1-1-1984

On Monotone Formulae with Restricted Depth

Maria M. Klawe
Harvey Mudd College

Wolfgang J. Paul
IBM Research Laboratory

Nicholas J. Pippenger
Harvey Mudd College

Mihalis Yannakakis
AT&T Bell Laboratories

ON MONOTONE FORMULAE WITH RESTRICTED DEPTH

(Preliminary Version)

Maria Klawe
IBM Research Laboratory
San Jose, CA 95193

Wolfgang J. Paul
IBM Research Laboratory
San Jose, CA 95193

Nicholas Pippenger
IBM Research Laboratory
San Jose, CA 95193

Mihalis Yannakakis
AT&T Bell Laboratories
Murray Hill, NJ 07974

ABSTRACT: We prove a hierarchy theorem for the representation of monotone Boolean functions by monotone formulae with restricted depth. Specifically, we show that there are functions with Π_k-formulae of size n for which every Σ_k-formula has size exp $\Omega(n^{1/(k-1)})$. A similar lower bound applies to concrete functions such as transitive closure and clique. We also show that any function with a formula of size n (and any depth) has a Σ_k-formula of size exp $O(n^{1/(k-1)})$. Thus our hierarchy theorem is the best possible.

1. Introduction

Circuits and formulae with unbounded fan-in but restricted depth have recently received attention for several reasons. They provide a convenient and elegant model for an important technology, programmed logic arrays, which has made it possible to give precise formulations and proofs for some widely-held beliefs about this technology (Furst, Saxe and Sipser [07] have shown that multiplication is "hard", while Chandra, Fortune and Lipton [04], [05] have shown that addition is "easy"). The model also provides a counterpart for circuit- and formula-based complexity theory of the notion of restricted alternation, which arises in a natural way in machine-based complexity theory (Furst, Saxe and Sipser [07] have indicated how the model might be used to establish results about the relativized polynomial-time hierarchy, while Sipser [15] indicates how even the unrelativized hierarchy can be attacked in this way.) Finally, the model can be used to obtain bounds on communication complexity (a notion introduced by Yao [19], and pursued by Papadimitriou and Sipser [14] and by Aho, Ullman and Yannakakis [02]).

When depth is held constant, sizes of circuits and formulae are polynomially related, so that results such as the ones in this paper apply equally to circuits and formulae. We shall state our results in terms of formulae and not mention circuits further. Our main concern in this paper is with monotone formulae, though some of our results have obvious extensions to the non-monotone case.

2. Lower Bounds

A Π_0-formula or Σ_0-formula is a literal (a variable or its negation). A Π_k-formula (respectively, a Σ_k-formula) is a conjunction (respectively, a disjunction) of Σ_{k-1}-formulae (respectively, of Π_{k-1}-formulae). The depth of a Π_k- or Σ_k-formula is k. The size of a Π_0- or Σ_0-formula is 1, and the size of a Π_k- or Σ_k-formula is the sum of the sizes of its constituent Σ_{k-1}- or Π_{k-1}-formulae. Formulae compute Boolean functions in an obvious way.

If f is a Boolean function, we shall let L(f) (respectively, $L_{\Pi_k}(f)$, $L_{\Sigma_k}(f)$) denote the minimum possible length of a formula (respectively, Π_k-formula, Σ_k-formula) computing f.

A formula is monotone if it involves only un-negated variables. If f is a monotone Boolean function, we shall let $L_M(f)$ (respectively, $L_{M\Pi_k}(f)$, $L_{M\Sigma_k}(f)$) denote the minimum possible length of a monotone formula (respectively, montone Π_k-formula, monotone Σ_k-formula) computing f.

The study of the complexity measures L_{Π_k} and L_{Σ_k} was initiated by Lupanov [11], [12], who showed that for "almost all" Boolean functions f of n variables, L(f), $L_{\Pi_3}(f)$ and $L_{\Sigma_3}(f)$ are all asymptotic to $2^n/\log_2 n$. He also showed that the function $\text{parity}_n(x_1, \ldots, x_n)$, which assumes the value 1 when an odd number of the variables $x_1, \ldots, x_n$ assume the value 1, both $L_{\Pi_2}(\text{parity}_n)$ and $L_{\Sigma_2}(\text{parity}_n)$ are equal to $n2^{n-1}$.

