JOURNAL OF COMPUTER AND SYSTEM SCIENCES 37, 118–129 (1988)

Meanders and Their Applications in Lower Bounds Arguments

NOGA ALON*

Department of Mathematics, Sackler Faculty of Exact Sciences, Tel Aviv University, Ramat Aviv, Tel Aviv, Israel and Bell Communications Research, Morristown, New Jersey 07960

AND

WOLFGANG MAASS†

Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, Chicago, Illinois 60680

Received August 1, 1987

The notion of a *meander* is introduced and studied. Roughly speaking, a meander is a sequence of integers (drawn from the set $N=\{1, 2, ..., n\}$) that wanders back and forth between various subsets of N *a lot*. Using Ramsey theoretic proof techniques we obtain sharp lower bounds on the minimum length of meanders that achieve various levels of wandering. We then apply these bounds to improve existing lower bounds on the length of constant width *branching programs* for various symmetric functions. In particular, an $\Omega(n \log n)$ lower bound on the length of any such program for the majority function of n bits is proved. We further obtain optimal time-space trade-offs for certain input oblivious branching programs and establish sharp lower bounds on the size of *weak superconcentrators* of depth 2.

1. MEANDERS

For a sequence of length m $M=x_1x_2\cdots x_m$ of integers $x_i\in\{1, 2, ..., n\}=N$ and for two disjoint sets $S, T\subseteq N$ we say that an interval $x_ix_{i+1}, ..., x_{i+j}$ of M is a *link between S and T* if $x_{i+1}, ..., x_{i+j-1}\notin S\cup T$ and $x_i\in S, x_{i+j}\in T$ or $x_i\in T, x_{i+j}\in S$. Note that any $x_L\notin S\cup T$ in M belongs to at most one link between S and T. We say that M is a *meander* if for any two disjoint sets S, $T, \subset\{1, 2, ..., n\}$ with $|S|=|T|$ there are in M at least $|S|$ links between S and T. More generally, for any function $g: N\to R^+$ we call a sequence $M\in\{1, ..., n\}^*$ a *g-meander* over

* Supported in part by Allon Fellowship and by a Bat Sheva de Rothschild grant.
† Supported in part by NSF Grant DCR-8504247.

0022-0000/88 $3.00

$\{1, ..., n\}$ if for any two disjoint sets S, $T \subseteq \{1, ..., n\}$ with $|S| = |T|$ there are in M at least $g(|S|)$ links between S and T. Let $L_g(n)$ denote the minimum possible length of a g-meander over $\{1, ..., n\}$. Note that a g-meander for $g(x) = x$ is just a meander.

One can easily check that for every nondecreasing function $g: N \to R^+$, $g(n/2) \leqslant L_g(n) \leqslant ng(n)$. The lower bound follows from the fact that there are at least $g(n/2)$ links in any g-meander between $\{1, ..., n/2\}$ and $\{n/2+1, ..., n\}$. The upper bound is a consequence of the fact that a concatenation of $g(n)$ copies of 1, 2, 3, ..., n is a g-meander of length $ng(n)$. Since, obviously, if $g(x) \leqslant \tilde{g}(x)$ for all x then $L_g(n) \leqslant L_{\tilde{g}}(n)$ for all n, we conclude that $L_g(n) = o(n \cdot \log n)$ if $g(x) = o(\log x)$ and $L_g(n) = w(n \cdot \log n)$ if $g(x) = w(x \cdot \log x)$. In this section we prove the somewhat surprising result that $L_g(n) = \Theta(n \cdot \log n)$ for all functions g in between, i.e., for all functions g such that $\Omega(\log x) \leqslant g(x) \leqslant O(x \log x)$ for all x.

In applications one often encounters sequences $M \in \{1, ..., n\}^*$ which satisfy the link property of a g-meander only for sets S, T, $\subseteq \{1, ..., n\}$ with $S \subseteq \{1, ..., n/2\}$ and $T \subseteq \{n/2+, ..., n\}$. We call a sequence M with this slightly weaker property a *g-bipartite-meander.*

Moreover, we will need a lower bound for the length of sequences that have an even weaker link property, namely sequences that satisfy the link property only for sets $S \subseteq \{1, ..., n/2\}$ and $T \subseteq \{n/2+1, ..., n\}$ of one fixed size. The following theorem supplies such a bound.

