On Lower Bounds for Read-k-Times Branching Programs*

A. Borodin
Department of Computer Science
University of Toronto
Toronto, Ontario, CANADA M5S 1A4

A. Razborov
Steklov Mathematical Institute
Vavilova 42, 117966, GSP–1
Moscow, RUSSIA

R. Smolensky
94 Alden Road
Concord, MA
U.S.A.

Abstract: A syntactic read-k-times branching program has the restriction that no variable occurs more than k times on any path (whether or not consistent) of the branching program. We first extend the result in [30], to show that the "$n/2$ clique only function", which is easily seen to be computable by deterministic polynomial size read-twice programs, cannot be computed by nondeterministic polynomial size read-once programs, although its complement can be so computed. We then exhibit an explicit Boolean function f such that every nondeterministic syntactic read-k-times branching program for computing f has size $\exp\left(\Omega\left(\frac{n}{4^k k^3}\right)\right)$.

*Supported by the Natural Sciences and Engineering Research Council of Canada and by the Information Technology Research Centre of Ontario. This research was performed while the second author was a visitor at the University of Toronto.

1 Introduction

Complexity theory, and in particular Boolean complexity theory, has had very limited success in the attempt to prove lower bounds for "explicitly defined" functions (e.g., Boolean functions in the class NP). With regard to general models such as Boolean circuits or branching programs, existing results either are quite weak or depend essentially on restrictions placed on these models. In particular, with and without restrictions the branching program model, introduced in [17, 19], has been studied extensively in the last decade (see for example [28]). In fact, the closely related models of switching networks and switching-and-rectifier networks were already well studied models preceding the study of branching programs (see e.g. [12], [25]). A survey of known lower bounds for all of these models (both with and without restrictions) can be found in [23].

Although our goal is to derive bounds for Boolean functions $f : \{0,1\}^n \to \{0,1\}$, it will be convenient to consider an R-way variant [9] of branching programs for computing functions $f : R^n \to \{0,1\}$ where R is a finite set. A nondeterministic (n-input) R-way branching program P (hereafter denoted n.b.p. when n and R are understood) is a directed acyclic multi-graph with a distinguished sink node labeled "accept" . For each nonsink node, the edges directed out of the node are either unlabeled or labeled "$x_i = d$" for some $d \in R$ and $i \in \{1, \ldots, n\}$. We think of each node as a state of the computation and the label "$x_i = d$" indicates that only inputs satisfying $x_i = d$ may follow this edge in the computation. Unlabeled edges allow all inputs to proceed. Our lower bound results hold for this nondeterministic model. A deterministic branching program (denoted b.p.) is a n.b.p. with the additional restriction that there are no unlabeled edges and for each nonsink node there is a variable x_i such that all out-edges from this node are labeled by "$x_i = d$" for some $d \in R$ and for each d there is exactly one such labeled edge. For a branching program P, we define size(P) as the number of labeled edges in P and length(P) as the length of the longest path in P. A n.b.p. P computes a function $f : R^n \to \{0,1\}$, in the obvious way; that is, $f(a_1, \ldots, a_n) = 1$ iff there is a computation on $< a_1, \ldots, a_n >$ leading to the accepting state. It is well understood that length(P) and $\log_2$ size(P) are nonuniform measures corresponding (simultaneously) to sequential time T and space S. Of course, in the nonuniform branching program model every function $f : R^n \to \{0,1\}$ can be computed in length n so that branching program length only becomes interesting in the context of restricted branching program size. Following [9], a $T \cdot S = \Omega(n^2)$ tradeoff result is established in [8] for the n-input, n-output sorting function $f : R^n \to R^n$ where $R = \{1, \ldots, n\}$. Such a result immediately yields corresponding lower bounds for Boolean multi-output functions. These results are further developed and extended in a number of papers [2], [1], [18], [29]. However, there are no such lower bounds presently known for single output or decision problems. A "complexity breakthrough" would be to exhibit an explicitly defined class of functions $\langle f_n | n \geq 0 \rangle$ such that for any sequence of (deterministic) branching programs $\langle P_n | n \geq 0 \rangle$ computing $< f_n >$, size$(P_n) = n^{O(1)}$ implies length$(P_n) \neq O(n)$; that is, logarithmic space implies nonlinear time (or equivalently, linear time implies nonlogarithmic space).

There are two restricted classes of branching programs for which such tradeoff results have been established, namely oblivious branching programs [3], [5] and read-once branching programs. An oblivious b.p. is a leveled b.p. (i.e. edges leaving nodes at level j enter nodes at level $j+1$) with the further restriction that for each level j there is a fixed input x_i that is associated with all out edges at that level (i.e. all labels are restricted to $x_i = d$ for some d). Bounded width b.p. (see for example [3], [7]) can be viewed as a special case of oblivious b.p. since such programs can be easily converted to be oblivious. At present, the best lower bound for oblivious b.p. [5] is that there is an explicitly defined class of Boolean functions $< f_n >$ such that for every sequence $\langle P_n \rangle$ of deterministic oblivious branching programs computing $\langle f_n \rangle$, length$(P_n) = o(n \log^2 n)$ implies width(P_n) and hence size$(P_n) = \exp(n^{\Omega(1)})$. With some modifications, these results extend to nondeterministic oblivious b.p. (see e.g. [15]).

A read-once b.p. [19] has the property that no input variable x_i appears more than once on any consistent computation path. (A path is consistent if for all i and for all $d_1 \neq d_2$, the labels "$x_i = d_1$" and "$x_i = d_2$" do not both appear on the path.) We note that for deterministic read-once b.p. it is equivalent to insist that no input variable x_i appears twice on any path (whether or not consistent) in the branching program. In this latter form, we would choose to refer to this restriction as syntactic read-once.

