Electronic Colloquium on Computational Complexity, Report No. 72 (2014)

Depth Lower Bounds against Circuits with Sparse Orientation

Sajin Koroth* Jayalal Sarma M.N.*

April 29, 2014

Abstract

We study depth lower bounds against non-monotone circuits, parametrized by a new measure of non-monotonicity: the orientation[1] of a function f is the characteristic vector of the minimum sized set of negated variables needed in any DeMorgan[2] circuit computing f. We prove trade-off results between the depth and the weight/structure of the orientation vectors in any circuit C computing the **CLIQUE** function on an n vertex graph. We prove that if C is of depth d and each gate computes a Boolean function with orientation of weight at most w (in terms of the inputs to C), then $d \times w$ must be $\Omega(n)$. In particular, if the weights are $o(\frac{n}{\log^k n})$, then C must be of depth $\omega(\log^k n)$. We prove a barrier for our general technique. However, using specific properties of the **CLIQUE** function (used in [5]) and the Karchmer-Wigderson framework [12], we go beyond the limitations and obtain lower bounds when the weight restrictions are less stringent. We then study the depth lower bounds when the structure of the orientation vector is restricted. Asymptotic improvements to our results (in the restricted setting), separates NP from NC. As our main tool, we generalize Karchmer-Wigderson game [12] for monotone functions to work for non-monotone circuits parametrized by the weight/structure of the orientation. We also prove structural results about orientation and prove connections between number of negations and weight of orientations required to compute a function.

1 Introduction

Deriving size/depth lower bounds for Boolean circuits computing NP-complete problems has been one of the main goals in circuit complexity. Attempts to prove size lower bounds against constant depth circuits has yielded useful results (see survey [1, 2] and textbook [11]).

*Department of CSE, Indian Institute of Technology Madras, Chennai 600036, India.

[1]A generalization of monotone functions are studied under the name *unate functions*(cf. [8]). We inherit the terminology of *orientation* from that setting. We remark that our definition is universal unlike the case of unate functions.

[2]Circuits where negations appear only at the leaves.

ISSN 1433-8092

However, despite many efforts, for computing explicit functions, the best size lower bound known against general circuits is still a constant factor on the number of inputs [9], and the best depth lower bound known against general bounded fan-in circuits is (derived from formula size lower bound due to Håstad [18]) which is less than $3 \log n$.

Notable progress has been made in proving lower bounds against monotone circuits. Razborov [16] proved a super-polynomial size lower bound against monotone circuits computing the **CLIQUE** function which is NP-hard. This was further strengthened to exponential lower bounds by Alon and Boppana [3]. A super polynomial monotone size lower bound is also known [17] for the **PMATCH** problem. The latter result also showed that non-monotonicity helps in size restricted settings as **PMATCH** is known to be in P[6].

Moving in the direction of non-monotonicity, Amano and Maruoka [5] established super-polynomial lower bounds against circuits with at most $\frac{1}{6} \log \log n$ negations computing the **CLIQUE** function. A chasm was already known at the $\log n$ negations; Fisher [7] proved that any circuit of polynomial size can be converted to a circuit of polynomial size that has only $\log n$ negations. In particular, this implies that if we are able to extend the technique of lower bounds to work against circuits having $\log n$ negations, then it separates P from NP. The gap was further tightened by Jukna [10] (for multi-output functions), where he showed a super-polynomial size lower bound against circuits with $\log n - 16 \log \log n$ negations.

In terms of depth lower bounds, it is known that **CLIQUE** function and the **PMATCH** function on graphs of n vertices require $\Omega(n)$ depth for any bounded fan-in monotone circuit computing them [15]. Thus, non-monotonicity is useful in the depth restricted setting also, as **PMATCH** is known to be in non-uniform NC^2 [13]. One main technique involved in the monotone depth lower bound for **PMATCH** [15] is a characterization of circuit depth using a communication game defined between two players. Raz and Wigderson [14] used this framework to obtain a lower bound of $\Omega(n^2)$ on the number of negations at the leaves for any $O(\log n)$ depth DeMorgan circuit solving the s-t connectivity problem. However, we do not know[3] depth lower bounds against circuits where there are negations at arbitrary locations using the Karchmer-Wigderson framework.

1.1 Our Results

We study an alternative way of limiting the non-monotonicity in the circuit. To arrive at our restriction, we define a new measure called *orientation* of a Boolean function.

Definition 1. *A function $f : \{0,1\}^n \to \{0,1\}$ is said to have* orientation $\beta \in \{0,1\}^n$ *if there is a monotone function $h : \{0,1\}^{2n} \to \{0,1\}$ such that : $\forall x \in \{0,1\}^n, f(x) = h(x, (x \oplus \beta))$.*

The orientation of a Boolean function is simply the indicator vector of the set of inputs which are required to be negated in any DeMorgan circuit computing the function f. Indeed, if f itself is monotone, the orientation is simply the all-0s vector. The *weight of the*

[3] Indeed, size lower bounds against bounded fan-in circuits in the presence of negations [5] also imply depth lower bounds against them. In particular, [5] implies that any circuit with $\frac{1}{6} \log \log n$ negation gates computing $\mathbf{CLIQUE}(n, (\log n)^{\sqrt{\log n}})$ requires depth $\Omega((\log n)^{\sqrt{\log n}})$.

orientation is simply the number of 1s in β, and can be thought of as a parameter indicating how "close" f is to a monotone function.

The same definitions can be extended to circuits as well. We consider circuits where the function computed at each gate can be non-monotone, but the corresponding orientation (with respected to the inputs to the circuit) must be of limited weight. We say a circuit C is weight w oriented if every internal gate of C computes a function which has an orientation β with $|\beta| \leq w$. The semantic restriction we study limits the weight of the orientation of the function computed at each gate of the circuit (in terms of the original inputs of the circuit). We prove the following theorem which presents a depth vs weight trade-off.

Theorem 1. *If C is a Boolean circuit of depth d and weight of the orientation w ($w > 0$), computing* **CLIQUE** *then, $d \times w$ must be $\Omega(n)$.*

In particular, if the weights are $o(\frac{n}{\log^k n})$, the **CLIQUE** function requires $\omega(\log^k n)$ depth. By contrast, any circuit computing **CLIQUE** has weight of the orientation at each gate at most n^2. We prove the above theorem by extending the Karchmer-Wigderson framework to the case of non-monotone sparsely oriented circuits. The proof critically requires the route via Karchmer-Wigderson games since it is unclear how to directly simulate the above non-monotone circuit model using a monotone circuit. We remark that the above theorem applies even to circuits computing **PMATCH**.

The difficulty in extending the above lower bound to more general lower bounds is the potential presence of gates computing densely oriented functions. In this context, we explore the usefulness of having gates with non-zero orientation in the circuit. We argue that allowing even a constant number of non-zero (but dense) oriented gates makes the circuit more powerful in the limited depth setting. In particular, we show (see Theorem 11) that:

Theorem 2. *There exists a monotone function f which cannot be computed by poly-log depth monotone circuits, but there is a poly-log depth circuit computing it such that there are at most two internal gates which has a non-zero orientation β.*

We note that the function in Theorem 11 is derived as a restriction from the non-uniform NC^2 circuit computing **PMATCH** and hence is *not explicit*. The above theorem indicates that the densely oriented gates are indeed useful, and that Theorem 1 cannot be improved in terms of the number of densely oriented gates it can handle, without using specific properties about the function(for example, **CLIQUE**) being computed.

Going beyond the above limitations, we exploit the known properties of the **CLIQUE** function and the generalized Karchmer-Wigderson games to prove lower bounds against less stringent weight restrictions (in particular, we can restrict the weight restrictions to only negation gates and their inputs).

Theorem 3. *For any circuit family $\mathcal{C} = \{C_m\}$ (where $m = \binom{n}{2}$) computing* $\mathbf{CLIQUE}(n, n^{\frac{1}{6\alpha}})$ *where there are $\ell + k$ negation gates, with $\ell \leq 1/6 \log\log n, \alpha = 2^{\ell+1} - 1$ and the k negation gates in C_m are computing functions which are sensitive only on w inputs (i.e., the orientation of their input as well as their output is at most w) and the remaining ℓ negations compute functions of arbitrary orientation:* $\mathbf{Depth}(C_m) \geq n^{\frac{1}{2^{2^\ell+8}}} - kw - \ell$

This theorem implies that **CLIQUE** cannot be computed by circuits with depth $n^{o(1)}$ even if we allow some constant number of gates to have non-zero (even dense) orientation - thus going beyond the earlier hurdle presented for **PMATCH**. We remark that the above theorem also generalizes the case of circuits with negations at the leaves ($\ell = 0$, and $w = 1$).

