is grateful to Pavel Pudlák and Dmitry Itsykson for fruitful discussions. The author also thanks Edward Hirsch, Dmitry Itsykson and anonymous reviewers for error correction.

References

[AB87] Noga Alon and Ravi B. Boppana. The monotone circuit complexity of boolean functions. *Combinatorica*, 7(1):1–22, 1987.

[BKT14] Samuel R. Buss, Leszek Aleksander Kolodziejczyk, and Neil Thapen. Fragments of approximate counting. *J. Symb. Log.*, 79(2):496–525, 2014.

[BPS07] Paul Beame, Toniann Pitassi, and Nathan Segerlind. Lower bounds for Lovász–Schrijver systems and beyond follow from multiparty communication complexity. *SIAM J. Comput.*, 37(3):845–869, 2007.

[DM16] Irit Dinur and Or Meir. Toward the KRW composition conjecture: Cubic formula lower bounds via communication complexity. In *31st Conference on Computational Complexity, CCC 2016, May 29 to June 1, 2016, Tokyo, Japan*, pages 3:1–3:51, 2016.

[GP14] Mika Göös and Toniann Pitassi. Communication lower bounds via critical block sensitivity. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 847–856, 2014.

[HC99] Armin Haken and Stephen A. Cook. An exponential lower bound for the size of monotone real circuits. *Journal of Computer and System Sciences*, 58(2):326–335, 1999.

[HN12] Trinh Huynh and Jakob Nordström. On the virtue of succinct proofs: amplifying communication complexity hardness to time-space trade-offs in proof complexity. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 233–248, 2012.

[Hru13] Pavel Hrubeš. A note on semantic cutting planes. *Electronic Colloquium on Computational Complexity (ECCC)*, 20:128, 2013.

[Kra97] Jan Krajíček. Interpolation theorems, lower bounds for proof systems, and independence results for bounded arithmetic. *J. Symb. Log.*, 62(2):457–486, 1997.

[Kra98] Jan Krajíček. Interpolation by a game. *Math. Log. Q.*, 44:450–458, 1998.

[Kra08] Jan Krajíček. An exponential lower bound for a constraint propagation proof system based on ordered binary decision diagrams. *J. Symb. Log.*, 73(1):227–237, 2008.

[Kra16] Jan Krajíček. A feasible interpolation for random resolution. *CoRR*, abs/1604.06560, 2016.

[KW90] Mauricio Karchmer and Avi Wigderson. Monotone circuits for connectivity require super-logarithmic depth. *SIAM J. Discrete Math.*, 3(2):255–265, 1990.

[Oli15] Igor Oliveira. Unconditional lower bounds in complexity theory. *PhD thesis, Columbia university*, 2015.

[Pud97] Pavel Pudlák. Lower bounds for resolution and cutting plane proofs and monotone computations. *J. Symb. Log.*, 62(3):981–998, 1997.

[Pud10] Pavel Pudlák. On extracting computations from propositional proofs (a survey). In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2010, December 15-18, 2010, Chennai, India*, pages 30–41, 2010.

[Raz95] A. A. Razborov. Unprovability of lower bounds on circuit size in certain fragments of bounded arithmetic. *Izvestiya RAN. Ser. Mat.*, pages 201–224, 1995.

[RM99] Ran Raz and Pierre McKenzie. Separation of the monotone NC hierarchy. *Combinatorica*, 19(3):403–435, 1999.

[RP16] Robere Robert and Tonian Pitassi. Strongly exponential lower bounds for monotone computation. *ECCC Report: TR16-188*, 2016.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il