Such programs are, of course, a severe restriction of linear length programs. With this restriction, several results on program size have been obtained [4], [11], [26], [28], [30]. In particular, it was proved in [30] (see also [28]) that the Boolean function CLIQUE-ONLY$_m$: $\{0,1\}^{\frac{m(m-1)}{2}} \rightarrow \{0,1\}$ requires deterministic read-once branching program size $\exp(\Omega(m))$ where CLIQUE-ONLY$_m(x_{11}, \ldots, x_{m-1,m}) = 1$ iff the input (x_{ij}) represents a graph which is exactly an $\frac{m}{2}$ clique. Note for comparison that there is a 2-b.p. of size $O(n^3)$ computing this function. In [4], a strictly exponential lower bound of $\exp(\Omega(m^2))$ is established for the Boolean function which counts mod 2 the number of triangles in a graph having m nodes.

Our work is motivated by the exposition in [26].

We extend the lower bound from [30] for the CLIQUE-ONLY function so as to hold for nondeterministic read-once and then observe that the complement problem can be solved by a polynomial size nondeterministic read-once program. This shows that both the Savitch [24] (applied to the complement problem) and Immerman-Szelepcsényi [13], [27] constructions necessarily require multiple reads. Our proof is essentially based upon obtaining a lower bound for the nondeterministic communication complexity in a model which strengthens the model introduced by Papadimitriou and Sipser in [21] that allows an arbitrary partition of variables.

A *syntactic read-k-times branching program* (denoted k-b.p. or k-n.b.p. in the nondeterministic case) is a branching program with the property that no input variable x_i appears more than k times on any path in the program.

We show that there exists an explicitly defined (in fact, NC^1-computable) class of functions $g_n : \mathbb{F}_q^{2n} \to \{0,1\}$ such that any sequence of F_q-way k-n.b.p. $\langle P_n \rangle$ which computes $\langle g_n \rangle$ must have size $\exp\left(\Omega\left(\frac{n}{4^k k^3}\right)\right)$. Specifically, if $q \geq 3$ is a power of a prime and if A is a Generalized Fourier Transform matrix over $\mathbb{F}_q$, then $g_n : \mathbb{F}_q^{2n} \to \{0,1\}$ is the function defined by $g_n(X,Y) = 1$ iff $XAY^T = 0$ where $X = (x_1, \ldots, x_n)$ and $Y = (y_1, \ldots, y_n)$. As a corollary of the proof technique, it is easily seen that the same size bound holds for oblivious programs of length $\leq kn$, although this bound is not as good as the bounds obtained in [5]. Like the proof in [5], our result can be understood in terms of a multi-party communication game but in our case each party has access to only a small fraction of the inputs rather than to all but a small fraction of the inputs. Our result immediately yields a corresponding result for the Boolean function $f_n : \{0,1\}^{O(n)} \to \{0,1\}$ which is derived from g_n by encoding the inputs of $\mathbb{F}_q$.

2 Decomposing Nondeterministic Syntactic Read-k-Times Branching Programs

An R-way switching-and-rectifier network is a tuple $\langle G, s, t, \mu \rangle$ where G is a directed multigraph (W, E) with two distinguished vertices s, t and μ is a *labelling function* which associates with *some* edges $e \in E$ their *labels* $\mu(e)$ of the form "$x_i = d$" where x_i is a variable $(1 \leq i \leq n)$ and $d \in R$. Edges which do not receive any label are called *free*. The network

$\langle G, s, t, \mu \rangle$ computes the function $f : R^n \to \{0,1\}$ defined as follows: for each $u \in R^n$ we let $f(u) = 1$ iff there exists at least one (directed) s-t path (called an *accepting* path for u) such that all labels along this path are consistent with u. The *size* of the network $\langle G, s, t, \mu \rangle$ is the total number of *labeled* edges in G.

Without loss of generality, in what follows we will consider only nondeterministic branching programs (abbreviated n.b.p.) which we will define as those switching-and-rectifier networks $\langle G, s, t, \mu \rangle$ for which the underlying digraph G is acyclic.[1]

A nondeterministic branching program $\langle G, s, t, \mu \rangle$ is *syntactic read-k-times* if and only if for each $1 \le i \le n$ and for each s-t path p, labels of the form "$x_i = d$" appear along p at most k times. Let NBP_k denote the corresponding size complexity measure.

Theorem 1 *Let $f : R^n \to \{0,1\}$ be a function in n variables; let k, a be positive integers. Let $T = (2NBP_k(f))^{2ka}$. Then f can be represented in the form*

$$f = \bigvee_{i=1}^{T} \bigwedge_{j=1}^{ka} f_{ij}(X_{ij}) \tag{0}$$

where f_{ij} is a function depending only on variables from $X_{ij} \subseteq \{x_1, \ldots, x_n\}$, $|X_{ij}| \le \lfloor n/a \rfloor$ and for any $1 \le i \le T$, each variable belongs to at most k of the sets $\{X_{i,1}, \ldots, X_{i,ka}\}$.

Proof. Let S be a syntactic read-k-times nondeterministic branching program of size $NBP_k(f)$ computing f. First we show that without loss of generality we may assume that the overall number of edges in S (including free edges) is at most $(2NBP_k(f))^2$.

Indeed, if S has a vertex w incident only to free edges

$$< w'_1, w >, < w'_2, w >, \ldots, < w'_b, w >; < w, w''_1 >, \ldots, < w, w''_c > \tag{1}$$

then we can remove w from the program S and replace the set of free edges (1) by bc free edges $\{< w'_i, w''_j > \mid 1 \le i \le b,\ 1 \le j \le c\}$. The resulting program has the same size, computes the same function f and retains the read-k-times property. Repeating this

[1] There is a well known construction for converting an arbitrary switching-and-rectifier network to one with an acyclic digraph G. We also note that our definition of a n.b.p. is somewhat nonstandard but equivalent to the model which introduces nondeterminism into branching programs by allowing "guessing nodes".

process, we eventually obtain a program in which each vertex is incident to a labeled edge. This implies that this program contains at most $2NBP_k(f)$ vertices and hence at most $NBP_k(f) + 2NBP_k(f)[2NBP_k(f) - 1] < (2NBP_k(f))^2$ edges.