As far as we know, the above theorem is the first instance of a lower bound which combines the approximation method with the Karchmer-Wigderson games. It also gives hope that by using properties of **CLIQUE** (like hardness of approximation [4] used by [5]) we can possibly push the technique further.

We also explore the question of the number of densely oriented gates that are required in an optimal depth circuit. We establish the following connection to the number of negations in the circuit.

Theorem 4. *For any circuit C with t negations, there is a circuit C' computing the same function such that* $\mathbf{Size}(C') \leq 2^t \times (\mathbf{Size}(C) + 2^t) + 2^t$, *and there are at most* $2^{t-1}(t+2) - 1$ *internal gates whose orientation is a non-zero vector.*

We now turn to circuits where the structure of the orientation is restricted. The restriction is on the number of vertices of the input graph involved in edges indexed by β.

Theorem 5. *If C is a circuit computing the* **CLIQUE** *function and for each gate g of C, the number of vertices of the input graph involved in edges indexed by β_g (the orientation vector of gate g) is at most w, then $d \times w$ must be $\Omega(\frac{n}{\log n})$.*

We also study a sub-class of the above circuits for which we prove lower bound results very close to the required ones. A circuit is said to be of *uniform orientation* if there exists a $\beta \in \{0,1\}^n$ such that every gate in it computes a function which has orientation β.

Theorem 6. *Let C be a circuit computing the* **CLIQUE** *function, with uniform orientation $\beta \in \{0,1\}^n$ such that there is a subset of vertices U, $|U| \geq \log^{k+\epsilon} n$ for which $\beta_e = 0$ for all edges e within U, then C must have depth $\omega(\log^k n)$.*

We remark that a DeMorgan circuit has an orientation of weight exactly equal to the number of negated variables. However, this result is incomparable with that of [14] against DeMorgan circuits for two reasons : (1) this is for the **CLIQUE** function. (2) the lower bounds and the class of circuits are different.

In contrast to the above theorem, we show that an arbitrary circuit can be transformed into one having our structural restriction on the orientation with $|U| = O(\log^k n)$.

Theorem 7. *If there is a circuit C computing* **CLIQUE** *with depth d then for any set of $c \log n$ vertices U, there is an equivalent circuit C' of depth $d + c\log n$ with orientation β such that none of the edges $e(u,v)$, $u, v \in U$ has $\beta_{e(u,v)} = 1$.*

Thus if either Theorem 6 is extended to $|U| = \Omega(\log^k n)$ or the transformation in Theorem 7 can be modified to give $|U| = O(\log^{k+\epsilon} n)$ for some constant $\epsilon > 0$, then a depth lower bound for **CLIQUE** function against general circuits of depth $O(\log^k n)$ will be implied.

2 Preliminaries

For $x, y \in \{0,1\}^n$, $x \leq y$ if and only if for all $i \in [n]$, $x_i \leq y_i$. A Boolean function f is said to be monotone if for all $x \leq y$, $f(x) \leq f(y)$. In other words value of a monotone function does not decrease when input bits are changed from 0 to 1.

For a set U, we denote by $\binom{U}{2}$ the set $\{\{u,v\} \,|\, u, v \in U\}$. In an undirected graph $G = (V, E)$, a clique is a set $S \subseteq V$ such that $\binom{S}{2} \subseteq E(G)$. $\mathbf{CLIQUE}(n,k)$ is a Boolean function $f : \{0,1\}^{\binom{n}{2}} \to \{0,1\}$ such that for any $x \in \{0,1\}^{\binom{n}{2}}$, $f(x) = 1$ if G_x, the undirected graph represented by the undirected adjacency matrix x has a clique of size k. $\mathbf{CLIQUE}(n,k)$ is a monotone function as adding edges (equivalent to turning 0 to 1 in adjacency matrix) cannot remove a k-clique, if one already exists. By $\mathbf{CLIQUE}$, we denote $\mathbf{CLIQUE}(n, \frac{n}{2})$. A perfect matching of an undirected graph $G = (V, E)$ is a $M \subseteq E(G)$ such that no two edges in M share an end vertex and it is such that every vertex $v \in V$ is contained as an end vertex of some edge in M. Corresponding Boolean function $\mathbf{PMATCH} : \{0,1\}^{\binom{n}{2}} \to \{0,1\}$ is defined as $\mathbf{PMATCH}(x) = 1$ if G_x contains a perfect matching. It is easy to note that $\mathbf{PMATCH}$ is also a monotone function.

A circuit is a directed acyclic graph whose internal nodes are labeled with $\wedge$, $\vee$ and $\neg$ gates, and leaf nodes are labeled with inputs. The function computed by the circuit is the function computed by a designated “root” node. All our circuits are of bounded fan-in. The depth of a circuit C, denoted by $\mathbf{Depth}(C)$ is the length of the longest path from root to any leaf, and $\mathbf{Depth}(f)$ denotes the minimum possible depth of a circuit computing f. By $\mathbf{Depth}_t(f)$ we denote the minimum possible depth of a circuit computing f with at most t negations. Size of a circuit is simply the number of internal gates in the circuit, and is denoted by $\mathbf{Size}(C)$. $\mathbf{Size}(f), \mathbf{Size}_t(f)$ are defined analogous to $\mathbf{Depth}(f), \mathbf{Depth}_t(f)$ respectively. We refer the reader to a standard textbook (cf. [19]) for more details.

We now review the Karchmer-Wigderson games and the related lower bound framework. The technique is a strong connection between circuit depth and communication complexity of a specific two player game where the players say Alice and Bob are given inputs $x \in f^{-1}(1)$ and $y \in f^{-1}(0)$, respectively. In the case of general circuits, the game is denoted by $\mathbf{KW}(f)$ and the goal is to find an index i such that $x_i \neq y_i$. In the case of monotone circuits, the game is denoted by $\mathbf{KW}^+(f)$ and the goal is to find an index i such that $x_i = 1$ and $y_i = 0$. We abuse the notation and use $\mathbf{KW}(f)$ and $\mathbf{KW}^+(f)$ to denote the number of bits exchanged in the worst case for the best protocols for the corresponding communication games. Karchmer and Wigderson [12] proved that for any function f depth of the best circuit computing f, denoted by $\mathbf{Depth}(f)$ is equal to $\mathbf{KW}(f)$. For any monotone function f the depth of the best monotone circuit computing f, denoted by $\mathbf{Depth}^+(f)$ is equal $\mathbf{KW}^+(f)$. Raz and Wigderson [15] showed that $\mathbf{KW}^+(\mathbf{CLIQUE})$ and $\mathbf{KW}^+(\mathbf{PMATCH})$ are both $\Omega(n)$.

2.1 Characterization of Orientation

We recall the definition of orientation, A function $f : \{0,1\}^n \to \{0,1\}$ is said to have *orientation* $\beta \in \{0,1\}^n$ if there is a monotone function $h : \{0,1\}^{2n} \to \{0,1\}$ such that :

$\forall x \in \{0,1\}^n, f(x) = h(x, (x \oplus \beta))$. Thus, if $\beta \in \{0,1\}^n$ is an orientation for a function f, then any $\beta' \geq \beta$ is also an orientation for f by definition of orientation.

We first show that any function $f(x)$ can be written in the form of definition as an $h(x, x \oplus \beta)$ for a monotone function h. Let C be any circuit computing f. Convert C into a DeMorgan circuit C' by pushing down the negations by repeated application of De-Morgan's laws. In C' replace every $\bar{x}_i$ with a new variable y_i for every $i \in [n]$. Thus C' on inputs x, y is a monotone function. Since there are n input variables at most n y_i's are needed. Let $h = C'(x_1, \ldots, x_n, y_1, \ldots, y_n)$ be the monotone function computed by C' after replacing the negated inputs by fresh variables. Clearly h satisfies the required form with β defined as $\beta_i = 1$ if and only if $\bar{x}_i$ appears in C'. Now if the function f has such a form, take any monotone circuit C_h computing h. Replace all the inputs $x_i \oplus \beta_i$ where $\beta_i = 1$ with $\bar{x}_i$ and all the inputs $x_i \oplus \beta_i$ where $\beta_i = 0$ with x_i in C_h. Thus we get a circuit C'' computing f, which is De-Morgan and has negations only on variables where $\beta_i = 1$. Thus for any function f whose orientation is β, there is a circuit C of uniform orientation β. Because any sub-circuit rooted at a gate of C'' is once again De-Morgan and has only negated variables which are a subset of negated variables required for computing f.

