Meanders, Ramsey Theory and lower bounds for branching programs

Noga Alon[+] [*]　　Wolfgang Maass[++] [**]

[+]Dept. of Math., Tel Aviv University
Ramat Aviv, Tel Aviv, Israel and
Bell Comm. Research, Morristown, N.J. 07960

[++]Dept. of Math., Stat., and Comp. Sci.
University of Illinois at Chicago
Chicago, Illinois 60680

ABSTRACT

A novel technique for obtaining lower bounds for the time versus space complexity of certain functions in a general input oblivious sequential model of computation is developed. This is demonstrated by studying the intrinsic complexity of the following set equality problem SE(n,m): Given a sequence $x_1,x_2,\dots,x_n, y_1,\dots,y_n$ of 2n numbers of m bits each, decide whether the sets $\{x_1,\dots,x_n\}$ and $\{y_1,\dots,y_n\}$ coincide. We show that for any $\log\log n \leq m \leq \frac{1}{2}\log n$ and any $1 \leq s \leq \log n$, any input oblivious sequential computation that solves SE(n,m) using $2^m/s$ space, takes $\Omega(n \cdot s)$ time. This result is sharp for all admissible values of n,m,s and is the first known nontrivial time space tradeoff lower bound (for space $= \omega(\log n)$ of a set recognition problem on such a general model of computation. Our method also supplies lower bounds on the length of arbitrary (not necessarily input oblivious) branching programs for several natural symmetric functions, improving results of Chandra, Furst and Lipton, of Pudlák and of Ajtai et. al. For example we show that for the majority - function any branching program of width w(n) has length $\Omega(n \cdot \log n / w(n) \cdot \log w(n))$, in particular for bounded width we get length $\Omega(n \log n)$ (independently of our work Babai et. al. [BPRS] have simultaneously proved this last result). Our lower bounds for branching programs imply lower bounds on the number of steps that are needed to pebble arbitrary computation graphs for the same computational problems.

To establish our lower bounds we introduce the new concept of a meander that captures superconcentrator-type properties of sequences. We prove lower bounds on the length of meanders via a new Ramsey theoretic lemma that is of interest in its own right. This lemma has other applications, including a tight lower bound on the size of weak superconcentrators of depth 2 that strengthens the known lower bound of Pippenger [Pi]. A surprising new feature of these applications of Ramsey theory in lower bound arguments is the fact that no numbers are required to be unusually large and that several of the resulting superlinear lower bounds are in fact optimal.

1. INTRODUCTION

A branching program that computes a Boolean function f of n Boolean variables $x_1,\dots,x_n$ is a model of computation that generalizes decision trees. The program is a directed acyclic graph, with a special vertex, that has no ingoing edges, denoted by S, and some other special vertices (sinks), that have no outgoing edges. All non-sink vertices are labeled by an input variable and all sinks are labeled 0 or 1. Every non-sink vertex has fan-out two, and the two edges leaving it are labeled 0 or 1. Each assignment of values b_i to the input variables defines a unique computation path from S to one of the sinks, which starts at S, and leaves every non-sink vertex labeled x_i through the edge labeled b_i. The program computes f if $f(b_1,\dots,b_n)$ is the label of the end-vertex of this path, for each possible $b_1,\dots,b_n$.

It is customary to assume, (and for most purposes this can be done without loss of generality), that each vertex has a level, where the level of S is 1, and edges go from each level only to the next one. The width of the program is the maximum number of vertices on a level, and its logarithm corresponds to the space of the computation. The length is the number of levels, and it corresponds to the time of the computation. The size is the total number of vertices in the program.

Branching programs describe a general sequential model of computation, when we identify the vertices in each level with all the possible internal states of the computational device. It is desirable to

[*]Supported in part by Allon Fellowship.

[**]Supported in part by NSF Grant DCR-8504247.

0272-5428/86/0000/0410$01.00 © 1986 IEEE

find functions (in P) that cannot be computed simultaneously in linear time and logarithmic space in such a general model, (i.e., that do not have linear length and polynomial width branching programs). One of the main problems raised by Borodin and Cook, [BC], who proved a time-space tradeoff for sorting in this model, is to obtain such a result for a one output bit function in P. Here we obtain such a result for input oblivious branching programs.

A program is input oblivious if all non-sink vertices in each level have the same label. Notice that every program can be made input oblivious by increasing its length by a factor of its width. In particular every branching program of bounded width can be assumed to be input oblivious (unless constant factors are important).

A slightly more powerful model of computation than a branching program is the R-way model, introduced by Borodin and Cook in [BC]. Here we compute a function f of n variables $x_1,\dots,x_n$, each being a number between 0 and $R-1$. Each non-sink vertex is now labeled by one of the x_i's, and has R outgoing edges labeled by $0,1,\dots,R-1$. The program branches in this vertex according to the value of x_i. Obviously any function of the considered type can be computed by an R-way input oblivious branching program of length n and width R^n.