THEOREM 1.1. *Let M be a sequence of length m over $N = \{1, 2, ..., n\}$. Let s be a positive real number and suppose that there is some positive integer l that satisfies $l \leqslant n/2^s$ such that for any two sets of cardinality l $S \subseteq \{1, 2, ..., n/2\}$ and $T \subseteq \{n/2+1, ..., n\}$ there are in M at least s links between S and T. Then $m \geqslant 1/8n\,(s-9)$.*

In order to prove Theorem 1.1 we need a Ramsey-theoretic lemma. Let $X = x_1, ..., x_r$ be a sequence of elements of N. For an ordered pair (a, b) of distinct elements of N we define the *order type vector* $v_x(a, b)$ to be the binary vector obtained from X by replacing each occurrence of a by 0 and each occurrence of b by 1, and by omitting all other numbers in X.

LEMMA 1.2. *Let $X = x_1, ..., x_r$ 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 nonempty sets. Then there are two subsets $S \subseteq N_1$, $T \subseteq N_2$, $|S| \geqslant |N_1|/2^{2k-1}$ and $|T| \geqslant |N_2|/2^{2k-1}$, such that the set of all the order type vectors $\{v_x(s, t): s \in S, t \in T\}$ contains only one element.*

Lemma 1.2 is proved in the next section. We now show how it implies Theorem 1.1

Proof of Theorem 1.1. Put $f = \lceil m/n \rceil$. If $8f + 1 \geqslant s$ then $m \geqslant 1/8n(s-9)$, as needed. Hence we may assume $8f + 1 \leqslant s$. Let L be the set of numbers in $\{1, ..., n/2\}$

which occur at most $4f$ times in M, and let U be the set of numbers in $\{n/2+1, ..., n\}$ which occur at most $4f$ times in M. Clearly $|L| \geqslant n/4$ and $|U| \geqslant n/4$. Let Y be the subsequence of M consisting of all occurrences of numbers from $L \cup U$ in M and let X be a sequence obtained from Y by adding to it at the end, if necessary, elements from $L \cup U$ so that each number in $L \cup U$ occurs precisely $4f$ times in X. Define $k = 4f$. Since $(n/4)/2^{2k-1} = n/2^{8f+1} \geqslant n/2^s \geqslant l$ we can apply Lemma 1.2 and conclude that there are sets $S \subseteq L$, $T \subseteq U$, with $|S| = |T| = l$ such that all the order type vectors $\{v(a, b) | a \in S \text{ and } b \in T\}$ are identical. One can easily check that the number of links between S and T in X is at most as large as the number of alternations between 0 and 1 in this common order type vector. Therefore it is bounded above by $8f$ (= the length of this order type vector). Hence, the number of links between S and T in M is at most $8f < s$, contradicting the hypothesis of the theorem. Thus $m \geqslant 1/8n(s-9)$, as needed. ∎

COROLLARY 1.3. *For any function g from N to R^+ the minimum length $L_g(n)$ of any g-meander over $\{1, 2, ..., n\}$ satisfies*

$$L_g(n) \geqslant n/8 \left(g\left(\left\lfloor \frac{n}{2^{8L_g(n)/n - 10}} \right\rfloor \right) - 10 \right).$$

In particular, $L_g(n)$ is superlinear in n if $g(x) \to \infty$ and $L_g(n) = \Omega(n \log n)$ if $g(x) \geqslant \Omega(\log x)$. The same lower bounds hold for the length of g-bipartite meanders, as well.

Proof. We prove the bound for g-bipartite meandes. (The proof for g-meanders is analogous.) Let m be the length of such a meander. Define $s = 8m/n + 10$ and $l = \lfloor n/2^s \rfloor$. If $g(l) \geqslant s$ then, by Theorem 1.1, $m \geqslant 1/8n(s-9) > m$, which is impossible. Hence $s > g(l)$, i.e., $m > n/8\,(g(l) - 10) = n/8\,(g(\lfloor n/2^{8m/n - 10} \rfloor) - 10)$. This completes the proof. ∎

Using probabilistic arguments, we next prove the following result, which shows that Corollary 1.3 is sharp for every function $g(x)$ that satisfies $\Omega(\log x) \leqslant g(x) \leqslant O(x \log x)$. For each such g the lower bound given by Corollary 1.3 for the length of the coresponding meander is $\Omega(n \log n)$.