We now establish that orientation is a well-defined measure. First we prove a sufficient condition for the β_i to be 1 in the orientation for a function f.

Proposition 8. *For any function f, if there exists a pair (u, v) such that $u_i = 0, v_i = 1$, $u_{[n]\setminus\{i\}} = v_{[n]\setminus\{i\}}$ and $f(u) = 1, f(v) = 0$ then any orientation β of the function must have $\beta_i = 1$.*

Proof. Let h be the monotone function corresponding to f for β such that $\forall x, f(x) = h(x, x \oplus \beta)$. Assume to the contrary that $\beta_i = 0$. Since $u_{[n]\setminus\{i\}} = v_{[n]\setminus\{i\}}$, we have that $u_{[n]\setminus\{i\}} \oplus \beta = v_{[n]\setminus\{i\}} \oplus \beta$ for any β. Hence $(u, u \oplus \beta), (v, v \oplus \beta)$ differs only in two indices, namely $i, n+i$. At i, $u_i = 0, v_i = 1$, and at $n+i$ since $\beta_i = 0$, $u_{n+i} = 0, v_{n+i} = 1$. Hence we get that $(u, u \oplus \beta) < (v, v \oplus \beta)$, but $h(u, u \oplus \beta) = 1, h(v, v \oplus \beta) = 0$ a contradiction to monotonicity of h. □

It is not a priori clear that the minimal (with respect to $<$ relation on the Boolean hypercube) orientation for a function f is unique. We prove that it is indeed the case.

Proposition 9. *Minimal orientation for a function $f : \{0,1\}^n \to \{0,1\}$ is well defined and it is $\beta \in \{0,1\}^n$ such that $\beta_i = 1$ if and only if there exists a pair (u, v) such that $u_i = 0, v_i = 1$, $u_{[n]\setminus\{i\}} = v_{[n]\setminus\{i\}}$ and $f(u) = 1, f(v) = 0$.*

Proof. From Proposition 8 it is clear that any orientation β' of a function f is such that $\beta \leq \beta'$. We claim that negations of variables in β suffices to compute f using a DeMorgan circuit. Define a partial function $h : \{0,1\}^{2n} \to \{0,1\}$ associated with orientation β of f as $h(x, x \oplus \beta) \triangleq f(x)$. We claim that this partial function has an extension which is a monotone function. We claim that for any $u, v \in \{0,1\}^n$ such that $u \leq v$ and $f(u) = 1, f(v) = 0$, there exists an $i \in [n]$ such that $u_i = 0, v_i = 1$ and $\beta_i = 1$. Let $w_0 = u \leq w_1 \leq \cdots \leq w_j \leq w_{j+1} \leq \cdots \leq w_k = v$ be a chain between u and v. Take the minimum j such that $f(w_j) = 1$ and $f(w_{j+1}) = 0$. Since w_j, w_{j+1} satisfies assumptions of Proposition 8, for the

i where w_j and w_{j+1} differs, $\beta_i = 1$. Since $u \leq w_j$ and ith bit of w_j is 0, we get $u_i = 0$. Similarly $v_i = 1$ as $v \geq w_{j+1}$ and jth bit of w_{j+1} is 1. With this claim we can prove that for any $(s, s \oplus \beta)$ and $(t, t \oplus \beta)$ either they are incomparable or $f(s) \geq f(t)$ if and only if $(s, s \oplus \beta) \geq (t, t \oplus \beta)$. Assume to the contrary that $f(s) < f(t)$ and $(s, s \oplus \beta) \geq (t, t \oplus \beta)$. Since $(s, s \oplus \beta) \geq (t, t \oplus \beta)$, $s \geq t$ and $f(s) = 0, f(t) = 1$ as $f(s) < f(t)$. But then we are guaranteed by the earlier claim an $i \in [n]$ such that $s_i = 1, t_i = 0, \beta_i = 1$. Since $\beta_i = 1$, $s_i \oplus \beta_i = 0$ and $t_i \oplus \beta_i = 1$ whereas $s_i = 1, t_i = 0$ implying that $(s, s \oplus \beta) \ngeq (t, t \oplus \beta)$, a contradiction. Thus the partial function we defined will never have a chain with a 1 to 0 transition. It is easy to verify for any partial function h which does not have a $1 \to 0$ transition on any of its chains, a monotone function extending it can be obtained. $\square$

3 Lower Bound Argument for Sparsely Oriented Circuits

In this section, we prove Theorem 1 which shows the trade-off between depth and weight of orientation of the internal gates of a circuit. We prove the following main lemma of our paper.

Lemma 1. *If C is a circuit of depth d such that each internal gate computes a Boolean function whose orientation has weight at most w and C is computing a monotone function $f : \{0,1\}^n \to \{0,1\}$ which is sensitive on all its inputs, then $d \times (4w + 1) \geq \mathbf{KW}^+(f)$.*

Proof. The proof idea is to devise a protocol for $\mathbf{KW}^+(f)$ using C having $\mathbf{Depth}(C)$ rounds and each round having a communication cost of $4w + 1$.

Alice is given $x \in f^{-1}(1)$ and Bob is given $y \in f^{-1}(0)$. The goal is to find an index i such that $x_i = 1, y_i = 0$. The protocol is described in Algorithm 1.

We now prove that the protocol (Algorithm 1) solves $\mathbf{KW}^+(f)$. The following invariant which is maintained during the run of the protocol is crucial for the proof.

Invariant: When the protocol is at a node which computes a function f with orientation vector β it is guaranteed a priori that the inputs held by Alice and Bob, x' and y' are equal on the indices where $\beta_i = 1$, $f(x') = 1, f(y') = 0$ and restriction of f obtained by fixing variables where $\beta_i = 1$ to $x'_i (= y'_i)$ is a monotone function.

Assuming that the invariant is maintained, we claim that when the protocol stops at an input node of the circuit computing a function f with $f(x') = 1$ and $f(y') = 0$ then $f = x_i$ for some $i \in [n]$. If the input node is a negative literal, say $\bar{x}_i$ then by Proposition 8, orientation of $\bar{x}_i$ has $\beta_i = 1$. By the guarantee that $x'_\beta = y'_\beta$, $x'_i = y'_i$, contradicting $f(x') \neq f(y')$. Hence whenever the protocol stops at leaf node it is guaranteed that the leaf is labeled by a positive literal. And when input node is labeled by a positive literal x_i, then a valid solution is output as $f(x') = 1, f(y') = 0$ implies $x'_i = 1$ and $y'_i = 0$. Note that during the run of the protocol we only changed x, y at some indices $i, x_i \neq y_i$ to $x'_i = y'_i$. Hence, any index where $x'_i \neq y'_i$ it is the case that $x_i = x'_i$ and $y_i = y'_i$.