One can easily show that almost all Boolean functions cannot be computed by a branching program of subexponential size. It is much more difficult to find functions in P (or even in NP) that require nonlinear size. Nechiporuk [Ne] (see also [Sa]) proved an $\Omega(n^2/\log^2 n)$ lower bound for the size of any branching program that computes a certain P-function of n variables. A barely nonlinear lower bound for the size of the branching programs of the majority function was proved using Ramsey theory by Pudlák [Pu]. All the other nontrivial known lower bounds deal with programs that are restricted in some sense. The most popular restriction is the case of bounded width branching programs. The main result of [BDFP] and [Ya] is a superpolynomial lower bound for width-2 branching programs that compute majority. Chandra, Furst and Lipton proved a nonlinear lower bound for the length of any bounded width branching program that computes the symmetric function of n Boolean variables $x_1,\dots,x_n$ whose value is 1 iff $\Sigma x_i = n/2$. Their lower bound is very close to linear, being $\Omega(nW(n))$, where $W(n)$ is the inverse of van der Waerden numbers. Pudlák [Pu] established an $\Omega(n\log\log n/\log\log\log n)$ lower bound for some symmetric functions and Ajtai et. al. [ABHKPRST] obtained an $\Omega(n\log n/\log\log n)$ lower bound for some other, not so natural, symmetric functions (it is easy to see that their arguments also apply to natural symmetric functions). Very recently, this lower bound has been improved in [BPRS] to $\Omega(n\log n)$. Our methods (developed independently of both [ABHKPRST] and [BPRS] enable us to establish a lower bound of $\Omega(n\cdot\log n/w\cdot\log w)$ on the length of arbitrary (not necessarily input oblivious) branching programs of width w for many natural symmetric functions, including all threshold functions T_k, for $n^\delta \le k \le n-n^\delta$, and including the function $\Sigma x_i = n/2$ considered in [CFL] ([BPRS] gives an additional lower bound on the size, but only a matching and for some values of w slightly weaker lower bound on the length of branching programs of unbounded width). In particular for branching programs of bounded width we get a lower bound of $\Omega(n\log n)$ for the previously mentioned symmetric functions. We note that Barrington's recent surprising result [Ba] asserts that the class of functions computable on branching programs of width 5 and polynomial length coincides with the class of functions that have log-depth polynomial size Boolean circuits (i.e., nonuniform NC^1). It seems difficult to obtain any nontrivial lower bounds for any function in this class (that contains, of course, all symmetric functions).

All the previously known results supply no nontrivial lower bound for the length of programs whose width is, say, n^2. Since the logarithm of the width of the program corresponds to the space of the computation this corresponds to space $O(\log n)$ and linear time, which is, of course, not so impressive. As mentioned in [BC] it is desirable to have explicit P-functions whose branching programs have nonlinear length, even when the width is greater than $n^{O(1)}$. Here we obtain nonlinear lower bounds for the length of input oblivious, R-way branching programs for several NC^1-functions of n bits, even when the width is much greater than $n^{O(1)}$.

Our bounds hold for several functions. Here we specify two examples. The first is the set equality function $SE(n,m)$. Its input is a sequence of $2n$ numbers, $x_1,\dots,x_n$, $y_1,\dots,y_n$, each having m bits, where $\log\log n \le m \le \frac{1}{2}\log n$, i.e., each in the range $(0,1,\dots,2^m-1)$. The function is 1 if and only if for each i, $1\le i\le n$ there is some j, $1\le j\le n$ such that $x_i = y_j$ and vice versa, i.e., iff the two sets $X=\{x_1,\dots,x_n\}$ and $Y=\{y_1,\dots,y_n\}$ coincide (without counting multiplicities). In Section 4 we prove that if the width of an input oblivious R-way branching program computing $SE(n,m)$ is at most $2^{2^m/\log n}$, then its length is $\Omega(n \log n)$. This result is optimal. Note that the total number of input bits here for, say, $m = 10 \log\log n$ is $O(n \log\log n)$.

The second example is the sequence equality function $Q(n)$. Its input is a sequence of $2n$ numbers $x_1, \dots, x_n$, $y_1, \dots, y_n$, each being 0, 1 or 2. The value of the function is 1 if and only if the sequence obtained from $x_1, \dots, x_n$ by omitting all occurrences of 2 coincides with the one obtained in the same manner from $y_1, y_2, \dots, y_n$. In Section 4 we show that for any $1 \le s \le \frac{1}{4} \log n$, if the width of an input oblivious 3-way branching program computing $Q(n)$ is at most $2^{n/2^s}$ then its length is $\Omega(n \cdot s)$. Thus the length is superlinear whenever the width is $2^{o(n)}$. This is, in a sense, best possible since obviously any Boolean function of n bits can be computed by an input oblivious branching program of length n and width 2^n.