THEOREM 1.4. *For every n there is an $\Omega(x \cdot \log x)$-meander M_n of length $O(n \cdot \log n)$. In fact, for sufficiently large n, almost all sequences containing $3 \cdot \lceil \log n \rceil$ occurrences of each $i \in \{1, ..., n\}$ are g-meanders for $g(x) = 1/7x \cdot \log n$ (and hence also for $g(x) = 1/7x \cdot \log x$).*

Proof. Define a function $g(x) = 1/7x \cdot \log n$ and let M be a random sequence in which each $i \in \{1, 2, ..., n\} = N$ 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. For simplicity we omit all the ceilings and floors.

Fix a number s, $1 \leqslant s \leqslant n/2$ 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

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

This is because the links between S and T depend only on the occurrences of elements from $S \cup T$ in M. Hence, the above probability is just the probability that a random binary sequence of $3s \log n$ 0's and $3s \log n$ 1's will have $j+1$ blocks of 0's and $j+1$ blocks of 1's. The denominator of the last expression counts the total number of binary sequences consisting of $3s \log n$ 0's and 1's. The numerator counts the number of such sequences with $j+1$ blocks of 0's and $j+1$ blocks of 1's. (There are $\binom{3s \log n - 1}{j}$ ways to split the $3s \log n$ 0's into $j+1$ nonempty blocks, $\binom{3s \log n - 1}{j}$ ways to split the 1's, and 2 ways to decide if the first block is a block of 0's or of 1's.) A similar expression for the probability of $2j$ links can be given. By a standard estimate $\binom{a}{b} \leqslant (ea/b)^b$ for all a, b and hence for every $i \leqslant g(s)/2$ $\binom{3s \log n}{i} \leqslant \binom{3s \log n}{g(s)/2} \leqslant (42e)^{g(s)/2}$. Thus 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 $\binom{6s \log n}{3s \log n} \geqslant n^{3s}$.) Therefore, the probability that there are two disjoint S, $T \subset N$ with $|S| = |T| \leqslant n/2$ and with less than $g(|S|)$ links between them is bounded by

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

$$\leqslant \sum_{s=1}^{\infty} \left(\frac{(en)^2 \cdot 4 \cdot s \log n \cdot (42e)^{(\log n)/7}}{(2s)^2 n^3}\right)^s \leqslant \sum_{s=1}^{\infty} \left(\frac{e^2 \log n \cdot n^{6,81/7}}{n}\right)^s$$

which tends to 0 as n tends to infinity. ∎

2. The Proof of the Ramsey Theoretic Lemma

In this section we prove Lemma 1.2 stated in the previous section. For $1 \leqslant p \leqslant k$, $1 \leqslant q \leqslant k$, and an ordered pair (a, b) of distinct elements of N, let $v^{(p,q)}(a, b)$ be the subsequence of $v_x(a, b)$ consisting of the initial p zeros and initial q ones in $v_x(a, b)$. Thus $v^{(p,q)}(a, b)$ 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 $S^{(2)} \subseteq N_1$, $T^{(2)} \subseteq N_2$, with $|S^{(2)}| \geqslant |N_1|/2$, $|T^{(2)}| \geqslant |N_2|/2$ such that all the vectors in the collection $\{v^{(1,1)}(s, t) : s \in S^{(2)},$

$t \in T^{(2)}\}$ are identical and are either all the vector 01 or all the vector 10. This is because either there are half of the elements of N_1 whose first occurrence precedes that of half of those of N_2, or vice versa.

Suppose now (by induction on $p+q$), that p, q are some numbers satisfying $1 \leqslant p, q \leqslant k$ and that we have already defined two subsets $S^{(p+q)} \subset N_1$ and $T^{(p+q)} \subset N_2$ satisfying $|S^{p+q}| \geqslant |N_1|/2^{p+q-1}$, $|T^{(p+q)}| \geqslant |N_2|/2^{p+q-1}$, such that all vectors in the collection $\{v^{(p,q)}(s,t): s \in S^{(p+q)}, t \in T^{(p+q)}\}$ are identical and their last two coordinates are distinct. Assume, without loss of generality, that each such $v^{(p,q)}(s,t)$ 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 $v^{(p,q)}(s,t)$ followed by $k-q$ 1's. If $p<k$ we claim that there are two sets $S^{(p+q+1)} \subseteq S^{(p+q)}$ and $T^{(p+q+1)} \subseteq T^{(p+q)}$ satisfying $|S^{(p+q+1)}| \geqslant |S^{(p+q)}|/2$ and $|T^{(p+q+1)}| \geqslant |T^{(p+q)}|/2$ such that all the vectors in the collection $\{v^{(p+1,q)}(s,t): s \in S^{(p+q+1)}, t \in T^{(p+q+1)}\}$ are identical, and their last two coordinates are distinct.