Now, for each pair of vertices w, w' denote by $X(w, w')$ the set of all variables which appear in labels on all possible paths from w to w'. By $f_{w,w'}$ we denote the function computed by the program $\langle G, w, w', \mu\rangle$; that is by the program which is obtained after moving the origin s to w and the sink t to w'. Clearly, $f_{w,w'}$ depends only on the variables $X(w, w')$. We call a sequence $e_1, e_2, \ldots, e_\ell$ (say $e_i = < w_i, w_i' >$) of edges a *trace* if and only if the following are true:

a) for each j, $1 \le j \le \ell + 1$, $|X(w'_{j-1}, w_j)| < n/a$,

b) for each j, $1 \le j \le \ell$, $|X(w'_{j-1}, w'_j)| \ge n/a$,

where we set $w'_0 = s$ and $w_{\ell+1} = t$.

It is easy to see that any s-t path p contains a (uniquely determined!) trace $(e_1, \ldots, e_\ell)$ where edges $e_1, \ldots, e_\ell$ appear along p in this prescribed order. Now we define the following function f^*:

$$f^* = \bigvee_{\text{trace } (e_1,\ldots,e_\ell)} \bigwedge_{j=1}^{\ell+1} (f_{w'_{j-1} w_j} \wedge \mu(e_j)), \text{ where } \mu(e_{\ell+1}) = 1. \tag{2}$$

This function expresses the fact that there exists at least one trace $(e_1, \ldots, e_\ell)$ and at least one accepting path p for the input being considered such that p contains edges $e_1, \ldots, e_\ell$ (which must appear in this order). Hence, by the remark above, $f^* = f$ and we only have to check that the representation (2) has the desired form (0).

Indeed, the function $f_{w'_{j-1} w_j} \wedge \mu(e_j)$ depends only on variables $X(w'_{j-1}, w_j) \cup \{\mu(e_j)\}$ and we take this set of variables as the corresponding X_{ij} in (0). Then $|X_{ij}| \le \lfloor n/a \rfloor$ follows from the part a) of the definition of a trace. If some variable x_α belonged to more than k sets among $X(w'_{j-1}, w'_j)$ $(1 \le j \le \ell + 1)$ then we could replace in p corresponding subpaths going from w'_{j-1} to w'_j by subpaths containing x_α. This would result in an s-t path along which x_α occurs more than k times which contradicts the read-k-times restriction. Hence each variable belongs to at most k sets among $X(w'_{j-1}, w'_j)$. The same is certainly true for $X(w'_{j-1}, w_j) \cup \{\mu(e_j)\}$ since $X(w'_{j-1}, w_j) \cup \{\mu(e_j)\} \subseteq X(w'_{j-1}, w'_j)$.

Part b) of the definition of trace implies $\sum_{j=1}^{\ell+1} |X(w'_{j-1}, w'_j)| \geq \frac{n\ell}{a} + |X(w'_\ell, t)|$. On the other hand, since each variable contributes to the left-hand part at most k times, $\sum_{j=1}^{\ell+1} |X(w'_{j-1}, w'_j)| \leq kn$. Hence $\ell \leq ka$ and the equality is possible only if $X(w'_\ell, t) = \emptyset$. But the latter implies that $f_{w'_\ell t}$ is a constant so in that case we can drop in (2) either the $(\ell+1)$'st conjunctive term or the whole term corresponding to our trace. We see that in any case each conjunction in (2) contains at most ka nontrivial terms. The fact $\ell \leq ka$ also implies that the total number of traces does not exceed T, which finishes the proof of the theorem. □

3 A Lower Bound for Read-Once Nondeterministic Branching Programs

We first apply Theorem 1 to the case $k = 1$ (i.e. read-once programs).

Corollary 1 *Let f be a Boolean function in n variables, n even. Then f can be represented in the form*

$$f = \bigvee_{i=1}^{T} [f_{i1}(X_{i1}) \wedge f_{i2}(X_{i2})] \tag{3}$$

where $\{X_{i1}, X_{i2}\}$ is a partition of $\{x_1, \ldots, x_n\}$ into two groups of equal size and $T \leq NBP_1(f)^{O(1)}$.

Proof. Set $R = \{0, 1\}$ and $a = 2$ in Theorem 1. If $X_{i1} \cup X_{i2}$ for some i misses certain variables then distribute them among X_{i1}, X_{i2} in an arbitrary way to make their cardinalities equal. □

Note that if the partition $\{X_{i1}, X_{i2}\}$ in (3) is independent of i, then $\log_2$ of the minimal possible T is equal to the nondeterministic communication complexity in the model which allows an arbitrary partition of inputs [21]. This implies that the lower bounds on T proved below also hold in the Papadimitriou-Sipser model.

Let $\text{CLIQUE}_{m,s}$ $\left[\text{respectively, CLIQUE-ONLY}_{m,s}\right]$ be the Boolean function in $\frac{m(m-1)}{2}$ variables which outputs 1 at an input $(x_{11}, \ldots, x_{m-1,m})$ iff the corresponding graph contains an s-clique [respectively, is exactly an s-clique].

Theorem 2 a) $NBP_1(\text{CLIQUE}_{m,s}) \geq \exp(\Omega(\min(s, m-s)))$
b) $NBP_1(\text{CLIQUE-ONLY}_{m,s}) \geq \exp(\Omega(\min(s, m-s)))$.

Proof. Let f be either $\text{CLIQUE}_{m,s}$ or $\text{CLIQUE-ONLY}_{m,s}$. We may assume without loss of generality that the number of variables $n = \frac{m(m-1)}{2}$ is even. Fix a representation of the form (3) for f. We will call functions of the form $f_{i1}(X_{i1}) \wedge f_{i2}(X_{i2})$ ($\{X_{i1}, X_{i2}\}$ is a partition into two groups of size $n/2$ each) *elementary rectangles.* Let u be a random variable uniformly distributed over the set of all ones of the $\text{CLIQUE-ONLY}_{m,s}$ function. Clearly, Theorem 2 (for either of the two functions) would follow from the following lemma (note that in the case $s \geq m/3$, $\text{CLIQUE}_{\frac{3}{2}(m-s), \frac{1}{2}(m-s)}$ can be easily reduced to $\text{CLIQUE}_{m,s}$ and we may apply Lemma 1 with $m := \frac{3}{2}(m-s)$ and $s := \frac{1}{2}(m-s)$).