Algorithm 1 Modified Karchmer-Wigderson Protocol

1: {Let x' and y' be the current inputs. At the current gate g computing f, with the input gates g_1 and g_2, f_1 and f_2 be the corresponding sub-functions and β_1, β_2 be the corresponding orientations (and are known to both Alice and Bob). If g_1 or g_2 is a negation gate, let γ_1 and γ_2 be the orientation vectors of input functions to g_1 and g_2, otherwise they are 0-vectors. Let $\alpha = \beta_1 \vee \beta_2 \vee \gamma_1 \vee \gamma_2$. Let $S = \{i : \alpha_i = 1\}$, x_S is the substring of x indexed by S. }
2: **if** g is $\wedge$ **then**
3: Alice sends x'_S to Bob. Bob compares x'_S with y'_S.
4: **if** there is an index $i \in S$ such that $x'_i = 1$ and $y'_i = 0$ **then**
5: Output i.
6: **else**
7: Define $y'' \in \{0,1\}^n$: $y''_S = x'_S$ and $y''_{[n]\setminus S} = y'_{[n]\setminus S}$.
8: Bob sends $i \in \{1,2\}$ such that $f_i(y'') = 0$ to Alice. They recursively run the protocol on g_i with $x' = x'$ and $y' = y''$.
9: **end if**
10: **end if**
11: **if** g is $\vee$ **then**
12: Bob sends y'_S to Alice. Alice compares y'_S with x'_S.
13: **if** there is an index $i \in S$ such that $x'_i = 1$ and $y'_i = 0$ **then**
14: Output i.
15: **else**
16: Define $x'' \in \{0,1\}^n$: $x''_S = y'_S$ and $x''_{[n]\setminus S} = x'_{[n]\setminus S}$.
17: Alice sends $i \in \{1,2\}$ such that $f_i(x'') = 1$ to Bob. They recursively run the protocol on g_i with $x' = x''$ and $y' = y'$.
18: **end if**
19: **end if**

Now we prove the invariant. Note that it is vacuously true at the root gate as f is a monotone function implying $\beta = 0^n$, and in the standard $\mathbf{KW}^+(f)$ game $x \in f^{-1}(1)$ and $y \in f^{-1}(0)$. We argue that, while descending down to one of the children of the current node the invariant is maintained. To begin with, we show that the protocol does not get stuck in step 8 (and similarly for step 17). To prove this, we claim that at an $\wedge$ gate $f = f_1 \wedge f_2$, if the protocol failed to find an i in step 4 such that $x'_i = 1, y'_i = 0$ then on the modified input y'' at least one of $f_1(y'')$ or $f_2(y'')$ is guaranteed to be zero. Since the protocol failed to output an i such that $x'_i = 1, y'_i = 0$, it must be the case that $x'_i \leq y'_i$ for indices indexed by β_1, β_2. Let U be the subset of indices indexed by β_1 and β_2 where $x_i = 0$ and $y_i = 1$. Bob obtains y'' from y' by setting $y''_i = 0$ for all $i \in U$. Thus we have made sure that x' and y'' are the same on the variables whose negations are required to compute f, f_1 and f_2.

Consider the functions $f', f'' : \{0,1\}^{n-|\beta_1 \vee \beta_2|} \rightarrow \{0,1\}$ which are obtained by restricting the variables indexed by orientation vectors of f_1 and f_2 to the value of those variables in x'. Both f' and f'' are monotone as they are obtained by restricting all negated input variables of the DeMorgan circuits computing f_1 and f_2 for orientations β_1 and β_2 respectively. The changes made to x', y' were only at places where they differed. Thus at all the indices where x', y' were same, x', y'' is also same. Hence monotone restriction $f_{x'_\beta}$ of f obtained by setting variables indexed by β to their values in x' is a consistent restriction for y'' also. It is easy to note that $y'' \leq y'$. Hence $f(y'') = 0$ because y'' agrees with y' on variables indexed by β (as x'' agrees with y' and y'' on variables indexed by β) implying $f_{x'_\beta}(y''_{[n]\setminus\beta}) \leq f_{x'_\beta}(y'_{[n]\setminus\beta}) = 0$. Since $f(y'') = 0$, it is guaranteed that one of $f_1(y''), f_2(y'')$ is equal to 0. Bob sets $y' = y''$ and sends 0 if it is $f_1(y'') = 0$ or 1 otherwise, indicating Alice which node to descend to. Note that $x'_{\beta_1} = y''_{\beta_1}$, $x'_{\beta_2} = y''_{\beta_2}$ and restriction of f_1, f_2 to $x'_{\beta_1}, x'_{\beta_2}$ respectively gives monotone functions f', f'' thus maintaining the invariant for both f_1 and f_2.

We claim that if any of the input gates g_1, g_2 to the current $\wedge$ gate g is a $\neg$ gate then the protocol will not take the path through the negation gate. To argue this, we use the following lemma.

Lemma 2. *If $\ell, \bar{\ell}$ are functions with orientations β, γ, then for all $x, y \in \{0,1\}^n$ such that $x_{\beta\vee\gamma} = y_{\beta\vee\gamma}$, $\ell(x) = \ell(y)$.*

Proof. We know that for a function ℓ, if there exists a pair $(u, v) \in \{0,1\}^n \times \{0,1\}^n$ with $u \leq v$, $u_i \neq v_i$, $u_{[n]\setminus\{i\}} = v_{[n]\setminus\{i\}}$ and $\ell(u) = 1, \ell(v) = 0$ then by Proposition 8 for every orientation β, $\beta_i = 1$. Let i be an index on which ℓ is sensitive, i.e., there exists $(u, v) \in \{0,1\}^n \times \{0,1\}^n$ with $u \leq v$, $u_i \neq v_i$, $u_{[n]\setminus\{i\}} = v_{[n]\setminus\{i\}}$ and $\ell(u) \neq \ell(v)$. Note that l is sensitive on i need not force $\beta_i = 1$, as it could be that $\ell(u) = 0$ and $\ell(v) = 1$. But in this case $\bar{\ell}(u) = 1$ and $\bar{\ell}(v) = 0$, hence $\gamma_i = 1$ for $\bar{\ell}$. Hence, ℓ is sensitive only on indices in $\beta \vee \gamma$. □ □

The lemma establishes that at every negation gate in weight w oriented circuit, a function which is sensitive on at most $2w$ indices is computed. Hence, the root gate cannot be a negation gate for a function sensitive on all inputs if $2w < n$. Suppose only one child is a negation gate, say f_1. Since we ensure $x'_{\beta_1\vee\gamma_1} = y''_{\beta_1\vee\gamma_1}$, the above lemma implies $f_1(x') = f_1(y'')$. But the protocol does not descend down a path where x', y'' are not separated. Hence the claim.

This also proves that when the protocol reaches an $\wedge$ node with both children negated, at the round for that node protocol outputs an index i and stops. Otherwise, since we ensure $x'_S = y''_S$, $f_1(y'') = f_1(x') = 1$ and $f_2(y'') = f_2(x') = 1$. But this contradicts the fact that at a node $f = f_1 \wedge f_2$ either $f_1(y'') = 0$ or $f_2(y'') = 0$ (or both).

Proof of equivalent claims for an $\vee$ gate is similar except for the fact that Alice modifies her input.

Thus, using the above protocol we are guaranteed to solve $\mathbf{KW}^+(f)$. Communication complexity of the protocol is upper bounded by $\mathbf{Depth}(C) \times (4w+1)$. Communication cost of a round is $4w + 1$. Because if any of the children is a negation gate then we have to send its orientation along with the orientation of its complement. The protocol clearly stops after $\mathbf{Depth}(C)$ many rounds. □

4 Dense Orientation

Currently our depth lower bound technique cannot handle orientations of weight $\frac{n}{\log^k n}$ or more for obtaining $\omega(\log^k n)$ lower bounds. In light of this, we explore the usefulness of densely oriented gates in a circuit. First we prove that any polynomial sized circuit can be transformed into an equivalent circuit of polynomial size but having only $O(n \log n)$ gates of non-zero orientation by studying the connection between orientations and negations. Next we present a limitation of our technique in a circuit having only two gates of non-zero (but dense) orientation. Thus, strengthening of our technique will have to use some property of the function being computed. Finally we show how to use a property of **CLIQUE** function to slightly get around the limitation.

4.1 From Negation Gates to Orientation

Since weight of the orientation can be thought of as a measure of non-monotonicity in a circuit, a natural question to explore is the connection between the number of negations and number of non-zero orientations required to compute a function f. We show the following:

Theorem 10. *For any function $f : \{0,1\}^n \to \{0,1\}$, if there is a circuit family $\{C_n\}$ computing f with $t(n)$ negations then there is also a circuit family $\{C'_n\}$ computing f such that $\mathbf{Size}(C'_n) \leq 2^t \times (\mathbf{Size}(C_n) + 2^t) + 2^t$, and there are at most $2^{t-1}(t+2) - 1$ internal gates whose orientation is non-zero.*

Proof. In C_n replace input of each negation by new variables $y_1, \ldots, y_t$, thus obtaining a circuit $C''_n(x_1, \ldots, x_n, y_1, \ldots, y_t)$. Let $g_1, \ldots, g_t$ be the input to the t negation gates (in topologically sorted order) in C_n. Note that for each setting of $y_1, \ldots, y_t$ to some $b \in \{0,1\}^t$, $C''_n(x, b)$ is monotone circuit computing a monotone function on $x_1, \ldots, x_n$. Hence the orientation of each internal gate in $C''_n(x, b)$ is zero. Let $g_{i,b}$ for $i \in [t], b \in \{0,1\}^t$ denote the monotone function computed by the sub-circuit C_{g_i} of C_n rooted at gate g_i, where $g_1, \ldots, g_{i-1}$ are set to $b_1, \ldots, b_{i-1}$ respectively. Thus we can write f as:

$$f(x_1, \ldots, x_n) = \bigvee_{b \in \{0,1\}^t} \left(\bigwedge_{i=1}^{t} g_{i,b}^{b_i}(x) \right) C''_n(x, b),$$

where g^0 denotes $\overline{g}$ and g^1 denotes g. When $t = 1$, then the above expression becomes $f(x) = g(x)C(x,1) + \overline{g}(x)C(x,0)$. Thus in this case, $\neg$ computing $\overline{g}$, $\wedge$ computing $\overline{g}(x)C(x,0)$ and the root gate if it is computing a non-zero orientation function are the only gates with non-zero orientation. Hence when $t = 1$ the circuit has at most three gates with non-zero orientation if the root gate is non-monotone and two otherwise.