Indeed, put $I = \{i: x_i$ is the $(p+1)$th occurrence of some $s \in S^{(p+q)}\}$ and $J = \{j: x_j$ is the qth occurrence of some $t \in T^{(p+q)}\}$. Clearly $|I| = |S^{(p+q)}|$ and $|J| = |T^{(p+q)}|$. Let $\bar{\imath}$ be the $\lceil |I|/2 \rceil$-smallest number in I and let $\bar{\jmath}$ be the $(\lceil |J|/2 \rceil + 1)$-smallest number in J. If $\bar{\imath} < \bar{\jmath}$, then we define $S^{(p+q+1)} = \{s \in S^{(p+q)}:$ the $(p+1)$th-occurrence of s in X is not after $x_{\bar{\imath}}\}$, and $T^{(p+q+1)} = \{t \in T^{(p+q)}:$ the qth occurrence of t in X is not before $x_{\bar{\jmath}}\}$. Clearly, in this case, for every $s \in S^{(p+q+1)}$ and $t \in T^{(p+q+1)}$, $v^{(p+1,q)}(s,t)$ is equal to the vector obtained from $v^{(p,q)}(s,t)$ by replacing its last coordinate (which is 1) by 01. If $\bar{\imath} \geqslant \bar{\jmath}$ we define, similarly, $S^{(p+q+1)} = \{s \in S^{(p+q)}:$ the $(p+1)$th occurrence of s in X is not before $x_{\bar{\imath}}\}$ and $T^{(p+q+1)} = \{t \in T^{(p+q)}:$ the qth occurrence of t in X is not after $x_{\bar{\jmath}}\}$. In this case, for every $s \in S^{(p+q+1)}$, and $t \in T^{(p+q+1)}$, $v^{(p+1,q)}(s,t)$ is $v^{(p,q)}(s,t)$ followed by a zero. In both cases $|S^{(p+q+1)}| \geqslant |S^{(p+q)}|/2$. $|T^{(p+q+1)}| \geqslant |T^{(p+q)}|/2$, all the vectors in the collection $\{v^{(p+1,q)}(s,t): s \in S^{(p+q+1)}, t \in T^{(p+q+1)}\}$ are identical and their last two coordinates are distinct. This proves the claim and completes the proof of Lemma 1.2. ∎

Remark 2.1. The assertion of Lemma 1.2 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 1.2 by applying the standard Ramsey theorem for graphs (see, e.g., [5]). A somewhat better result can be proved using the known results about the problem of Zarankiewicz (see [5]). Using these, we can obtain the assertion of Lemma 1.2 for S, T of size $\Omega(\log n/2k)$ (if $|N_1| = |N_2| = n$). Both results are considerably weaker than the one proved above.

Remark 2.2. Lemma 1.2 is not far from being the best possible. For every k and n, we can construct a sequence X, satisfying the hypothesis of the lemma, in which there are 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 $l = k/2$. For each $1 \leqslant i \leqslant l$ let $N_{i0}(N_{i1})$ be the sequence of all elements of $N = \{0, 1, ..., n-1\}$ whose ith 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 $i=1, ..., l$ and $(N_{i1}N_{i0})$ for $i=1, ..., l$. For example, if $N=\{0, 1, 2, 3, 4, 5, 6, 7\}$ and $k=2$ we take $X=0, 2, 4, 6, 1, 3, 5, 7, 1, 3, 5, 7, 0, 2, 4, 6$.

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 $|S| \leqslant \lceil n/2^l \rceil = \lceil n/2^{k/2} \rceil$. Indeed, otherwise, there are $s_0, s_1 \in S$ which differ in the ith coordinate for some $1 \leqslant i \leqslant l$ and one can easily check that $v_X(s_0, t) \neq v_X(s_1, t)$.