Lemma 1 *Let $s \leq m/3$ and let R be an elementary rectangle such that $R \leq CLIQUE_{m,s}$. Then $\mathbb{P}[R(u) = 1] \leq \exp(-\Omega(s))$.*

Proof of Lemma 1. Let R be as above; $R = f_1(X_1) \wedge f_2(X_2)$ where $\{X_1, X_2\}$ is a partition of all edges into two equal parts. Let U_ν be the set of all assignments of X_ν which make f_ν equal 1. Let E be the bipartite graph on $U_1 \times U_2$ defined by $(u_1, u_2) \in E$ iff $\text{CLIQUE-ONLY}_{m,s}(u_1, u_2) = 1$. We are going to prove first that E is a star.

Otherwise E contains two edges (u_1, u_2) and (u_1', u_2') where $u_1 \neq u_1'$ and $u_2 \neq u_2'$. For any assignment $u_\nu \in U_\nu$ (think of u_ν as a set of edges in the input graph) denote by $s(u_\nu)$ the number of vertices of the input graph incident to at least one edge in u_ν. Since (u_1, u_2) and (u_1', u_2') represent s-cliques, all four values $s(u_1), s(u_2), s(u_1'), s(u_2')$ are at most s. Now, either $s(u_1) = s(u_1') = s$ or $s(u_2) = s(u_2') = s$. Otherwise we would have two assignments $u_\nu'' \in U_\nu$ ($\nu = 1, 2$) with $s(u_1'') \leq s-1$, $s(u_2'') \leq s-1$. That would imply that (u_1'', u_2'') can not contain a clique of size s which would contradict the assumption $R \leq \text{CLIQUE}_{m,s}$.

Without loss of generality we may assume $s(u_1) = s(u_1') = s$. Also we may assume that $|u_2'| \leq |u_2|$ where $|u|$ is the number of ones in the assignment u. Now look at (u_1, u_2') (considered as an assignment of X). On the one hand, since (u_1, u_2') must contain an

s-clique and $|(u_1, u_2')| \leq |(u_1, u_2)| = \frac{s(s-1)}{2}$, it follows that $(u_1, u_2') \in E$. Since $u_2' \neq u_2$, this s-clique is different from (u_1, u_2). But this is impossible because $s(u_1) = s$ implies that there is at most one s-clique containing u_1. This contradiction shows that E does not contain any 2-matching and hence is a star.

So, we may assume without loss of generality that E is formed by $(u_1, u_2^1), (u_1, u_2^2), \ldots, (u_1, u_2^h)$ for some h. Let V be the set of vertices incident to edges from u_1 and let $|V| = s(u_1) = a$. For any i, $1 \leq i \leq h$, (u_1, u_2^i) is an s-clique on a set of the form $V \dot\cup V_i$, $|V_i| = s - a$. In particular, all edges from the clique on V_i belong to u_2^i and hence to X_2. Since $|X_2| = \frac{m(m-1)}{4}$ we may apply the Lemma of Appendix 1 (with $n := m$, $e := \frac{m(m-1)}{4}$ and $s := s - a$) and conclude that $h \leq m \cdot \binom{\left\lfloor\sqrt{\frac{m(m-1)}{2}}\right\rfloor}{s-a-1} \leq m \cdot \binom{\left\lfloor\sqrt{\frac{m(m-1)}{2}}\right\rfloor}{s-1}$ since $s \leq m/3$. So, $\mathbb{P}[R(u) = 1] = \frac{h}{\binom{m}{s}} \leq \exp(-\Omega(s))$. This completes the proofs of Lemma 1 and Theorem 2. $\square$

Theorem 3 $NBP_1(\neg\text{CLIQUE-ONLY}_{m,s}) \leq O(m^4)$.

Proof. Follows immediately from the following description of the function $\neg$CLIQUE-ONLY$_{m,s}$:

a graph G is not an s-clique if and only if at least one of the following is true:

a) there are two edges (v_1, v_2) and (v_3, v_4) in G such that $v_1 \neq v_3$ and (v_1, v_3) is not an edge;

b) there exists at least one vertex whose degree differs from both 0 and $s - 1$;

c) G is empty.

Now we only have to note that the disjunction of functions computable by read-once branching programs is computable by the read-once *nondeterministic* branching program obtained by placing the original programs in parallel. $\square$

4 A Lower Bound for Nondeterministic Syntactic Read-k-Times Branching Programs

Let A be an $n \times n$ matrix over a finite field $\mathbb{F}_q$. We consider the function $g_A : \mathbb{F}_q^{2n} \to \{0,1\}$ defined by $g_A(X,Y) = 1$ iff $X^T AY = 0$ (X and Y are n-column vectors over $\mathbb{F}_q$).

We denote by $\alpha_A(s)$ the minimal possible rank of a minor of the matrix A with at least s entries.

Theorem 4 *Let A be an $n \times n$ matrix over $\mathbb{F}_q$ (q is a constant). Let k be an arbitrary integer and*

$$s = \frac{n^2}{2 \cdot 4^k}.$$

Then

$$NBP_k(g_A) \geq \exp(\Omega(\alpha_A(s)/k^2)).$$

Proof.