Consider the above formulation of a circuit C' computing f. Clearly $\mathbf{Size}(C') \leq 2^t \times (\mathbf{Size}(C_n) + 2^t) + 2^t$. All internal gates in $C''_n(a, b)$ are monotone. Non-zero orientation is needed for computing:

- $\displaystyle\bigwedge_{i \in [t], b_i = 0} \overline{g_{i,b}}$

- $\wedge$ of $\bigwedge_{i\in[t],b_i=0} \overline{g_{i,b}}$ with $\bigwedge_{i\in[t],b_i=1} g_{i,b} \wedge C''(x,b)$

- the $\vee$-tree, computing $\sum$ of 2^t terms which are potentially of non-zero orientation.

For computing $\bigwedge_{i\in[t],b_i=0} \overline{g_{i,b}}$, we need an $\wedge$ tree of $t - |b|_1$ many leaves. Number of internal nodes in the tree is $t-|b|_1-1$ (for $t > 1$). To compute the $\wedge$ of this intermediate product with $\bigwedge_{i\in[t],b_i=1} g_{i,b} \wedge C''(x,b)$ one more gate is need. Thus total number of gates needed is $t-|b|_1$. Let us call number of such gates K_1. By the above analysis, $K_1 = \sum_{b\in\{0,1\}^t}(t - |b|_1) = t \times 2^{t-1}$. The remaining gates are the internal gates in the $\vee$ tree implementing the sum of terms. Since there are 2^t leaves, number of internal nodes in the tree, say K_2 is $2^t - 1$. Hence total number of nodes with non-zero orientation is at most $K_1 + K_2 = 2^{t-1}(t+2) - 1$. $\square$

Remark 1. *In conjunction with the result of Fisher [7], this implies that it is enough to prove lower bounds against circuits with at most $O(n \log n)$ internal nodes of dense orientations, to obtain lower bounds against the general circuits.*

4.2 Power of Dense Orientation

We show that even as few as two densely oriented internal gates can help to reduce the depth from super poly-log to poly-log for some functions.

Theorem 11. *There exists a monotone Boolean function f such that it cannot be computed by poly-log depth monotone circuits, but there is a poly-log depth circuit computing it such that at most two internal gates have non-zero orientation β.*

Proof. It is known [15] that **PMATCH** does not have monotone circuits of poly-log depth and if arbitrary negations are allowed then there is a $O(\log^2 n)$ depth circuit computing **PMATCH** [13]. Monotone function f claimed in the theorem is obtained from poly-log depth circuit C computing **PMATCH**. Fischer's theorem guarantees that C will have at most $\log n$ negations.

If there is a poly-log depth circuit having exactly one negation, then Theorem 4 can be applied to get a circuit of poly-log depth having at most two non-zero orientation gates. Otherwise, the circuit has $t \geq 2$ negations, and there is no poly-log depth circuit computing the same function with one negation. Let g_1 denote the input to the first negation gate(in the topological sorted order) in C. From C obtain C' by replacing g_1 with a new variable, say y_1. Let C'_0, C'_1 denote the circuits obtained by setting y_1 to 0,1 respectively. The corresponding functions f_0, f_1 need not be monotone. Hence we define monotone functions f'_0, f'_1 from f_1, f_0 :

$$\begin{aligned} f'_0(x) &= f_0(x) \vee g_1(x) \\ f'_1(x) &= f_1(x) \wedge g_1(x) \end{aligned}$$

When $g_1(x) = 0$, $f_0(x) = f(x)$ and when $g_1(x) = 1$, $f_0'(x) = 1$. Hence f_0' is monotone. A similar argument can be used to establish that f_1' is monotone. Note that both f_0', f_1' have poly-log depth circuits computing it with at most $t-1$ negation gates.

We claim that one of f_0', f_1' does not have a monotone circuit of poly-log depth. Otherwise from poly-log depth monotone circuits computing f_0', f_1' and the monotone circuit of poly-log depth computing g_1 we can get a poly-log depth circuit computing f with one negation : use $\overline{g_1}(x)$ as a selector to select $f_1'(x)$ or $f_0'(x)$ as which is appropriate. This circuit computes f because, by definition, $(g_1(x) \wedge f_1'(x)) \vee (\overline{g_1}(x) \wedge f_0'(x)) = f(x)$. This contradicts our assumption that there is no circuit of poly-log depth computing f with one negation.

Applying the procedure once, we get a monotone function f' which has a $t-1$ negation poly-log depth circuit computing it, but it has no monotone circuit of poly-log depth computing it. If the function f' has a poly-log depth circuit with one negation then Theorem 4 can be applied to get the desired function. Otherwise apply the procedure on f' as f' is a monotone function which does not have any poly-log depth circuit with at most one negation computing it. Applying the procedure at most t ($t \leq \log n$) times we get to a monotone function f' having a poly-log depth circuit with one negation, but has no monotone poly-log depth circuit computing it. Applying Theorem 4, a poly-log depth circuit with at most two non-zero orientation gates is obtained. $\square$

This theorem combined with the sparse orientation protocol implies that the two non-zero orientations β_1, β_2 is such that $|\beta_1| + |\beta_2|$ is not only non-zero but is super poly-log. Because our protocol will spend $|\beta_1| + |\beta_2|$ for handling these two gates, and on the remaining gates in the circuit it will spend 1 bit each. Hence the cost of the sparse orientation protocol will be at most $|\beta_1| + |\beta_2| + \mathbf{Depth}(C)$, thus $|\beta_1| + |\beta_2|$ is at least $\mathbf{KW}^+(f) - \mathbf{Depth}(C)$ which is super poly-log as $\mathbf{Depth}(C)$ is poly-log and $\mathbf{KW}^+(f)$ is super poly-log.

Remark 2. *By Theorem 11 we get a function which has an NC^2 circuit with two non-zero orientation gates which has no monotone circuit of poly-log depth. Thus our bounds cannot be strengthened to handle higher weight without incorporating the specifics of the function being computed. In section 4.3, we rescue the situation slightly using the specific properties of the* **CLIQUE** *function.*

Remark 3. *The proof of Theorem 11 also implies that there is a monotone function f (not explicit) such that there is a one negation circuit in NC^2 computing it, but any monotone circuit computing f requires super-poly-log depth.*

4.3 Lower Bounds for CLIQUE function

The number of gates with high orientations can be arbitrary in general. In this subsection we give a proof for Theorem 3. We first extend our technique to handle the low weight negations efficiently so that we get a circuit on high weight negations (see Lemma 3 below). To complete the proof of Theorem 3, we appeal to depth lower bounds against negation-limited circuits computing f.