3. Weak Superconcentrators of Depth 2

We show in this section that the lower bound on the length of g-meanders (Corollary 1.3) implies corresponding lower bounds on the size (= number of edges) of superconcentrators of depth 2. Moreover, the same bounds hold for the size of networks with weaker connectivity properties which we call *weak* superconcentrators.

An n-network is an acyclic directed graph with n distinguished vetices called inputs and n other distinguished vertices called outputs. For any function $g: N \to R^-$ we call an n-network C_n a *g-superconcentrator of depth* 2 if

(i) each path from an input to an output has length 2, and

(ii) for any set S of inputs and any set T of outputs with $|S| = |T|$ there are at least $g(|S|)$ vertex-disjoint paths from S to T.

A superconcentrator of depth 2 is the special case of a g-superconcentrator of depth 2, where $g(x)=x$ (see [11]). It is thus reasonable to refer to each g-superconcentrator with $g(x)=o(x)$ as a weak superconcentrator.

Pippenger [11] showed that every superconcentrator of depth 2 has $\Omega(n \cdot \log n)$ edges. Our lower bound for the length of g-meanders enables us to strengthen this and show that every log *x-superconcentrator* of depth 2 has $\Omega(n \cdot \log n)$ edges. This lower bound is optimal (simply take log n interior vertices, each adjacent to all inputs and all outputs).

Apparently for $g(x)=o(x)$ no lower bound on the size of g-superconcentrators of depth 2 was previously available (a trivial upper bound is $O(n \cdot g(n))$ as above).

The following lemma is due to Pippenger.

LEMMA 3.1. *Let $g: N \to R^+$ be arbitrary and let G be a g-superconcentrator of depth 2 with n inputs, n outputs, and e edges. Then there is a g-meander M over $\{1, ..., n\}$ with length $(M)=e$.*

Proof. Identify both the inputs and the outputs of G with the set $\{1, ..., n\}$. For any interior vertex v of G let $I_v \subseteq \{1, ..., n\}$ be the set of inputs adjacent to v and let $O_v \subseteq \{1, ..., n\}$ be the set of outputs adjacent to v. Let $K_v \in \{1, ..., n\}^*$ consist of all

numbers in I_v (in any order) followed by all numbers in O_v (in any order). Define M as the concatenation of these sequences K_v for all interior vertices v in G. Clearly length $(M) = e$. We claim that M is a g-meander over $N = \{1, 2, ..., n\}$. Indeed, let S, T be two disjoint subsets of N, $|S| = |T|$. Since G is a g-superconcentrator it contains $r = g(|S|)$ vertex-disjoint paths from S to T. Let $v_1, v_2, ..., v_r$ be the interior vertices of these paths. Clearly each K_{v_i} contains a link between S and T for each $1 \leqslant i \leqslant r$. Hence M is a g-meander, as claimed. ∎

COROLLARY 3.2. *If* $\forall x(g(x) \geqslant \log x)$ *then any g-superconcentrator of depth* 2 *with n inputs and n outputs has* $\Omega(n \cdot \log n)$ *edges. Moreover, for any function g with* $g(x) \to \infty$ *the minimum size of g-superconcentrators of depth* 2 *with n inputs and n outputs is superlinear in n.*

Proof. This follows immediately from Corollary 1.3 and Lemma 3.1. ∎

4. LOWER BOUNDS FOR BRANCHING PROGRAMS

A branching program that computes a Boolean function f of n Boolean variables $x_1, ..., x_n$ is a model of computation that generalizes decision trees. The program is a directed acyclic graph, with a special vertex S, that has no ingoing edges, 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 $f(b_1, ..., b_n)$ is the label of the end-vertex of this path, for each possible $b_1, ..., b_n$.

A generalization of this type of branching program is the R-way model, introduced by Borodin and Cook in [6]. Here we compute a function f of n variables $x_1, ..., x_n$, 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 $0, 1, ..., R-1$. The program branches in this vertex according to the value of x_i.

It is customary to assume, (and for most purposes this can be done without loss of generality although with some loss of power), 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 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, [6], who proved a time-space trade-off for sorting in the R-way 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. Obviously any function of the considered type can be computed by an R-way input oblivious branching program of length n. Also 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). Input oblivious branching programs also arise if one pebbles *arbitrary* computation graphs for a problem (see Remark 4.4).