Let $a = 4k$, $T = (2NBP_k(g_A))^{8k^2}$. Applying Theorem 1 to g_A we represent it in the form

$$g_A = \bigvee_{i=1}^{T} \bigwedge_{j=1}^{4k^2} f_{ij}(X_{ij}, Y_{ij})$$

where $|X_{ij}| + |Y_{ij}| \leq \lfloor n/2k \rfloor$ and each variable belongs to at most k sets among X_{ij}, Y_{ij} for each fixed i. Again, as in the proof of Theorem 2, we call each function of the form $\bigwedge_{j=1}^{4k^2} f_j(X_j, Y_j)$ (with $|X_j| + |Y_j| \leq \lfloor n/2k \rfloor$ and each variable belonging to at most k sets among X_j, Y_j) an *elementary rectangle*. As in the proof of Theorem 2, we only have to show that for each elementary rectangle R such that $R \leq g_A$, $\mathbb{P}[R(u) = 1] \leq \exp(-\Omega(\alpha_A(s)))$ where again u is the uniform distribution on the set of all 1's of g_A. Let (ϕ, ψ) denote the uniform distribution on $\mathbb{F}_q^{2n}$. Since $\mathbb{P}[g_A(\phi,\psi) = 1] \geq 1/q$, we may replace u by (ϕ, ψ) and prove that

$$\mathbb{P}[R(\phi,\psi) = 1] \leq \exp(-\Omega(\alpha_A(s))). \tag{4}$$

We call a pair of variables (x, y) *good* iff there is no $j \in \{1, \ldots, 4k^2\}$ for which $x \in X_j$, $y \in Y_j$ and *bad* otherwise. The total number of bad pairs does not exceed $4k^2 \cdot \frac{1}{4}\lfloor \frac{n}{2k} \rfloor^2$.

(Note that for a given j the number of bad pairs is maximized when $\#X_j = \#Y_j = \frac{1}{2}\lfloor\frac{n}{2k}\rfloor$). Hence there are at least $\frac{n^2}{2}$ good pairs (without loss of generality, we assume $\lfloor\frac{n}{2k}\rfloor \le \sqrt{2}\frac{n}{2k}$).

Now, consider a uniformly distributed random coloring $\chi : \{1,\dots,4k^2\} \to \{0,1\}$. We associate with it the following sets of variables: $X^0 = \cup\{X_j \mid \chi(j) = 0\}$, $Y^1 = \cup\{Y_j \mid \chi(j) = 1\}$. For each good pair (x,y) the events $x \notin X^0$, $y \notin Y^1$ are independent and hence $\mathbb{P}[x \notin X^0, y \notin Y^1 \mid (x,y) \text{ is a good pair}] \ge 4^{-k}$. Here we use the property that x (respectively y) occurs in at most k of the sets X_j (respectively Y_j). This implies $\mathbb{E}[\#\{(x,y) \mid x \notin X^0, y \notin Y^1\}] \ge \frac{n^2}{2\cdot 4^k} = s$. Fix an arbitrary χ for which $\#\{(x,y) \mid x \notin X^0, y \notin Y^1\} \ge s$.

We know that for each j either $X_j \subseteq X^0$ or $Y_j \subseteq Y^1$. Therefore our rectangle R can be represented in the following simplified form: $R = f^0(X^0, Y) \wedge f^1(X, Y^1)$. We prove that even after an arbitrary assignment ρ to variables from $X^0 \cup Y^1$ we still have

$$\mathbb{P}[R(\phi,\psi) = 1 \mid (\phi,\psi \text{ satisfies } \rho)] \le \exp(-\Omega(\alpha_A(s))).$$

Let A' be the submatrix of A corresponding to rows $X' = \text{co-}X^0$ and columns $Y' = \text{co-}Y^1$. Then $\rho(g_A) = 1$ if and only if $(X')^T A' Y' + L(X', Y') = 0$ where $L(X', Y')$ is a linear affine function. By definition of $\alpha_A(s)$, we know that rank $(A') \ge \alpha_A(s)$. Let $\#X' = t$ and $\#Y' = u$ and note that $t \cdot u \ge s$. We let $\tilde{f}^1(X') = \rho(f^1)$, $\tilde{f}^2(Y') = \rho(f^0)$, $\Delta_1 = \{v \in F_q^t \mid \tilde{f}^1(v) = 1\}$ and $\Delta_2 = \{w \in F_q^u \mid \tilde{f}^2(w) = 1\}$. In more combinatorial terms, the goal is to show that $\#\Delta_1 \cdot \#\Delta_2 < q^{t+u} \cdot \exp(-\Omega(\alpha_A(s)))$, knowing that there exist $z_1 \in F_q^t$, $z_2 \in F_q^u$, $c \in F_q$ such that

(*) for all $v \in \Delta_1,\ w \in \Delta_2\ \ v^T A' w + v^T \cdot z_1 + z_2^T \cdot w + c = 0.$

Now fix any $v_0 \in \Delta_1$, $w_0 \in \Delta_2$ and let $\tilde{\Delta}_1 = \{v - v_0 \mid v \in \Delta_1\}$, $\tilde{\Delta}_2 = \{w - w_0 \mid w \in \Delta_2\}$. From (*), we obtain

(**) for all $\tilde{v} \in \tilde{\Delta}_1,\ w \in \Delta_2\ \ \tilde{v}^T A' w + \tilde{v}^T z_1 = 0$

and then

(***) for all $\tilde{v} \in \tilde{\Delta}_1,\ \tilde{w} \in \tilde{\Delta}_2\ \ \tilde{v}^T A' \tilde{w} = 0.$

Letting V be equal to the vector space generated by $\tilde{\Delta}_1 \subseteq \mathbb{F}_q^t$ and W equal to the vector space generated by $\tilde{\Delta}_2 \subseteq F_q^u$, we then have

(****) for all $v \in V,\ w \in W \quad v^T A' w = 0.$

Since $\#\tilde{\Delta}_1 = \#\Delta_1 \leq q^{\dim(V)}$ and $\#\tilde{\Delta}_2 = \#\Delta_2 \leq q^{\dim(W)}$, it is sufficient to show $\dim(V) + \dim(W) \leq u + t - \alpha_A(s)$. Let $H = \{v^T A' \mid v \in V\} \subseteq F_q^u$. Then $\dim(H) \geq \text{rank }(A') - (t - \dim(V)) \geq \dim(V) + \alpha_A(s) - t$. Since H and W are orthogonal, $\dim(H) + \dim(W) \leq u$ which implies the desired bound $\dim(V) + \dim(W) \leq u + t - \alpha_A(s)$. This completes the proof of Theorem 4. □

5 Matrices Whose Submatrices Have Large Rank

In this section we are going to look at Generalized Fourier Transform matrices (GFT for short) and prove that every sufficiently large submatrix of a GFT matrix has large rank (by a submatrix we mean throughout any matrix which can be obtained from the original one by deleting arbitrary rows and columns).