Lemma 3. *For any circuit family $\mathcal{C} = \{C_n\}$ computing a monotone function f where there are k negations in C_n computing functions which are sensitive only on w inputs (i.e., the orientation of their input as well as their output is at most w) and the remaining ℓ negations compute functions of arbitrary orientation:* $\mathbf{Depth}(C_n) \geq \mathbf{Depth}_{2^\ell}(f) - kw - \ell$

Proof. First the given circuit C_n is transformed by replacing the ℓ negations with new inputs variables $y_1, y_2, y_3, \ldots, y_\ell$. Note that for each setting of $y_1, \ldots, y_\ell$ to some $b \in \{0,1\}^\ell$, $C_n^{''}(x,b)$ is monotone circuit computing a monotone function on $x_1, \ldots, x_n$. Hence $C_n^{''}(x,b)$ is a monotone circuit. Let $g_1, \ldots, g_\ell$ be the inputs to the ℓ negation gates (in topologically sorted order) in C_n. Let $g_{i,b}$ for $i \in [\ell], b \in \{0,1\}^\ell$ denote the monotone function computed by the sub-circuit C_{g_i} of C_n rooted at gate g_i, where $g_1, \ldots, g_{i-1}$ are set to $b_1, \ldots, b_{i-1}$ respectively. It is easy to verify that,

$$f(x_1, \ldots, x_n) = \bigvee_{b \in \{0,1\}^\ell} \left(\bigwedge_{i=1}^{\ell} g_{i,b}^{b_i}(x) \right) C_n^{''}(x,b)$$

where g^0 denotes $\overline{g}$ and g^1 denotes g.

Let K denote the k negation gates stated in the lemma. We play the game as follows: To begin, for each negation gate h in K, Alice sends w bits of her input on which h is sensitive. This takes up kw bits of communication. Bob checks if there is any index i in the input bits revealed to him by Alice where $x_i = 1, y_i = 0$ and accepts if that is the case. Otherwise, he sends to Alice a bit indicating that no such index exists. In this case, on any index i where at least one of the negation gates in K is sensitive, $x_i \leq y_i$. Also, Bob knows x_i for such indices. He sets $y_i = x_i$ for all such i, obtaining a new input y'. By monotonicity of f, and the fact that $y' \leq y$, $f(y') = 0$.

Now they play the standard Karchmer-Wigderson game starting at the root of the circuit described by the above equation. Since we made sure that for any of the k negation gates x, y' agree on the w bits on which they are sensitive, these negation gates evaluates to the same value on x and y'. Since the standard Karchmer-Wigderson game traces a path in the circuit where evaluation of x and y' are different, we are guaranteed to never reach any of the gates in K. Note that the circuit corresponding to a Karchmer-Wigderson protocol is a formula and the first ℓ levels of the circuit described in the equation is also a formula. Hence the circuit corresponding to the protocol will have only 2^ℓ negation gates corresponding to the ℓ negation gates in the circuit C_n we started with. Thus we obtain a 2^ℓ negation gate circuit computing f of depth $kw + \ell + \mathbf{Depth}(C_n)$. Hence the theorem. $\square$

By a straight forward application of technique used in [5] to prove size lower bounds against circuits with limited negations we obtain the size version of following lemma (For completeness, we include the relevant part in the Appendix A).

Lemma 4. *For any circuit C computing* $\mathbf{CLIQUE}(n, n^{\frac{1}{6\alpha}})$ *with ℓ negations where $\ell \leq 1/6 \log\log n$ and $\alpha = 2^{\ell+1} - 1$,*

$$\mathbf{Depth}_\ell(f) \;\; \geq \;\; n^{\frac{1}{81\alpha}}$$

Combining Lemma 3 and Lemma 4 completes the proof of Theorem 3.

5 Structural Restrictions on Orientation

In this section we study structural restrictions on the orientation and prove stronger lower bounds.

5.1 Restricting the Vertex Set indexed by the Orientation

We first consider restrictions on the set of vertices[4] indexed by the orientation - in order to prove Theorem 5 stated in the introduction. As in the other case, we argue the following lemma, which establishes the trade-off result. By using the lower bound for $\mathbf{KW}^+$ games for **CLIQUE** function, the theorem follows.

Lemma 5. *Let C be a circuit of depth d computing* **CLIQUE***, with each gate computing a function whose orientation is such that the number of vertices of the input graph indexed by the orientation β is at most $\frac{w}{\log n}$, then d is $\Omega\left(\frac{\mathbf{KW}^+(f)}{4w+1}\right)$.*

Proof. It is enough to solve the $\mathbf{KW}^+(f)$ on the min-term, max-term pairs which in case of $\mathbf{CLIQUE}(n,k)$ is a k-clique and a complete $k-1$-partite graph. We play the same game as in the proof of Theorem 1, but instead of sending edges we send vertices included in the edge set indexed by β with some additional information. If it is Alice's turn, then x'_β defines an edge sub-graph of her clique. Both Alice and Bob know β and hence knows which vertices are spanned by edges $e_{u,v}$ such that $\beta_{e(u,v)} = 1$. So Alice can send a bit vector of length at most w (in the case of Alice we can handle up to w), indicating which of these vertices are part of her clique. This information is enough for Bob to deduce whether any $e_{u,v}$ indexed by β is present in Alice's graph or not. Since Bob makes sure that $x'_\beta = y'_\beta$ by modifying his input, and Alice keeps her input unchanged, Alice knows what modifications Bob has done to his graph.

Similarly on Bob's turn, he sends the vertices in the partition induced by y_β and the partition number each vertex belongs to (hence the $\log n$ overhead for Bob) to Alice. With this information Alice can deduce whether any $e_{u,v} \in \beta$ is present in Bob's graph or not. Inductively they maintain that they know of the changes made to other parties input in each round. Hence the game proceeds as earlier. This completes the proof of the theorem. □

5.2 Restricting the Orientation to be Uniform

In this section, we consider the circuits where the orientation is uniform and study its structural restrictions. We proceed to the proof of Theorem 6. **Theorem 6:** *Let C be a circuit computing the* **CLIQUE** *function with uniform orientation $\beta \in \{0,1\}^n$ such that there is subset of vertices U and $\epsilon > 0$ such that $|U| \geq \log^{k+\epsilon} n$ for which $\beta_e = 0$ for all edges e within U, then C must have depth $\omega(\log^k n)$.*

[4] Notice that the input variables to the **CLIQUE** function represents the edges. This makes the results of this section incomparable with the depth lower bounds of [14].

Proof. We prove by contradiction. Suppose there is a circuit C of depth $c \log^k n$. In the argument below we assume $c = 1$ for simplicity. Without loss of generality, we assume that $|U| = \log^{k+\epsilon} n$. Fix inputs to circuit C in the following way:

- Choose an arbitrary $K_{\frac{n}{2}-\frac{|U|}{2}}$ comprising of vertices from $[n] \setminus U$ and set those edges to 1.
- For every edge in $\binom{[n]\setminus U}{2}$ which is not in the clique chosen earlier, set to 0.
- For every edge between $[n] \setminus U$ and U set it to 1.

Since every edge $e(x, y)$ which has $\beta_e = 1$ has at least one of the end points in $[n] \setminus U$, by above setting, all those edges are turned to constants. Thus we obtain a monotone circuit C'' computing **CLIQUE**$(|U|, \frac{|U|}{2})$ of depth at most $(\log n)^k$. In terms of the new input, $(\log n)^k = ((\log n)^{k+\epsilon})^{\frac{k}{k+\epsilon}} = (|U|)^{\frac{k}{k+\epsilon}}$, this contradicts the Raz-Wigderson [15] lower bound of $\Omega(|U|)$, as $\frac{k}{k+\epsilon} < 1$ for $\epsilon > 0$. $\square$

Note that for Clique function, with the above corollary we can handle up to weight $\frac{n^2}{(\log n)^{2+2\epsilon}}$ if the vertices spanned by β is up to $\frac{n}{(\log n)^{1+\epsilon}}$ and still get a lower bound of $(\log n)^{1+\epsilon}$. This places us a little bit closer to the goal of handling β of weight n^2, from handling just $(\log n)^{1+\epsilon}$.

A contrasting picture: Any function has a circuit with a uniform orientation $\beta = 1^n$ ($|\beta| = n$). We show that the weight of the orientation can be reduced at the expense of depth, when the circuit is computing the **CLIQUE** function.