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 in NP) that require nonlinear size. Nechiporuk [10] (see also [13]) 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 any branching program for the majority function was proved using Ramsey theory by Pudlak [12]. 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 [7, 15] (see also [14]) 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 $x_1, ..., x_n$ whose value is 1 iff $\sum x_i = n/2$ [8]. Their lower bound is very close to linear, being $\Omega(nW(n))$, where $W(n)$ is the inverse of van der Waerden numbers, and it implies a similar lower bound for the majority function. Pudlák [12] established an $\Omega(n \log\log n/\log\log\log n)$ lower bound for some symmetric functions and Ajtai *et al.* [1] obtained an $\Omega(n \log n/\log\log n)$ lower bound for some other symmetric functions.

Very recently, this lower bound has been improved in [2] to $\Omega(n \log n)$. Our methods (developed independently of both [1, 2]) enable us to establish a lower bound of $\Omega(n \log n/\log w)$ on the length of input oblivious branching programs of width w for many symmetric functions, including all threshold functions T_k, for $n^\delta \leqslant k \leqslant n - n^\delta$, and including the function $\sum x_i = n/2$ considered in [8]. Since any branching program can be made input oblivious by increasing its length by a factor of its width, this implies an $\Omega(n \log n/w \cdot \log w)$ lower bound on the length of arbitrary (i.e., not necessarily input oblivious) branching programs of width w for these functions. ([2] 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 [3] 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 nonrivial 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 [6] 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 lower bounds follow by proving that the sequence of labels of the levels in any input oblivious R-way branching program of the considered width has a meander-type property (if it computes correctly the function in question).

Our first example is the well-known set equality function $\mathrm{SE}(n, m)$. Its input is a sequence of $2n$ numbers, $x_1, \ldots, x_n, y_1, \ldots, y_n$, each having m bits, where $\log\log n \leqslant m \leqslant \frac{1}{2}\log n$, i.e., each x_i and y_i is in the range $(0, 1, \ldots, 2^m - 1)$. The function is 1 if and only if for each i, $1 \leqslant i \leqslant n$, there is some j, $1 \leqslant j \leqslant n$, such that $x_i = y_j$ and vice versa, i.e., iff the two sets $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_n\}$ coincide (without counting multiplicities).

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 [9] and [4], 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* $\log\log n \leqslant m \leqslant \frac{1}{2}\log n$, $1 \leqslant s \leqslant \frac{1}{2}\log n$, *and* $R = 2^m$. *Then any R-way input oblivious branching program of width $2^{2^m/s}$ computing* $\mathrm{SE}(n, m)$ *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* $\mathrm{SE}(n, m)$.

Proof. The upper bound is staightforward (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 $\mathrm{SE}(n, m)$, of width $w \leqslant 2^{2^m/s}$. Let M be a sequence of length $\tilde{m}$ over $\{1, 2, \ldots, 2n\}$ whose ith element is j if the ith 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 $T \subseteq \{n+1, \ldots, 2n\}$ with $|S| = |T| = 2^{m-1}$,

there are $\Omega(s)$ links between S and T in M. This, together with Theorem 1.1 implies that $\tilde{m} = \Omega(n \cdot s)$ (for $s(n) \leqslant \frac{1}{2} \log n$ we have $2^{m-1} \leqslant n/2^{s(n)}$).

Fix sets S, T as above. Consider inputs $I_A = \langle z_1, ..., z_{2n} \rangle \in \mathrm{SE}(n, m)$, where $z_l = 0$ for all $l \notin S \cup T$ 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 $\{1, ..., 2^m - 1\}$. 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 $I_{A'} = \langle z'_1, ..., z'_{2n} \rangle$ with $A \neq A'$ 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. This is because otherwise B would also accept an amalgamated input $\tilde{I} = \langle \tilde{z}_1, ..., \tilde{z}_{2n} \rangle \notin \mathrm{SE}(n, m)$ given by $\tilde{z}_i = z_i$ for $i \leqslant n$ and $\tilde{z}_i = z'_i$ for $i > n$. There are $\sum_{i=0}^{2^{m-1}} \binom{2^{m-1}}{i} \geqslant 2^{2^{m-2}}$ different choices for A and thus $w^{|L|} \geqslant 2^{2^{m-2}}$. Since $w \leqslant 2^{2^m/s}$ this implies that $|L| = \Omega(s)$. ∎

Our second example is the sequence equality function $Q(n)$. Its input is a sequence of $2n$ numbers $x_1, ..., x_n, y_1, ..., y_n$, each being 0, 1, or 2. The value of the function is 1 if and only if the sequence obtained from $x_1, ..., x_n$ by omitting all occurrences of 2 coincides with the one obtained in the same manner from $y_1, y_2, ..., y_n$. We show that for any $1 \leqslant s \leqslant \frac{1}{4} \log n$, 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.