All lower bounds on the length of input oblivious branching programs imply lower bounds on the number of steps that are needed to pebble *arbitrary* computation graphs for the same problem. We will discuss this conneetion in more detail at the end of Section 4.

Our lower bounds for branching programs follow from a combination of a "cut and paste" argument (= crossing sequence argument) with a new Ramsey theoretic lemma, interesting in its own right, which is proved in Section 2. Our use of Ramsey theory differs in two aspects from previous applications of Ramsey theory in lower bound arguments. According to "common knowledge" every application of Ramsey theory in lower bound arguments requires that certain parameters (typically the size of the considered input numbers) have to be "very large". Therefore the dederived lower bounds tend to be not applicable to ranges of the input parameters that are of practical interest. Furthermore the derived superlinear lower bounds are typically "barely superlinear" and far away from the best known upper bounds. A surprising feature of the applications of the Ramsey-theoretic Lemma 2.1 in the lower bound arguments of Theorems 3.1, 4.1, 4.2 and 4.3 is the fact that the numbers are not required to be "very large" and that the derived lower bounds are optimal or close to optimal (typically in the $n \log n$-range).

Our Ramsey-theoretic lemma has other interesting applications. In Section 3 we define sequences that are useful in lower bound arguments, called *meanders*, and use our lemma to show that each such sequence must be "long". As an application we prove an optimal lower bound for the size of $\log x$ - superconcentrators (s.c.) of depth 2. For a function $g(x)$, a $g(x)$ - s.c. of depth 2 is a directed acyclic graph G with n inputs and n outputs that has the following properties:

(1) Each path from an input to an output has length 2.

(2) For all sets S of k inputs and T of k outputs, there are at least $\lfloor g(k) \rfloor$ vertex disjoint paths in G from S to T.

Pippenger [Pi] showed that every x-s.c. of depth 2 has at least $\Omega(n \log n)$ edges. Our lower bound for the length of meanders enables us to strengthen this and show that every $\log x$ - s.c. of depth 2 has at least $\Omega(n \log n)$ edges. This result is optimal.

2. A RAMSEY THEORETIC LEMMA.

Put $N = \{1, 2, \dots, n\}$ and let $X = (x_1, x_2, \dots, x_r)$ be a sequence of elements of N. For an ordered pair (a,b) of distinct elements of N, we define $v_X(a,b)$ to be the binary vector obtained from X by replacing each a by 0 and each b by 1, and by omitting all $x_i \in N - \{a,b\}$. We call $v_X(a,b)$ the *order type vector* of (a,b) in X.

LEMMA 2.1. Let $X = (x_1, \dots, x_r)$ be a sequence in which each $a \in N$ appears precisely k times $(r = n \cdot k)$, and suppose $N = N_1 \cup N_2$ is a partition of N into two disjoint sets. Then there are two subsets $S \subseteq N_1$, $T \subseteq N_2$, $|S| \ge |N_1|/2^{2k-1}$ and $|T| \ge |N_2|/2^{2k-1}$, such that all the order type vectors $\{v_X(s,t) : s \in S,\ t \in T\}$ are identical.

REMARK 2.2. The assertion of Lemma 2.1 is a Ramsey-theoretic result. It is possible to use some known Ramsey-type results to obtain weaker versions of it. Indeed by considering the complete graph on the elements of N in which the edge (a,b) for $a < b$ is colored by $v_X(a,b)$, one can prove some weak version of Lemma 2.1 by applying the standard Ramsey theorem for graphs (see, e.g., [Bo]). A somewhat better result can be proved using the known results about the problem of Zarankiewicz (see [Bo]). Using these, we can obtain the assertion of Lemma 2.1 for S, T of size $\Omega(\log n/2k)$ (if $|N_1| = |N_2|$). Both results are considerably weaker than the one proved above.

REMARK 2.3. Lemma 2.1 is not far from being the best possible. For every k and n, we can construct a sequence X, in which each $a \in N$ appears precisely k times, with no two disjoint sets S, T of size bigger than $\lceil n/2^{k/2} \rceil$ that satisfy the assertion of the lemma. Indeed, put

$\ell =\ k/2$. For each $1 \leq i \leq \ell$, let $N_{i0}(N_{i1})$ be the seauence of all elements of N whose i-th least significant bit is 0 (1, respectively), ordered in an increasing order. Let X be the concatenation of the following 2ℓ permutations of N: $(N_{i0}N_{i1})$ for $i = 1,\dots,\ell$ and $(N_{i1}N_{i0})$ for $i = 1,\dots,\ell$. We claim that if $S \subset N$ and there is even a single $t \in N - S$ such that all vectors $\{v_x(s,t) : s \in S\}$ are identical, then $|S| \leq \lceil n/2^{\ell} \rceil = \lceil n/2^{\ k/2}\ \rceil$. Indeed, otherwise, there are $s_0, s_1 \in S$ which differ in the i-th coordinate for some $1 \leq i \leq \ell$ and one can easily check that $v_x(s_0,t) \neq v_x(s_1,t)$.