There are examples of explicit matrices over $\mathbb{Q}$ such that all minors (i.e. square submatrices) of all sizes are nonsingular. However it is not hard to see that over a finite field any $n \times n$ matrix has an $\Omega(\log n) \times \Omega(\log n)$ submatrix of rank at most 1.

Proving good lower bounds for the minimal rank of a $u \times t$ submatrix of an explicit $n \times n$ matrix over a finite field is an interesting combinatorial problem in its own right and has applications to complexity questions other than those discussed in this paper. For example such bounds can be used in lower bound proofs for constant depth circuits computing an explicitly given linear transformation over F (see [22]).

For an abelian group G and a splitting field F of G (see Appendix 2) we define the Generalized Fourier Transform matrix A that corresponds to G and F as follows.

The rows of A are indexed by elements of G and the columns are indexed by elements of G^*. For $g \in G$ and $\chi \in G^*$ the corresponding entry of A is $\chi(g)$. The columns of A describe the homomorphisms from G to F^* and thus they are linearly independent over F; moreover, since $\#(G) = \#(G^*)$, A is an $n \times n$ invertible matrix (see Appendix 2).

Observe that the rows of A (as well as the columns of A) form a group under componentwise multiplication.

Conversely if A is an invertible matrix over F whose rows form a group G under componentwise multiplication then each column of A defines a different homomorphism from G to F^* and hence A is the GFT matrix corresponding to G and F. In particular the columns of A form a group under componentwise multiplication.

Example. Let $G = (\mathbb{F}_2)^d$ and $F = \mathbb{F}_3$ be the field having 3 elements. Since all elements of G have order 2, $\mathbb{F}_3$ is a splitting field of G and G^* is isomorphic to G.

Choosing a basis for G uniquely defines a dual basis for G^* since -1 is the only primitive square root of unity in $\mathbb{F}_3$.

Now representing the elements of G and G^* in terms of basis elements we can view the corresponding GFT matrix A as a $2^d \times 2^d$ matrix with rows and columns indexed by binary vectors of length d. An entry a_{ij} of A is 1 (in $\mathbb{F}_3$) if the dot product over $\mathbb{F}_2$ of vector i and vector j is 0 and a_{ij} is -1 (in $\mathbb{F}_3$) if the dot product over $\mathbb{F}_2$ of i and j is 1. Such matrices are called *Sylvester matrices.* Clearly A is NC^1-computable.

The ranks of submatrices of a GFT matrix

In what follows let A be a GFT matrix corresponding to an abelian group G and its splitting field F.

For a subset V of G and a subset W of G^* we denote by $A_{V,W}$ the submatrix of A whose rows are indexed by V and whose columns are indexed by W. The following proposition is an elementary property for submatrices of any matrix.

Proposition 1 *For any $V_1, V_2 \subseteq G$ and $W \subseteq G^*$,* rank $(A_{V_1 \cup V_2, W}) \leq$ rank $(A_{V_1,W})$ + rank $(A_{V_2,W})$. $\square$

Together with the next proposition, which is of central importance, these are all the tools that we need for estimating the ranks of submatrices.

Proposition 2 *For any $V \subseteq G$, $W \subseteq G^*$ and any element $g \in G$ we have* rank $(A_{gV,W})$ = rank $(A_{V,W})$.

Proof. Since $a_{gv,w} = a_{v,w}a_{g,w}$, the columns of the matrix $A_{gV,W}$ are obtained from the corresponding columns of $A_{V,W}$ by multiplying them by non-zero constants $a_{g,w}$. Hence columns of $A_{gV,W}$ generate the same linear subspace as those of $A_{V,W}$ and hence the matrices have the same rank. $\square$

Now we are ready to bound from below the rank of $A_{V,W}$. Let $\#(G) = \#(G^*) = n$.

Theorem 5 *If* $V \subseteq G$ *and* $W \subseteq G^*$ *with* $\#(V) = t$ *and* $\#(W) = u$ *then* rank $(A_{V,W}) \geq \frac{ut}{2n\ln(2n/u)}$.

Proof. Let $\ell = \frac{n}{t}\ln\left(\frac{2n}{u}\right)$. Independently choose ℓ random elements $g_1, g_2, \ldots, g_\ell$ of G. Let the set Z consists of those elements of G that do not belong to the union of the sets g_iV where $i \in \{1, 2, \ldots, \ell\}$.

For every $g \in G$, $g \in g_iV$ if and only if $g_i^{-1} \in g^{-1}V$; hence for any fixed $g \in G$, the probability that $g \notin g_iV$ is $1 - \frac{t}{n}$ and these are independent events for all $i \in \{1, 2, \ldots, \ell\}$. Hence $\mathbb{E}[\#Z] = n(1-\frac{t}{n})^\ell$. Thus there is a particular choice of ℓ elements $g_1, g_2, \ldots, g_\ell \in G$ such that $\#(Z) \leq n(1-\frac{t}{n})^\ell = n(1-\frac{t}{n})^{\frac{n}{t}\ln(2n/u)} \leq \frac{u}{2}$.

Now $Z \cup g_1V \cup g_2V \cdots \cup g_\ell V = G$. So by Proposition 1, rank $(A_{Z,W}) +$ rank $(A_{g_1V,W}) + \cdots +$ rank $(A_{g_\ell V,W}) \geq$ rank $(A_{G,W})$.