The complexity of monotone formulae for a function $f_m(x_1, \ldots, x_m, y_1, \ldots, y_m)$ of n=2m variables relevant to the process of carry propagation in addition wasalso studied by Lupanov [13]. He showed that, although $L_M(f_m)=2m$, if k≥2 is fixed, then both $L_{M\Pi_k}(f_m)$ and $L_{M\Sigma_k}(f_m)$ are $\Omega(m^{1/(k-1)})$. The complexity of monotone and non-monotone circuits for this function (and related ones) has been studied by Chandra, Fortune and Lipton [04], [05]. (Of course, for bounds such as these, which lie between linear and quadratic, the differences between formulae and circuits are important.)

The complexity of formulae computing parity was further studied by Furst, Saxe and Sipser [07] and independently by Ajtai [01]. They showed that for every k≥2 and ℓ,

$$L_{\Pi_k}(\text{parity}_n) = L_{\Sigma_k}(\text{parity}_n) = \Omega(n^{\ell})$$

(that is, there are no fixed depth, polynomial size formulae for parity). In fact, Ajtai has shown by a modification of the argument in [01] that for every k≥2,

$$L_{\Pi_k}(\text{parity}_n) = L_{\Sigma_k}(\text{parity}_n) = \exp \Omega((\log n)^2).$$

It seems unlikely that this result is the best possible; the sharpest upper bound known is

$$L_{\Pi_k}(\text{parity}_n) = L_{\Sigma_k}(\text{parity}_n) = \exp O(n^{1/(k-1)}).$$

For the purpose of obtaining results about relativized polynomial-time computations, it would be useful to know that for every k≥2 and ℓ,

$$L_{\Pi_k}(\text{parity}_n) = L_{\Sigma_k}(\text{parity}_n) = \exp \Omega((\log n)^{\ell}),$$

but this seems beyond the range of current techniques.

The foregoing results concerning parity also apply to the function $\text{majority}_n(x_1, \ldots, x_n)$, which assumes the value 1 if more than one-half of the variables $x_1, \ldots, x_n$ assume the value 1, since parity is reducible in fixed depth and polynomial size to majority (see Furst, Saxe and Sipser [07]). Since majority is monotone, one may hope to obtain stronger results for monotone formulae computing majority. This has been done by Yao [20], who has shown that

$$L_{M\Pi_3}(\text{majority}_n) = L_{M\Sigma_3}(\text{majority}_n) = \exp \Omega(n^{1/10}),$$

and by Boppana [03], who has shown that for every k≥2,

$L_{M\Pi_k}(\text{majority}_n) = L_{M\Sigma_k}(\text{majority}_n) = \exp \Omega(n^{1/(k-1)})$. This result cannot be far from the best possible; the sharpest upper bound known is

$L_{M\Pi_k}(\text{majority}_n) =$

$L_{M\Sigma_k}(\text{majority}_n) = \exp O(n^{1/(k-1)}(\log n)^{1-1/(k-1)})$.

Our main interest in this paper is the relative complexity of Π_k-formulae and Σ_k-formulae. For non-monotone formulae, this question has been studied by Sipser [15]. He showed that for every $k\geq 2$ and ℓ, there are functions f of n Boolean variables such that $L_{M\Pi_k}(f)=n$, but

$$L_{M\Sigma_k}(f) = \Omega(n^{\ell}).$$

(He actually states his result is a weaker form concerning variants of Σ_k and Σ_{k-1}, but the foregoing result is an immediate corollary.) For the purpose of obtaining results about relativized polynomial-time computations, it would be useful to know that

$$L_{\Sigma_k}(f) = \exp \Omega((\log n)^{\ell}),$$

but this seems beyond the range of current techniques.

Our main result in this paper is an analogue of the foregoing result for the case of monotone formulae. In return for the restriction to monotone formulae, we obtain much stronger lower bounds. Indeed, we shall see in the next section that our result is the best possible.