<u>PROOF OF LEMMA 2.1</u>: For $1 \leq p \leq k$, $1 \leq q \leq k$, and an ordered pair (a,b) of distinct elements of N, let $v(a_p,b_q)$ be the subsequence of $v_x(a,b)$ consisting of the initial p-zeros and initial q 1's in $v_x(a,b)$. Thus $v(a_p,b_q)$ is the order type vector of (a,b) in the sequence obtained from X by omitting every occurrence of a besides the first p, and every occurrence of b besides the first q.

We claim that there are two sets $S^2 \subset N_1$, $T^2 \subset N_2$, with $|S^2| \geq |N_1|/2$, $|T^2| \geq |N_2|/2$ such that all vectors $\{v(s_1,t_1) : s \in S^2, t \in T^2\}$ are identical and are either all 01 or 10. Indeed, either half of the elements of N_1 precede half of those of N_2, or vice versa.

Suppose now, that p,q are some numbers satisfying $1 \leq p$, $q \leq k$ and that we have already defined two subsets $S^{p+q} \subset N_1$ and $T^{p+q} \subset N_2$ satisfying $|S^{p+q}| \geq |N_1|/2^{p+q-1}$, $|T^{p+q}| \geq |N_2|/2^{p+q-1}$, such that all vectors $\{v(s_p,t_q) : s \in S^{p+q}, t \in T^{p+q}\}$ are identical and their last two coordinates are distinct. Assume, without loss of generality, that each such $v(s_p,t_q)$ ends with a 1. If $p = k$, we are done, since for each $s \in S^{p+q}$, $t \in T^{p+q}$, $v_x(s,t)$ is just $v(s_p,t_q)$ followed by $k - q$ 1's. If $p < k$ we claim that there are $S^{p+q+1} \subseteq S^{p+q}$ and $T^{p+q+1} \subseteq T^{p+q}$, satisfying $|S^{p+q+1}| \geq |S^{p+q}|/2$ and $|T^{p+q+1}| \geq |T^{p+q}|/2$ such that all vectors $\{v(s_{p+1},t_q) : s \in S^{p+q+1}, t \in T^{p+q+1}\}$ are identical, and their last two coordinates are distinct. Indeed, put $I = \{i : x_i$ is the $(p+1)$ occurrence of some $s \in S^{p+q}\}$ and $J = \{j : x_j$ is the q occurrence of some $t \in T^{p+q}\}$. Clearly $|I| = |S^{p+q}|$ and $|J| = |T^{p+q}|$. Let $\bar{i}$ be the $\lceil |I|/2 \rceil$-smallest number in I and let $\bar{j}$ be the $(\lceil |J|/2 \rceil + 1)$-smallest number in J. If $\bar{i} < \bar{j}$, then we define $S^{p+q+1} = \{s \in S^{p+q}$: the $(p+1)$ occurrence of s in X is not after $x_{\bar{i}}\}$, and $T^{p+q+1} = \{t \in T^{p+q}$: the q occurrence of t in X is not before $x_{\bar{j}}\}$. Clearly, in this case, for every $s \in S^{p+q+1}$ and $t \in T^{p+q+1}$, $v(s_{p+1},t_q)$ is equal to the vector obtained from $v(s_p,t_q)$ by replacing its last coordinate (which is 1) by 01. If $\bar{i} \geq \bar{j}$, we define, similarly, $S^{p+q+1} = \{s \in S^{p+q}$: the $(p+1)$ occurrence of s in X is not before $x_{\bar{i}}\}$, and $T^{p+q+1} = \{t \in T^{p+q}$: the q occurrence of t in X is not after $x_{\bar{j}}\}$. In this case, for every $s \in S^{p+q+1}$ and $t \in T^{p+q+1}$, $v(s_{p+1},t_q)$ is $v(s_p,t_q)$ followed by a zero. In both cases, $|S^{p+q+1}| \geq |S^{p+q}|/2$, $|T^{p+q+1}| \geq |T^{p+q}|/2$ and all vectors $\{v(s_{p+1},t_q) : s \in S^{p+q+1}, t \in T^{p+q+1}\}$ are identical and their last two coordinates are distinct. This proves our claim and completes the proof of Lemma 2.1. □

3. <u>MEANDERS AND WEAK SUPERCONCENTRATORS</u>.