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 $w \leqslant 2^{n/2^h}$. Let M be a sequence of length m over $\{1, 2, ..., 2n\}$ whose ith element is j if the ith level vertices of B are labeled x_j and is $n+j$ if they are labeled y_j. Set $s = h/2$ and suppose $S \subseteq \{1, ..., n\}$ and $T \subseteq \{n+1, ..., 2n\}$ satisfy $|S| = |T| = 2n/2^s$. By Theorem 1.1 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 $I_A = \langle z_1, ..., z_{2n} \rangle$, 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 $w^l \geqslant 2^{|S|} = 2^{2n/2^s}$, i.e., $l \geqslant (2n/2^s)(2^h/n) = 2^{s+1} \geqslant s$.

Finally we consider lower bounds for some symmetric functions.

THEOREM 4.3. *Let $T_k = T_k(x_1, ..., x_n)$ be the Boolean function of n variables whose value is 1 if and only if $\sum x_i \geqslant k$. Fix any constant $\frac{1}{2} > \delta > 0$. Then any input oblivious branching program of width w that computes T_k for some k with $n^\delta \leqslant k \leqslant n - n^\delta$ has length $\Omega(\delta n \log n / \log w)$.*

Proof. The proof is similar to the previous two ones. Let $\tilde{m}$ be the length of an input oblivious branching program B of width w that computes T_k. Let M be the sequence of length $\tilde{m}$ over $\{1, 2, ..., n\}$ whose ith element is j if the ith level vertices of B are labeled x_j, Let S and T be two fixed disjoint subsets of $\{1, 2, ..., n\}$, of cardinality n^δ each. Fix a subset V of cardinality $k-n^\delta$, $V \subseteq \{1, ..., n\} \setminus (S \cup T)$. For each i, $0 \leqslant i \leqslant n^\delta$, define an input $I_i = \langle z_1, ..., z_n \rangle \in T_k$ as follows: $z_l = 1$ for each $l \in V$, $z_l = 1$ for the first i members of S and for the first $n^\delta - i$ members of T, and $z_l = 0$ in any other case. Let L be the set of links between S and T in M. A standard crossing sequence argument implies that for any two distinct inputs I_i and $I_j = \langle z'_1, ..., z'_n \rangle$ with, say, $i < j$ there is a link l in L, such that the computation path in B for I_i differs from that of I_j on that level of the branching program B that corresponds to the last element of the link. This is because otherwise B would also accept an amalgamated input $\tilde{I} = \langle \tilde{z}_1, ..., \tilde{z}_n \rangle \notin T_k$ given by $\tilde{z}_l = z_l (= z'_l)$ for $l \notin S \cup T$, $\tilde{z}_l = z_l$ for $l \in S$ and $\tilde{z}_l = z'_l$ for $l \in T$. We thus conclude that $w^{|L|} \geqslant n^\delta$, i.e., $|L| \geqslant \delta \log n / \log w$. We can now apply Theorem 1.1 with $s = \delta \log n / \log w$ (note that $n/2^s \geqslant n/n^\delta \geqslant n^\delta = |S|$) to conclude that $\tilde{m} = \Omega(\delta n \log n / \log w)$, as needed. ∎

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 [8]) of n Boolean variables $x_1, ..., x_n$ whose value is 1 if $\sum x_i = n/2$. 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 $\sum x_i$ modulo each prime p satisfying $p \leqslant 10 \cdot \log n$ and by using the Chinese remainder theorem. Similarly, the above bound for this function can be shown to be sharp for all $\log n \leqslant w \leqslant n$. We do not know if is sharp for fixed w (and suspect it is not).