<u>Theorem 1</u>: For all $k\geq 2$ and n, there is a monotone Boolean function f such that $L_{M\Pi_k}(f)=n$, but

$$L_{M\Sigma_k}(f) = \exp \Omega(n^{1/(k-1)}).$$

If f is a Boolean function of the variables x_1, ..., x_n, then the <u>dual</u> of f, which will be denoted f^*, is the function $\neg f(\neg x_1, \ldots, \neg x_n)$, where "$\neg$" denotes negation. If f is monotone, then so is f^*. If a formula F computes f, then the formula obtained from F by exchanging conjunction and disjunction (and, if F involves constants, exchanging 0 and 1), which will be called the <u>dual</u> of F and denoted F^*, computes f^*. If F is a Π_k-formula, then F^* is a Σ_k-formula and <u>vice versa</u>.

For $k\geq 1$ and $m\geq 1$, we shall define the Boolean function $f_{m,k}$ of $n_{m,k}=2m^{k-1}$ Boolean variables. For k=1, define $f_{m,1}$ to be the conjunction of $n_{m,1}=2$ distinct variables. For $k\geq 2$, let X_1, ..., X_m be disjoint sets each comprising $n_{m,k-1}=2m^{k-2}$ variables (these sets will be called <u>beads</u>). For $1\leq\ell\leq m$, let $f_{m,k-1}(X_\ell)$ denote the result of substituting the variables in X_ℓ for the variables of $f_{m,k-1}$. Define $f_{m,k}$ to be the conjunction of the m functions $f_{m,k-1}(X_1)^*$, ..., $f_{m,k-1}(X_m)^*$.

Clearly, $L_{M\Pi_k}(f_{m,k})=n_{m,k}$. Thus it will suffice to show that

$$L_{M\Sigma_k}(f_{m,k}) = \exp \Omega(m),$$

since $m=\Omega((n_{m,k})^{1/(k-1)})$.

Let f be a monotone Boolean function. A subset P of the variables of f will be called a <u>path</u> for f if the funbtion obtained from f by substituting 1 for the variables in P is identically 1. We shall say that P is a <u>minimal</u> path if P is a path but no proper subset of P is a path. A subset Q of the variables of f will be called a <u>cut</u> for f if the function obtained from f by substituting 0 for the variables in Q is identically 0. We shall say that Q is a <u>minimal</u> cut if Q is a cut but no proper subset of Q is a cut. A minimal path for f is a minimal cut for f^* and <u>vice versa</u>.

Let $\mathbf{P}_{m,k}$ denote the set of minimal paths for $f_{m,k}$ and let $\mathbf{Q}_{m,k}$ denote the set of minimal cuts for $f_{m,k}$. Let $p_{m,k}=|\mathbf{P}_{m,k}|$ and let $q_{m,k}=|\mathbf{Q}_{m,k}|$. From the definition of $f_{m,k}$, we have $p_{m,1}=1$, $q_{m,1}=2$ and the recurrences $p_{m,k}=(q_{m,k-1})^m$, $q_{m,k}=mp_{m,k-1}$.

Let G be a monotone Boolean formula, let P belong to $\mathbf{P}_{m,k}$ and let Q belong to $\mathbf{Q}_{m,k}$. We shall say that G recognizes P if P is a path for G and that G recognizes Q if Q is a cut for G. We shall say that G approximates $f_{m,k}$ if G recognizes at least $p_{m,k}/m$ paths in $\mathbf{P}_{m,k}$ and at least $q_{m,k}/2$ cuts in $\mathbf{Q}_{m,k}$.

Proposition 1.1: For all $k\geq 2$ and $m\geq 3$, if G is a Σ_k-formula that approximates $f_{m,k}$, then $L(G)>2^{m/2}/2m$.

Proof: Suppose that G is a Σ_k-formula that approximates $f_{m,k}$ and that $L(G)\leq 2^{m/2}/2m$. We shall derive a contradiction.