For a sequence $M = x_1x_2\dots x_m$ of numbers $x_i \in \{1,2,\dots,n\} \equiv N$ and for disjoint sets $S,T \subseteq N$ we say that an interval $x_ix_{i+1}\dots x_{i+j}$ is a <u>link between S and T</u> if $x_{i+1},\dots,x_{i+j-1} \notin S \cup T$ and $x_i \in S$, $x_{i+j} \in T$ or $x_i \in T$, $x_{i+j} \in S$. The length of M (also written $|M|$) is m. In analogy to the definition of a superconcentrator we say that a sequence M over n numbers is a <u>meander</u> if for any two disjoint sets $S,T \subseteq \{1,\dots,n\}$ with $|S| = |T|$ there are in M at least $|S|$ links between S and T. More generally, for any function $g : N \to R^+$, M is a <u>g(x)-meander</u> (over n numbers) if for any disjoint sets $S,T \subsetneq \{1,2,\dots,n\}$ with $|S| = |T|$ there are in M at least $g(|S|)$ links between S and T. We call M a <u>g(x)-bipartite-meander</u> if this link property is restricted to sets S, T with $S \subseteq \{1,\dots,n/2\}$ and $T \subseteq \{n/2+1,\dots,n\}$.

<u>THEOREM</u> <u>3.1</u>. If M is a $g(x)$-bipartite meander over n numbers of length $n \cdot f$, then $f \geq \frac{1}{8}\, g(n/2^{8f+1})$. Hence, if M is a $g(x)$-meander of length $n \cdot f$ the same inequality holds. In particular, every $\log x$-meander has length $\Omega(n \log n)$ and for every $g(x) \to \infty$, the length of a $g(x)$-meander is superlinear.

PROOF: Let $\tilde{N}$ be the set of all numbers that appear at most $4f$ times in M. Put $N_1 = \tilde{N} \cap \{1,2,\dots,n/2\}$, $N_2 = \tilde{N} \cap \{n/2+1,\dots,n\}$. Clearly $|N_1|$, $|N_2| \geq n/4$. Let Y be the subsequence of M consisting of all occurrences of numbers from $\tilde{N} = N_1 \cup N_2$ in M and let X be a sequence obtained from Y by adding to it in the end, if necessary, elements from $\tilde{N}$ such that each $x \in \tilde{N}$ appears precisely $4f$ times in X. By Lemma 2.1 there are $S \subset N_1$, $T \subset N_2$, $|S| = |T| = n/2^{8f+1}$, such that all the order type vectors $\{v_M(s,t) : s \in S,\ t \in T\}$ are identical. One can easily check that the number of links between S and T, is, at most, $8f$. This implies the result. □

A noteworthy feature of the preceding lower bound proof is the fact that it uses the link property only for sets S,T of one fixed size $(n/2^{8f+1})$. Therefore we can easily derive the following corollary, which provides a handy criterion for lower bound arguments in complexity theory.

COROLLARY 3.2. Assume $s(n)$ is some arbitrary function and M is a sequence over $\{1,\dots,n\}$. In order to prove that $|M| = \Omega(n \cdot s(n))$ it is sufficient to show for some $k \leq n/2^{s(n)}$ that for any two sets $S \subseteq \{1,\dots,n/2\}$ and $T \subseteq \{n/2+1,\dots,n\}$ of size k there are in M at least $s(n)$ links between S and T.

PROOF: We first observe that if $k \leq k'$ and M satisfies the link property of the corollary for sets S, T of size k, then it also satisfies this link property for sets of size k'. Let $|M| = n \cdot f(n)$ for some suitable $f(n)$. Assume $8f(n) + 1 \leq s(n)$ (otherwise we are done). Let $g : \mathbb{N} \to \mathbb{R}^+$ be any function with $g(n/2^{8f(n)+1}) = s(n)$. Since $k \leq n/2^{s(n)} \leq n/2^{8f(n)+1}$ we know by our observation above that M satisfies the weaker link property which was actually needed in the proof of Theorem 3.1. Therefore we have $f(n) \geq \frac{1}{8} g(n/2^{8f(n)+1}) = \frac{1}{8} s(n)$, which contradicts our assumption $8f(n)+1 \leq s(n)$. □

Using probabilistic arguments, we next prove the following result, which shows that Theorem 3.1 is sharp for every function $g(x)$ that satisfies $\Omega(\log x) \leq g(x) \leq O(x \log x)$. For each such g, the bound given by Theorem 3.1 for the length of the meander is $\Omega(n \log n)$.