Theorem 7: *If there is a circuit C computing* **CLIQUE** *with depth d then for any set of $c \log n$ vertices U, there is an equivalent circuit C' of depth $d + c \log n$ with orientation β such that none of the edges $e(u, v)$, $u, v \in U$ has $\beta_{e(u,v)} = 1$.*

Proof. The proof idea is to devise a **KW** protocol based on circuit C such that for $e(u, v)$ where $u, v \in U$ the protocol is guaranteed to output in the monotone way, i.e., $x_{e(u,v)} = 1$ and $y_{e(u,v)} = 0$. The modified protocol is as follows:

- Alice chooses an arbitrary clique $K_{\frac{n}{2}} \in G_x$ (which she is guaranteed to find as $x \in f^{-1}(1)$). She then obtains x' by deleting edges $e(x, y)$ from $\binom{U}{2}$ which are outside the chosen clique $K_{\frac{n}{2}}$. Note that since $K_{\frac{n}{2}} \in G_{x'}$, $f(x') = 1$.
- Alice then sends the characteristic vector of vertices in $K_{\frac{n}{2}} \cap U$ which is of length at most $c \log n$ to Bob.
- Bob then obtains y' from y by removing edges in $\binom{U}{2}$ which are outside the clique formed by $K_{\frac{n}{2}} \cap \binom{U}{2}$. By monotonicity of **CLIQUE** $f(y') = 0$.
- If there is an edge $e(u, v) \in K_{\frac{n}{2}} \cap \binom{U}{2}$ which is missing from y' Bob outputs the index $e(u, v)$. Otherwise they run the standard Karchmer-Wigderson game on x', y' using the circuit C to obtain an $e(x, y)$ such that $e(x, y)$ is exclusive to either $G_{x'}$ or $G_{y'}$.

The cost of the above protocol is $d + c\log n$. For any $e(u,v) \in E(G) \setminus \binom{U}{2}$, $x'_{e(u,v)} = x_{e(u,v)}$ and $y'_{e(u,v)} = y_{e(u,v)}$. The protocol never answers non-monotonically$(i, x'_i = 0, y'_i = 1)$ for an edge $e(u,v)$ with $u, v \in U$. Because our protocol ensures that for any $e \in \binom{U}{2}$, $x'_e \geq y'_e$, ruling out such a possibility. By the connection between $\mathbf{KW}(f)$ and circuit depth, we get a circuit having desired properties. □

Thus we get the following corollary.

Corollary 1. *If there is a circuit $C \in \mathsf{NC}^k$ computing $\mathbf{CLIQUE}(n,k)$, then there is a circuit $C' \in \mathsf{NC}^k$ of uniform orientation β computing $\mathbf{CLIQUE}(n,k)$ such that there are $(c\log n)^k$ vertices V' with none of the edges $e(u,v)$having $\beta_{e(u,v)} = 1$.*

Proof. It follows by setting $d = O((\log n)^k)$ and modifying the protocol to work over a V' of size $(c\log n)^k$. The analysis and proof of correctness of the protocol remains the same, but the communication cost becomes $O((\log n)^k) + (c\log n)^k = O((\log n)^k)$. □

In other words, if we improve Theorem 5 to the case when the orientation "avoids" a set of $\log n$ vertices (instead of $(\log n)^{(1+\epsilon)}$ as done), it will imply $\mathsf{NC}^1 \neq \mathsf{NP}$.

6 Discussion and Open Problems

In this work, we studied lower bounds against non-monotone circuits with a new measure of non-monotonicity - namely the orientation of the functions computed at each gate of the circuit. As the first step, we proved that the lower bound can be obtained by modifying the Karchmer-Wigderson game. We studied the weight of the orientation of the functions at internal gates as a parameter of the circuit, and explored the usefulness of densely oriented gates. We also showed the connections between negation limited circuits and orientation limited circuits. A main open problem that arises from our work is to improve upon the weight restriction of the orientation vector $(\Omega(\frac{n}{\log n}))$ for which we can prove depth lower bounds.

7 Acknowledgements

We thank the anonymous referees for the useful comments.

References

[1] Eric Allender. Circuit Complexity before the Dawn of the New Millennium. In *Proceedings of FST & TCS*, volume 1180 of *LNCS*, pages 1–18. 1996.

[2] Eric Allender. Cracks in the defenses: Scouting out approaches on circuit lower bounds. In *Proc. of 3rd CSR*, pages 3–10, 2008.

[3] Noga Alon and Ravi B. Boppana. The Monotone Circuit Complexity of Boolean Functions. *Combinatorica*, 7(1):1–22, 1987.

[4] K. Amano and A. Maruoka. Potential of the approximation method. In *Proceedings of 37th Annual Symposium on Foundations of Computer Science*, pages 431 –440, oct 1996.

[5] Kazuyuki Amano and Akira Maruoka. A superpolynomial lower bound for a circuit computing the clique function with at most (1/6) log log n negation gates. *SIAM Journal on Computing*, 35(1):201–216, 2005.

[6] Jack Edmonds. Paths, trees, and flowers. *Canad. J. Math.*, 17:449–467, 1965.

[7] Michael Fischer. The Complexity of Negation-limited Networks — a Brief Survey. In *Automata Theory and Formal Languages 2nd GI Conference Kaiserslautern*, volume 33 of *Lecture Notes in Computer Science*, pages 71–82. 1975.

[8] Russell Impagliazzo, Ramamohan Paturi, and Michael E. Saks. Size-depth tradeoffs for threshold circuits. *SIAM Journal of Computing*, 26(3):693–707, 1997.

[9] Kazuo Iwama and Hiroki Morizumi. An explicit lower bound of 5n-o(n) for boolean circuits. In *MFCS*, volume 2420 of *LNCS*, pages 353–364. 2002.

[10] Stasys Jukna. On the minimum number of negations leading to super-polynomial savings. *Information Processing Letters*, 89(2):71 – 74, 2004.

[11] Stasys Jukna. *Boolean Function Complexity: Advances and Frontiers*, volume 27 of *Series: Algorithms and Combinatorics*. Springer New York Inc., 2012.

[12] Mauricio Karchmer and Avi Wigderson. Monotone Circuits for Connectivity Require Super-logarithmic Depth. In *STOC*, pages 539–550, 1988.

[13] László Lovász. On determinants, matchings, and random algorithms. In *Symposium on Fundamentals of Computation Theory (FCT)*, pages 565–574, 1979.

[14] Ran Raz and Avi Wigderson. Probabilistic communication complexity of boolean relations. In *Proc. of the 30th FOCS*, pages 562–567, 1989.

[15] Ran Raz and Avi Wigderson. Monotone circuits for matching require linear depth. *Journal of ACM*, 39(3):736–744, July 1992.

[16] AA Razborov. Lower Bounds for Monotone Complexity of Some Boolean Functions. *Soviet Math. Doklady.*, pages 354–357, 1985.

[17] Alexander A Razborov. Lower bounds on monotone complexity of the logical permanent. *Mathematical Notes*, 37(6):485–493, 1985.

[18] Johan Håstad. The shrinkage exponent of de morgan formulas is 2. *SIAM Journal on Computing*, 1998.

[19] Heribert Vollmer. *Introduction to Circuit Complexity: A Uniform Approach*. Springer New York Inc., 1999.

Appendix

A Proof of Lemma 4 - Choice of parameters in [5]

In this section we give the arguments for Lemma 4. Since the trade-off result stated in Lemma 4 is not explicitly stated and proved in [5], in this section, we present the relevant part of the proof technique in [5] with careful choice of parameters obtaining the trade-off. For consistency with notation used in [5], *for the remainder of this section we will be denoting the number of vertices in the graph by m.*

The main idea in [5] is to consider the boundary graph of a function f, defined as $G_f = \{(u,v) | \Delta(u,v) = 1, f(u) \neq f(v)\}$ where $\Delta(u,v)$ is the hamming distance. They prove that if there is a t negations circuit C computing f then the boundary graph f must be covered by union of boundary graphs of 2^{t+1} functions obtained by replacing the negations in C by variables and considering the input functions of t negation gates and the output gate where the negations in the sub-circuit considered are restricted to constants.

They prove that,

Lemma 6. *[5, Theorem 3.2] Let f be a monotone function on n variables. For any positive integer t,*

$$\mathbf{Size}_t(f) \geq \min_{F' = \{f_1, \ldots, f_\alpha\} \subseteq \mathcal{M}^n} \left\{ \max_{f' \in F'} \{size_{mon}(f')\} \mid \bigcup_{f' \in F'} G(f') \supseteq G(f) \right\}$$

where $\alpha = 2^{t+1} - 1$ and $G(f')$ denotes the boundary graph of the function f'.

The size lower bound they derive crucially depends on the following lemma which states that no circuit of "small" size can "approximate" clique in the sense that either it rejects all the "good" graphs or accepts a huge fraction of "bad" graphs.