Remark 4.4. Input oblivious R-way branching programs arise in a natural way if one analyses the pebbling of *arbitrary* computation graphs for a function. Let $f(x_1, x_2, ..., x_n)$ be a function with arguments and value in $\{0, 1, ..., R-1\}$. A *computation graph* G for $f(x_1, ..., x_n)$ is defined as follows. G is a directed, acyclic graph with n special vertices, called sources, labelled $x_1, ..., x_n$, which have no ingoing edges, and a special vertex, called sink, labelled f, which has no outgoing edges. Each non source vertex is labelled by a function of the values in its immediate predecessors. For every given values for $x_1, x_2, ..., x_n$ in $\{0, 1, ..., R-1\}$ the graph computes a value for each of its non-source vertices by applying the function with which it is labelled to the values of its immediate predecessors. In particular, this process assigns a value to the sink f. We assume that all intermediate results that are computed on nodes of G are from $\{0, 1, ..., R-1\}$. We say that G *computes* f if for every admissible values for $x_1, ..., x_n$ the computation on G assigns the value $f(x_1, ..., x_n)$ to the sink f.

If one pebbles the graph G one is only allowed to place a pebble on a node v if all immediate predecessors of G are currently occupied by pebbles. It is easy to see that any pebbling of G with p pebbles in T steps defines an *input oblivious* R-way branching program for f of width R^p and length T (the R^p vertices on each level of

the R-way branching program correspond to the R^p possible values on those p nodes that are currently occupied by pebbles, an input variable x_i is queried in the branching program if a pebble is placed on a source node of G that is labeled by x_i). Thus our preceding lower bounds on the length of input oblivious R-way branching programs of width $\leqslant R^p$ yield lower bounds on the number T of steps that are required to pebble with p pebbles arbitrary computation graphs for the same function. For example it follows from Theorem 4.2 that any pebbling of a computation graph for the sequence equality function $Q(n)$ with $o(n)$ pebbles requires superlinear time.

ACKNOWLEDGMENTS

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

REFERENCES

1. M. AJTAI, L. BABAI, P. HAJNAL, J. KOMLOS, P. PUDLÀK. V. RÖDL, E. SZEMEREDI, AND GY. TURÀN, Two lower bounds for branching programs, *in* "Proceedings, 18th ACM Symp. Theory of Comput., 1986, pp. 30–38.
2. L. BABAI, P. PUDLÀK, V. RÖDL, AND E. SZEMEREDI, Lower bounds to the complexity of symmetric Boolean functions, preprint.
3. D. A. BARRINGTON, Bounded width polynomial size branching programs recognize exactly those languages in NC^1, *in* "Proceedings, 18th ACM STOC, 1986," pp. 1–5.
4. M. BEN-OR, Lower bounds on algebraic computation trees, *in* "Proceedings, 15th ACM Symp. Theory of Comput, 1983, pp. 80–86.
5. B. BOLLOBÁS, "Extremal Graph Theory," Academic Press. New York/London, 1976.
6. A. BORODIN AND S. COOK, A time-space tradeoff for sorting on a general sequential model of computation, *SIAM J. Comput.* **11** (1982), 287–297.
7. A. BORODIN, D. DOLEV, F. FICH, AND W. PAUL, Bounds for width 2 branching programs, *in* "Proceedings, 15th ACM Symp. Theory of Comput., 1983, pp. 87–93.
8. A. CHANDRA, M. FURST, AND R. LIPTON, Multiparty protocols, *in* "Proceedings, 15th ACM Symp. Theory of Comput., 1983," pp. 94–99.
9. W. MAASS, On the use of inaccessible numbers and order indiscernables in lower bound arguments for random access machines, *J. Symbolic Logic*, in press.
10. E. NECHIPORUK, On a Boolean function, *Dokl. Akad. Nauk SSSR* **169**, No. 4 (1966), 765–766.
11. N. PIPPENGER, Superconcentrators of depth 2, *J.* Comput. *System Sci.* **24** (1982), 82–90.
12. P. PUDLÀK, A lower bound on the complexity of branching programs, "Proceedings, Conf. on the Math. Found. of Computer Science 1984," Lecture Notes in Computer Science Vol. 176, pp. 480–489, Springer-Verlag, Berlin/New York, 1984.
13. J. E. SAVAGE, "The Complexity of Computing," Wiley, London, 1976.
14. J. B. SHEARER, announced in [3].
15. A. C. YAO, Lower bounds by probabilistic arguments, *in* "Proceedings, 24th IEEE Found. of Comput. Sci., 1983, pp. 420–428.