THEOREM 3.3. For every sufficiently large n, there is an $\frac{1}{7} x \log n$-meander M_n of length $3n\lceil \log n \rceil$. In fact, almost all sequences containing $3\lceil \log n \rceil$ occurrences of each $i \in \{1,\dots,n\}$ are $\frac{1}{7} x \log n$ (and hence also $\frac{1}{7} x \log x$)-meanders.

PROOF: Define a function $g(x) = \frac{1}{7} x \log n$ and let M be a random sequence in which each $i \in \{1,2,\dots,n\} = N$ occurs $3 \cdot \lceil \log n \rceil$ times. We show that the probability that M is a $g(x)$-meander tends to 1 as n tends to infinity.

Fix a number s, $1 \leq s \leq n/2$ and fix two arbitrary disjoint sets $S,T \subseteq N$ with $|S| = |T| = s$. An easy combinatorial argument shows that the probability that M has exactly $2j+1$ links between S and T is precisely

$$\frac{2 \cdot \binom{3s \log n - 1}{j}^2}{\binom{6s \cdot \log n}{3s \cdot \log n}} .$$

Indeed, the denominator here counts the total number of binary sequences consisting of $3s \log n$ 0's and 1's and the numerator counts the number of those sequences with j blocks of 0's and j blocks of 1's. As the links between S and T depend only on the occurrences of elements from $S \cup T$ in M the above expression for the probability of $2j$ links can be given analogously. By a standard estimate, for every $i \leq g(s)/2$

$\binom{3s \log n}{i} \leq \binom{3s \log n}{g(s)/2} 1 \leq (42e)^{g(s)/2}$ and

hence the probability that there are less than $g(s)$ links between S and T can be bounded by

$$\frac{2 \cdot g(s) \cdot (42e)^{g(s)}}{n^{3s}} .$$

(Here we used the trivial estimate $\binom{6s \log n}{3s \log n} \geq n^{3s}$.) Therefore, the probability that there are two disjoint $S,T \subset N$ with $|S| = |T| \leq n/2$ and with less than $g(|S|)$ links between them is

$$\sum_{s=1}^{n/2} \binom{n}{2s}\binom{2s}{s} \cdot \frac{2g(s) \cdot (42e)^{g(s)}}{n^{3s}} \leq$$

$$\leq \sum_{s=1}^{\infty} \left(\frac{en}{2s}\right)^{2s} \cdot 2^{2s} \cdot \frac{2s \log n (42e)^{\frac{1}{7} s \log n}}{7 \cdot n^{3s}} \leq$$

$$\leq \sum_{s=1}^{\infty} \left(\frac{(en)^2 \cdot 4 \cdot s \log n \cdot (42e)^{(\log n)/7}}{(2s)^2\, n^3}\right)^s \leq$$

$$\leq \sum_{s=1}^{\infty} \left(\frac{e^2 \log n \cdot n^{6.81/7}}{n}\right)^s$$

which tends to 0 as n tends to infinity. □

Recall the definition of a $g(x)$-s.c. of depth 2 given in Section 1. The following lemma is due to Pippenger.

LEMMA 3.4. If there is a $g(x)$-superconcentrator G of depth 2 with n inputs, n outputs, and e edges, then there is a $g(x)$-meander M over n numbers of length e.

PROOF: Let $B_1 : I \to N$ and $B_2 : O \to N$ be two fixed bijections from the set of inputs I of G and the set of outputs O of G to $N = \{1,2,\ldots,n\}$. For any interior vertex v (i.e., $v \notin I \cup O$) of G, let I_v be the set of inputs adjacent to v and let O_v be the set of outputs adjacent to v. Let M be the concatenation of the sequences $B_1(I_v)$, $B_2(O_v)$ for all v. One can easily check that if G is a depth 2 $g(x)$-s.c. with e edges then M is a $g(x)$-meander over n numbers of length e. □

Theorem 3.1 and the last lemma imply that any depth-2 $\log x$-s.c. with n inputs and n outputs has $\Omega(n \log n)$ edges. One can easily check that this is optimal (simply take $\log n$ interior vertices, each adjacent to all inputs and all outputs.

4. LOWER BOUNDS FOR BRANCHING PROGRAMS.

Recall the R-way branching programs defined in Section 1. Our lower bounds follow by proving that the labels of the levels in any input oblivious R-way branching program of small width for the considered functions satisfy certain meander-type conditions. The bound follows then from Corollary 3.2. Our first example is the well known set equality function $SE(n,m)$ defined in Section 1.

One can easily check that a RAM with 2^m registers can compute this function in time $O(n)$ (by writing each number x_i in the register whose address is x_i), whereas a RAM with $n^{O(1)}$ registers can solve it in time $O(n \log n)$ (via sorting). $\Omega(n \log n)$ lower bounds for a RAM with $n^{O(1)}$ registers and for algebraic computation trees (for the case $m \gg n$) appear in [Ma] and [Be], respectively. To the best of our knowledge no lower bound exists for the (realistic) case $m < n$. Here we obtain lower bounds for $m \ll n$ on R-way input oblivious branching programs.