Lemma 7. *[5, Theorem 4.1] Let s_1, s_2 be positive integers such that $64 \leq s_1 \leq s_2$ and $s_1^{1/3} s_2 \leq \frac{m}{200}$. Suppose that C is a monotone circuit and that the fraction of good graphs in $I(m, s_2)$ such that C outputs 1 is at least $h = h(s_2)$. Then at least one of the following holds:*

- *The number of gates in C is at least $(h/2)2^{s^{1/3}/4}$.*
- *The fraction of bad graphs in $O(m, s_1)$ such that C outputs 0 is at most $2/s_1^{1/3}$.*

where a "good" graph in $I(m, s_2)$ is a clique of size s_2 on m vertices and no other edges and a "bad" graph in $O(m, s_1)$ is an $(s_1 - 1)$-partite graph where except for at most one partition the partitions are balanced and of size $\lceil \frac{m}{s_1 - 1} \rceil$ each.

Lemma 8. *For any circuit C computing $\mathbf{CLIQUE}(m, m^{\frac{1}{6\alpha}})$ with t negations with $t \leq 1/6 \log\log m$, size of C is at least $2^{m^{\frac{1}{81\alpha}}}$ where $\alpha = 2^{t+1} - 1$.*

Proof. The proof is similar to the proof ([5, Theorem 5.1]) by Amano and Maruoka except for change of parameters. Assume to the contrary that there is a circuit C with at most t negations computing $\mathbf{CLIQUE}(m, m^{\frac{1}{6\alpha}})$ with size M, $M < 2^{m^{\frac{1}{81\alpha}}}$. By Lemma 6 there are $\alpha \triangleq 2^{t+1}-1$ functions $f_1, \ldots, f_\alpha$ of size at most M (as they are obtained by restrictions of the circuit C) such that $\cup_{i=1}^{\alpha} G(f_i) \supseteq G(f)$. Let $s = m^{\frac{1}{6\alpha}}$ and let $l_0, l_1, \ldots, l_\alpha$ be a monotonically increasing sequence of integers such that $l_0 = s, l_\alpha = m$ and $l_i = m^{1/10+(i-1)/(3\alpha)}$. Note that $s^{1/3} l_i \leq l_{i+1}$ as $s^{1/3} l_i = m^{1/(18\alpha)+1/10+(i-1)/(3\alpha)} < m =^{1/10+(i)/(3\alpha)} = l_{i+1}$. Also $\left[l_0 = s = m^{\frac{1}{6\alpha}}\right] < \left[l_1 = m^{1/10}\right]$ as $\alpha = 2^{t+1}-1 \geq 2^2-1$, $l_{\alpha-1} < m^{1/10+1/3} < m$. Thus, $l_0 < l_1 < \cdots < l_i < l_{i+1} < \cdots < l_\alpha$. The definition of "bad" graphs and "good" graphs at layer l_i remains the same as in [5]. Note that [5, Corollary 5.2] is true for our choice of parameters as $s^{1/3} l_{i-1} \leq l_i$. Equations 5.1 to 5.3 of [5] is valid in our case also as these equations does not depend on the value of the parameters. The definition of a dense set remains the same, and $h \geq \frac{1}{\alpha} \geq \frac{1}{m}$ (as $m \geq \log m \geq \alpha$) is such that $(h/2) 2^{s^{1/3}/4} \geq \frac{1}{m} 2^{m^{\frac{1}{18\alpha}}/4}$ is strictly greater than $M = 2^{m^{\frac{1}{81\alpha}}}$. Hence Equation 5.4 of [5] is also true in our setting. Claim 5.3 of [5] is independent of choice of parameters, hence is true in our setting also.

Claim 12. *[5, Claim 5.3]*

Suppose $c_1 > 1$ and $c_2 > 1$. Put $c_3 = \alpha$. Let $f_1, \ldots, f_{c_3}$ be the monotone functions such that $\cup_{i=1}^{c_3} G(f_i) \supseteq G(\mathbf{CLIQUE}(m, s))$ and $size_{mon}(f_i) \leq M$ for any $1 \leq i \leq c_3$. Suppose that for distinct indices $i_1, \ldots, i_k \in [c_3]$,

$$\Pr_{L_k \in \mathcal{L}_k}\left[\Pr_{u \in O_{L_k}}[f_{i_1}(u) = \cdots = f_{i_k}(u) = 1] \geq \frac{1}{c_1}\right] \geq \frac{1}{c_2}$$

holds. If $c_1 c_2 c_3 \leq s_1^{1/3}/8$, then there exists $i_{k+1} \in [c_3] \setminus \{i_1, \ldots, i_k\}$ such that

$$\Pr_{L_{k+1} \in \mathcal{L}_{k+1}}\left[\Pr_{u \in O_{L_{k+1}}}[f_{i_1}(u) = \cdots = f_{i_k}(u) = 1] \geq \frac{1}{4c_1 c_2 c_3}\right] \geq \frac{1}{2c_1 c_2}$$

Now for any $k \in [\alpha]$ there are k distinct indices $i_1, \ldots, i_k \in [\alpha]$ such that

$$\Pr_{L_k \in \mathcal{L}_k}\left[\Pr_{u \in O_{L_k}}[f_{i_1}(u) = \cdots = f_{i_k}(u) = 1] \geq \frac{1}{2^{k^2(t+2)}}\right] \geq \frac{1}{2^{k(t+2)}} \tag{1}$$

The proof is by induction on k. Base case is when $k = 1$ and follows from Equation 5.4 of [5] which is established to be true in our setting also. Suppose the claim holds for $k \leq l$ and let $k = l + 1$. From induction hypothesis we get that

$$\Pr_{L_l \in \mathcal{L}_l}\left[\Pr_{u \in O_{L_l}}[f_{i_1}(u) = \cdots = f_{i_l}(u) = 1] \geq \frac{1}{2^{l^2(t+2)}}\right] \geq \frac{1}{2^{l(t+2)}} \tag{2}$$

Like in [5] put $c_1 = 2^{l^2(t+2)}, c_2 = 2^{l(t+2)}$ and $c_3 = \alpha$. Note that the bounds $4c_1 c_2 c_3 \leq 2^{(l+1)^2(t+2)}$, $2c_1 c_1 \leq 2^{(l+1)(t+2)}$ and $c_1 c_2 c_3 \leq 2^{2^{3t}}/8$ are valid in our setting also as they do not depend on values of these parameters. Since $t \leq 1/6 \log\log m$, $2^{3t} \leq (\log m)^{1/3}$ and

$2^{2^{3t}} \leq 2^{(\log m)^{1/3}}$ whereas $s^{1/3}$ is $m^{\frac{1}{18\alpha}} \geq 2^{(\log m)(\frac{1}{18(\log m)^{1/6}})} = 2^{(\log m)^{5/6}/18} > 2^{(\log m)^{1/3}}$. Hence $s^{1/3}/8 \geq 2^{2^{3t}}/8$. . Thus Claim 12 applies giving us

$$\Pr_{L_{l+1} \in \mathcal{L}_{l+1}} \left[\Pr_{u \in O_{L_{l+1}}} \left[f_{i_1}(u) = \cdots = f_{i_{l+1}}(u) = 1 \right] \geq \frac{1}{2^{(l+1)^2(t+2)}} \right] \geq \frac{1}{2^{(l+1)(t+2)}} \tag{3}$$

The proof of the main theorem is completed by noting that $\mathcal{L}_\alpha = \{V\}$ and setting k in Equation (1) to α gives $Pr_{u \in O_V}[\forall i \in [\alpha], f_i(u) = 1] > 0$. Thus there exists a bad graph u belonging to $\mathbf{CLIQUE}(m,s)^{-1}(0)$ on which all of $f_1, \ldots, f_\alpha$ outputs 1, and hence (u, u^+), where $u^+ \in \mathbf{CLIQUE}(m,s)^{-1}(1)$ is a graph obtained from u by adding an edge, which is in $G(f)$ is not covered by any of the $G(f_i)$'s. A contradiction. Hence the proof. □ □

Since for a bounded fan-in circuit size lower bound of $2^{m^{\frac{1}{81\alpha}}}$ implies a depth lower bound of $m^{\frac{1}{81\alpha}}$ we have,

Lemma 4: *For any circuit C computing $\mathbf{CLIQUE}(m, m^{\frac{1}{6\alpha}})$ with ℓ negations where $\ell \leq 1/6 \log\log m$, where $\alpha = 2^{\ell+1} - 1$*

$$\mathbf{Depth}_\ell(f) \geq m^{\frac{1}{81\alpha}}$$

ECCC ISSN 1433-8092

http://eccc.hpi-web.de