Since the columns of $A_{G,W}$ are linearly independent we know that rank $(A_{G,W}) = \#(W) = u$. On the other hand rank $(A_{Z,W}) \leq \frac{u}{2}$ since $\#(Z) \leq \frac{u}{2}$, and by Proposition 2 rank $(A_{g_iV,W}) =$ rank $(A_{V,W})$ for any $i \in \{1, 2, \ldots, \ell\}$. We conclude $\ell \cdot$ rank $(A_{V,W}) \geq \frac{u}{2}$ and rank $(A_{V,W}) \geq \frac{ut}{2n\ln(2n/u)}$. $\square$

Since the columns of A also form a group under componentwise multiplication by the dual argument we have rank $(A_{V,W}) \geq \frac{ut}{2n\ln(2n/t)}$.

Combining these two results we obtain the following corollary:

Corollary 2 *If* $A_{V,W}$ *is a* $t \times u$ *submatrix of* A *with area* $s = tu$*, then rank* $(A_{V,W}) \geq \frac{s}{2n(\ln(2n) - \frac{1}{2}\ln(s))}$.

Proof. Either $u \geq \sqrt{s}$ or $t \geq \sqrt{s}$. Apply theorem 5 in the first case and the dual bound in the second case. $\square$

It only remains to combine this Corollary with Theorem 4 of the previous section to establish the following result.

Theorem 6 *Let A be a GFT matrix over $\mathbb{F}_q$, q fixed. Let $g_A : \mathbb{F}_q^{2n} \to \{0,1\}$ be defined by $g_A(X,Y) = 1$ if $X^TAY = 0$. Then $NBP_k(g_A) \geq \exp\left(\Omega\left(\frac{n}{4^k k^3}\right)\right)$.*

Proof. As in Theorem 4, we set $s = \frac{n^2}{2\cdot 4^k}$. Then by the above Corollary

$$\begin{aligned} \alpha_A(s) &\geq \frac{s}{2n((\ln 2n) - \frac{1}{2}\ln(s))} \\ &= \Omega\left(\frac{n}{k \cdot 4^k}\right). \end{aligned}$$

Theorem 4 completes the proof. $\square$

Finally, we can now establish the result claimed in the abstract by showing how each function g_A can be used for constructing a *Boolean* function f with the same lower bound for $NBP_k(f)$ as in Theorem 6. We show how to do this for Sylvester matrices (see Example above).

Let d be an integer and α, β run over $(F_2)^d$. Introduce $n = 4 \cdot 2^d$ variables $x_{\alpha,1}, x_{\alpha,2}, y_{\beta,1}, y_{\beta,2}$ and define

$$f(x,y) = 1 \text{ iff } \sum_{\alpha,\beta} (-1)^{<\alpha,\beta>} (x_{\alpha,1} + x_{\alpha,2})(y_{\beta,1} + y_{\beta,2}) \equiv 0 \bmod 3$$

(The inner product $< \alpha, \beta >$ is taken over $\mathbb{F}_2$).

Theorem 7 $NBP_k(f) \geq \exp\left(\Omega\left(\frac{n}{4^k k^3}\right)\right)$.

Proof. Let A be the Sylvester matrix of order $2^d \times 2^d$. Each k-n.b.p computing f is converted into an 3-way k-n.b.p. computing g_A as follows.

Replace labels "$x_{\alpha,1} = 0$", "$x_{\alpha,1} = 1$", "$x_{\alpha,2} = 0$", "$x_{\alpha,2} = 1$" by "$x_\alpha = 0$ or 1", "$x_\alpha = 2$", "$x_\alpha = 0$" and "$x_\alpha = 1$ or 2" respectively (by "$x_\alpha = 0$ or 1" we mean two parallel edges labeled "$x_\alpha = 0$" and "$x_\alpha = 1$" respectively). Do the same for y-variables. It is easy to see that the resulting 3-way program computes g_A.

Now the proof is completed by applying Theorem 6. $\square$

Appendix 1

Lemma. Each graph with n vertices and e edges contains at most $n \cdot \binom{\lfloor\sqrt{2e}\rfloor}{s-1}$ cliques of size s.

Proof. Order vertices of the graph in such a way $v_1, v_2, \ldots, v_n$ that $d_1 \geq d_2 \geq \cdots \geq d_n$ where d_i is the degree of v_i. Denote the *weight* $w(K)$ of an s-clique K to be the maximum i for which $v_i \in K$. It is sufficient to check that for each w, $1 \leq w \leq n$, there are at most $\binom{\lfloor\sqrt{2e}\rfloor}{s-1}$ cliques of weight w.

It is obvious in the case $w \leq \lfloor\sqrt{2e}\rfloor$ since in that case all cliques K with $w(K) = w$ belong to $\{v_1, v_2, \ldots, v_{\lfloor\sqrt{2e}\rfloor}\}$.

Assume that $w > \lfloor\sqrt{2e}\rfloor$. Since $\sum_{i=1}^{n} d_i = 2e$ and $d_1 \geq \cdots \geq d_w$, we see that $d_w < \lfloor\sqrt{2e}\rfloor$. Which implies that there are at most $\binom{\lfloor\sqrt{2e}\rfloor}{s-1}$ possibilities to bring up an s-clique from the vertex v_w. $\square$

Appendix 2

We introduce some notation and basic results from representation theory.

Let G be a finite abelian group and F be a field. The set of all homomorphisms from G to F^* (also known as the set of linear characters of G in F) forms a group under pointwise multiplication; i.e. $(h_1 \circ h_2)(g) = h_1(g)h_2(g)$. This group is called the dual of G with respect to F and will be denoted by G^*.

Theorem (Dedekind). Any set $\chi_1, \chi_2, \ldots, \chi_n \in G^*$ of distinct characters of G in F is linearly independent over F.

Proof. (See [14], p. 291.)