THEOREM 4.1. Suppose $\log\log n \le m \le \frac{1}{2}\log n$, $1 \le s \le \log n$ and $R = 2^m$. Then any R-way input oblivious branching program of width $2^{2^m/s}$ computing $SE(n,m)$ has length $\Omega(n \cdot s)$. This bound is sharp, i.e., for all n,m,s in this range there is an R-way input oblivious branching program of width $2^{2^m/s}$ and length $O(n \cdot s)$ computing $SE(n,m)$.

PROOF: The upper bound is straightforward (partition $\{0,\ldots,R-1\}$ into $s+1$ intervals and check separately for each interval which elements of it occur in the input). To prove the lower bound we argue as follows. Let $\tilde{m}$ be the length of an input oblivious R-way branching program B computing $SE(n,m)$, of width $w \le 2^{2^m/s}$. Let M be a sequence of length $\tilde{m}$ over $\{1,2,\ldots,2n\}$ whose i-th element is j if the i-th level vertices of B are labeled x_j, and is $n+j$ if they are labeled y_j. We claim that for any $S \subset \{1,2,\ldots,n\}$ and $T \subset \{n+1,\ldots,2n\}$ with $|S| = |T| = 2^{m-1}$, there are $\Omega(s)$ links between S and T in M. This, together with Corollary 3.2 implies that $\tilde{m} = \Omega(n \cdot s)$ (for $s(n) \le \frac{1}{2}\log n$ we have $2^{m-1} \le n/2^{s(n)}$).

Fix sets S,T as above. Consider inputs $I_A = \langle z_1,\ldots,z_{2n}\rangle \in SE(n,m)$, where $z_i = 0$ for all $i \notin S \cup T$ and $A = \{z_i : i \in S\} = \{z_j : j \in T\}$, where A is a set of cardinality at most $|S| = 2^{m-1}$ of elements from $\{1,\ldots,2^m-1\}$. Let L be the set of links between S and T in M. A standard "cut and paste" argument (= "crossing sequence" argument) implies that for any two inputs I_A and $I_{A'} = \langle z'_1,\ldots,z'_{2n}\rangle$ with $A \ne A'$ there is a link ℓ in L, such that the computation path in B for I_A differs from that of $I_{A'}$ on that level of the branching program B that corresponds to the last element of the link. Otherwise B would also accept an amalgamated input $\tilde{I} = \langle \tilde{z}_1,\ldots,\tilde{z}_{2n}\rangle \notin SE(n,m)$ given by $\tilde{z}_i = z_i$ for $i \le n$ and $\tilde{z}_i = z'_i$ for $i > n$. There are $\sum_{i=0}^{2^{m-1}} \binom{2^m-1}{i} \ge 2^{2^m-2}$ different choices for A and thus $w^{|L|} \ge 2^{2^{m-2}}$. Since $w \le 2^{2^m/s}$ this implies that $|L| = \Omega(s)$. □

Our second example is the sequence equality function $Q(n)$ defined in Section 1.

THEOREM 4.2. Any (3-way) input oblivious branching program of width $2^{n/2^{h(n)}}$ computing $Q(n)$ has length $\Omega(n \cdot h(n))$. In particular, if the width is $2^{o(n)}$ then the length is superlinear.

PROOF: The proof is similar to the previous one. Let B be an input oblivious 3-way branching program for $Q(n)$ of length m and width $w \leq 2^{n/2^h}$. Let M be a sequence of length m over $\{1,2,\ldots,2n\}$ whose i-th element is j if the i-th level vertices of B are labeled x_j and is $n+j$ if they are labeled y_j. Set $s := h/2$ and suppose $S \subseteq \{1,\ldots,n\}$ and $T \subseteq \{n+1,\ldots,2n\}$ satisfy $|S| = |T| = 2n/2^s$. By Corollary 3.2 it is sufficient to show that there are in M at least s links between S and T. To bound the number ℓ of links between S and T one considers inputs $I_A = \langle z_1,\ldots,z_{2n}\rangle$, where $z_i = 2$ for $i \notin S \cup T$ and A is a binary sequence of length $|S|$ which coincides with the two sequences $\langle z_i\rangle_{i \in S}$ and $\langle z_j\rangle_{j \in T}$. The standard crossing sequence argument implies that $w^\ell \geq 2^{|S|} = 2^{2n/2^s}$, i.e., $\ell \geq \frac{2n}{2^s} \cdot \frac{2^h}{n} = 2^{s+1} \geq s$. □

Finally we consider lower bounds for some symmetric functions.