Since G is abelian we can decompose it into a direct product of cyclic groups. Let $g_1, g_2, \ldots, g_r$ be the generators of these groups. For a character $\chi \in G^*$, $\chi(g_i)$ is an m_i-th root of unity in F where m_i is the order of g_i. Hence the number of distinct characters is at most $\prod_{i=1}^{r} m_i = \#G$. On the other hand every map from the g_i's to the corresponding roots of unity can be extended to a character.

Hence $\#(G) = \#(G^*)$ if and only if F contains a primitive m-th root of unity for every m dividing $\#G$. In this case G is isomorphic to G^*. To describe the above situation we say that F is a splitting field of G.

Acknowledgements. We are indebted to Larry Ruzzo, Laci Babai and an anonymous referee for their helpful comments on an earlier version of this paper.

Added in proof. After this paper was submitted for publication, we learned that similar bounds were independently proved by other authors. Namely, [16] established exponential lower bounds for nondeterministic read-once branching programs computing the function PERFECT-MATCHING-ONLY (in a bipartite graph). [20] proved exponential lower bounds for deterministic syntactic read-k-times branching programs computing the characteristic functions of some linear codes.

References

[1] K. Abrahamson. Generalized string matching. *SIAM Journal on Computing*, 16(6):1039–1051, 1987.

[2] K. Abrahamson. Time-space tradeoffs for algebraic problems on general sequential machines. *JCSS*, 43(2):269-289, Oct 1991.

[3] N. Alon and W. Maass. Meanders and their applications in lower bounds arguments. *JCSS*, 37:118–129, 1988.

[4] L. Babai, P. Hajnal, E. Szemerédi, and G. Turan. A lower bound for read-once branching programs. *JCSS*, 35(2):153–162, Oct 1987.

[5] L. Babai, N. Nisan, and M. Szegedy. Multiparty protocols and logspace-hard pseudorandom sequences. Proceedings of the 21st Annual ACM Symposium on Theory of Computing, pages 1-11, 1989.

[6] L. Babai, P. Pudlák, V. Rödl, and E. Szemerédi. Lower bounds to the complexity of symmetric boolean functions. *Theoretical Computer Science*, 74:313–324, 1990.

[7] D. Barrington and H. Straubing. Superlinear lower bounds for bounded-width branching programs. 6th Structure in Complexity Theory, pages 305–313, 1991 (to appear in JCSS).

[8] P. Beame. A general sequential time-space tradeoff for finding unique elements. *SIAM Journal on Computing*, 20:270–277, 1991.

[9] A. Borodin and S. Cook. A time-space tradeoff for sorting on a general sequential model of computation. *SIAM Journal on Computing*, 11(2):287–297, 1982.

[10] A. Chandra, M. Furst, and R. Lipton. Multiparty protocols. Proceedings of the 15th Annual ACM Symposium on Theory of Computing, pages 94–99, 1983.

[11] P.E. Dunne. Lower bounds on the complexity of 1-time only branching programs. FCT 85, Lecture Notes in CS n. 199, Springer, New York, pages 90-99, 1985.

[12] M.A. Gavrilov. The theory of relay and switching circuits. Nauka, 1950. In Russian.

[13] N. Immerman. Nondeterministic space is closed under complementation. *SIAM J. Comput.*, 17:935–938, 1988.

[14] N. Jacobson. Basic Algebra I. W.H. Freeman and Company, New York, second edition, 1985.

[15] M. Krause, C. Meinel, and S. Waack. Separating complexity classes related to certain input oblivious logarithmic space bounded Turing machines. 4th Structure in Complexity Theory Conference, pages 240–259, 1989.

[16] M. Krause, C. Meinel, and S. Waack. Separating the eraser turing machine classes $L_e, NL_e, co - NL_e$ and P_e. *Theoretical Computer Science*, 86:267–275, 1991.

[17] C. Y. Lee. Representation of switching cirucits by binary-decision programs. *Bell System Technical Journal*, 38:985-999, 1959.

[18] Y. Mansour, N. Nisan, and P. Tiwari. The computational complexity of universal hashing. Proceedings of the 22nd Annual ACM Symposium on the Theory of Computing, pages 235–243, 1990.

[19] W. Masek. A fast algorithm for the string editing problem and decision graph complexity. M.Sc. Thesis, Massachusetts Institute of Technology, Cambridge, May 1976.

[20] E. A. Okolnishnikova. Lower bounds for branching programs computing characteristic functions of binary codes. *Metody discretnogo analiza*, 51:61–83 (in Russian), 1991.

[21] C. H. Papadimitriou and M. Sipser. Communication Complexity. *JCSS*, 28:260–269, 1984.

[22] P. Pudlák. On bounded depth circuits with arbitrary gates. Preprint.

[23] A. Razborov. Lower bounds for deterministic and nondeterministic branching programs. Proceedings of the 8th FCT, Lecture Notes in Computer Science, 529, pages 47–60, New York/Berlin, 1991. Springer-Verlag.

[24] W. J. Savitch. Relationships between nondeterministic and deterministic tape complexities. *J. Comput. System Sci.*, 4:177–192, 1970.

[25] C. Shannon. A symbolic analysis of relay and switching networks. *Transactions of American Institute of Electrical Engineers*, 57:713–723, 1938.

[26] J. Simon and M. Szegedy. Lower bound techniques for read only once branching programs. Preprint, Nov 1990.

[27] R. Szelepcsényi. The method of forcing for nondeterministic automata. *Bull. European Assoc. Theoret. Comput. Sci.*, 33:96–100, 1987.

[28] I. Wegener. *The complexity of Boolean functions.* Wiley-Teubner Series in Comp. Sci., New York – Stuttgart, 1987.

[29] Y. Yesha. Time-space tradeoffs for matrix multiplication and the discrete Fourier transform on any general sequential random-access computer. *SIAM Journal on Computing*, 29:183–197, 1984.

[30] S. Žák. An exponential lower bound for one-time-only branching programs. In *Proceedings of the* 11*th MFCT, Lecture Notes in Computer Science,* 176, pages 562–566, Springer-Verlag, New York/Berlin, 1984.