THEOREM 4.3. Let $T_k = T_k(x_1,\ldots,x_n)$ be the Boolean function of n variables whose value is 1 if and only if $\Sigma x_i \geq k$. Fix any constant $\delta > 0$. Then any input oblivious branching program of width w that computes T_k for some k with $n^\delta \leq k \leq n - n^\delta$ has length $\Omega(n \log n/\log w)$.

PROOF: The proof is similar to the two previous ones. For each pair of disjoint subsets S and T of $\{1,\ldots,n\}$ of size n^δ we consider for each $i \leq n^\delta$ an input I_i that has $k - n^\delta$ many 1's on some fixed subset of $\{x_j \mid j \notin S \cup T\}$, i many 1's in $\{x_j \mid j \in S\}$ and $n^\delta - i$ many 1's in $\{x_j \mid j \in T\}$. These inputs will show via a crossing sequence argument that in any input oblivious branching program of width w for T_k the number ℓ of links between S and T satisfies $w^\ell \geq n^\delta$, thus $\ell \geq \delta \cdot \log n / \log w$. Therefore we can apply Corollary 3.2 with $s := \delta \cdot \log n/ \log w$ (we have $n/2^s \geq n/n^\delta \geq n^\delta = |S|$). □

Analogous results for other symmetric functions can be proved similarly. In particular, we get an $\Omega(n \log n / \log w)$ bound for the function f (considered in [CFL]) of n Boolean variables $x_1,\ldots,x_n$ whose value is 1 if $\Sigma x_i = n/2$. It is not too difficult to show that this is sharp. Indeed, for, say $w = \theta(\log n)$ one can compute f in length $O(n \log n / \log\log n)$ by computing Σx_i modulo each prime p satisfying $p \leq C \cdot \log n$ and by using the Chinese Remainder Theorem. Similarly, the above bound for this function can be shown to be sharp for all $\log n \leq w \leq n$.

Finally we would like to point out that our lower bounds on the length of input oblivious branching programs imply lower bounds on the number of steps that are needed to pebble any computation graph for the same problem. Assume G is a computation graph for a function $f(x_1,\ldots,x_n)$ where all arguments and intermediate results that are computed at nodes of G are from $\{0,\ldots,R-1\}$. Then any pebbling of G with p pebbles in T steps defines an input oblivious branching program for f of width R^p and length T.

ACKNOWLEDGEMENT

We would like to thank A. Borodin, N. Pippenger and Gyorgy Turan for fruitful discussions.

REFERENCES

[ABHKPRST] M. Ajtai, L. Babai, P. Hajnal, J. Komlos, P. Pudlák, V. Rodl, E. Szemeredi and Gy. Turan, Two lower bounds for branching programs, Proceedings 18th ACM STOC, (1986), 30-38.

[BPRS] L. Babai, P. Pudlák, V. Rodl and E. Szemeredi, Lower bounds to the complexity of symmetric Boolean functions, preprint.

[Ba] D.A. Barrington, Bounded width polynomial size branching programs recognize exactly those languages in NC^1, Proceedings 18th ACM STOC, (1986), 1-5.

[Be] M. Ben Or, Lower bounds on algebraic computation trees, Proceedings 15th ACM STOC, (1983), 80-86.

[Bo] B. Bollobas, Extremal Graph Theory, Academic Press, 1976.

[Bc] A. Borodin and S. Cook, A time-space tradeoff for sorting on a general sequential model of computation, SIAM J. on Comp. 11 (1982), 287-297.

[BDFP] A. Borodin, D. Dolev, F. Fich and W. Paul, Bounds for width 2 branching programs, Proceedings 15th ACM STOC, (1983), 87-93.

[CFL] A. Chandra, M. Furst and R. Lipton, Multiparty protocols, Proceedings 15th ACM STOC, (1983), 94-99.

[Ma] W. Maass, On the use of inaccessible numbers and order indiscernibles in lower bound arguments for random access machines, J. of Symbolic Logic, to appear.

[Ne] E. Nechiporuk, On a Boolean function, Dokl Akad. Nauk SSSR, 169 No.4 (1966), 765-766.

[Pi] N. Pippenger, Superconcentrators of depth 2, J. Comp. Sys. Sci. 24 (1982), 82-90.

[Pu] P. Pudlák, A lower bound on the complexity of branching programs, Proc. Conf. on the Math. Found. of Computer Science 1984, Springer Lecture Notes in Computer Science 176 (1984), 480-489.

[Sa] J.E. Savage, The Complexity of Computing, (1976).

[Ya] A.C. Yao, Lower bounds by probabilistic arguments, Proceedings 24th IEEE FOCS, (1983), 420-428.