Since G is a Σ_k-formula, it is the disjunction of some set $\{G_1, \ldots, G_j\}$ of Π_{k-1}-formulae. If $j>2^{m/2}/2m$, we are done, so suppose $j\leq 2^{m/2}/2m$. Since G is monotone, if G recognizes a path P, then one of the subformulae $G_1, \ldots, G_j$ must recognize P. Since G recognizes at least $p_{m,k}/m$ paths in $\mathbf{P}_{m,k}$, some subformula G_i for $1\leq i\leq j$ must recognize at least $(p_{m,k}/m)/(2^{m/2}/2m)=p_{m,k}/2^{m/2-1}$ paths in $\mathbf{P}_{m,k}$. Of course, if G recognizes a cut Q, then each of the subformulae $G_1, \ldots, G_j$ must recognize Q. Thus G_i also recognizes at least $q_{m,k}/2$ cuts in $\mathbf{Q}_{m,k}$.

We now proceed by induction on k. If k=2, then G_i recognizes at least $p_{m,2}/2^{m/2-1}=2^{m/2+1}$ of the 2^m paths in $\mathbf{P}_{m,2}$ and at least $q_{m,2}/2=m/2$ of the m cuts in $\mathbf{Q}_{m,2}$. Since G_i is a Π_1-formula, it is the conjunction of some set Γ_i of variables. The cuts in $\mathbf{Q}_{m,2}$ are the beads $X_1, \ldots, X_m$, and G_i recognizes a cut X_ℓ for $1\leq \ell\leq m$ if and only if Γ_i includes a variable in X_ℓ. Thus Γ_i contains at least m/2 variables. The paths in $\mathbf{P}_{m,2}$ are the systems of distinct representatives from $\{X_1, \ldots, X_m\}$, and G_i recognizes a path $\{x_1, \ldots, x_m\}$ if and only if $\{x_1, \ldots, x_m\}$ contains Γ_i. The number of such paths is at most $2^{m-m/2}=2^{m/2}$, contradicting the assumption that G_i recognizes at least $2^{m/2+1}$ paths in $\mathbf{P}_{m,2}$. This completes the proof for k=2.

Now suppose $k\geq 3$. If $1\leq \ell\leq m$ and P_ℓ is a minimal path for $(f_{m,k-1}(X_\ell))^*$, we shall say that G_i respects P_ℓ if G_i recognizes some path in $\mathbf{P}_{m,k}$ that contains P_ℓ. For $1\leq \ell\leq m$, let a_ℓ denote the number of minimal paths for $(f_{m,k-1}(X_\ell))^*$ that are respected by G_i.

If G_i recognizes a path P in $\mathbf{P}_{m,k}$, then for $1\leq \ell\leq m$, it respects the intersection of P and X_ℓ, which is a minimal path for $(f_{m,k-1}(X_\ell))^*$. Since G_i recognizes at least $p_{m,k}/2^{m/2-1}$ paths in $\mathbf{P}_{m,k}$, we have $p_{m,k}/2^{m/2-1}\leq a_1\ldots a_m$.

Since a minimal path for $(f_{m,k-1}(X_\ell))^*$ is a minimal cut for $f_{m,k}(X_\ell)$, we have $a_\ell\leq q_{m,k-1}$. Let us say that a bead X_ℓ is weak if $a_\ell<q_{m,k-1}/2$. Let m' denote the number of weak beads. Then $a_1\ldots a_m\leq (q_{m,k-1}/2)^{m'}(q_{m,k-1})^{m-m'}$. Combining these inequalities for $a_1\ldots a_m$ and using the relation $p_{m,k}=(q_{m,k-1})^m$, we obtain $m'\leq m/2-1$.

If $1\leq \ell\leq m$ and Q_ℓ is a minimal cut for $(f_{m,k-1}(X_\ell))^*$, then Q_ℓ is also a minimal cut for $f_{m,k}$. For $1\leq \ell\leq m$, let b_ℓ denote the number of minimal cuts for $(f_{m,k-1}(X_\ell))^*$ that are recognized by G_i.

A cut in $\mathbf{Q}_{m,k}$ is a minimal cut for $(f_{m,k-1}(X_\ell))^*$ for some $1\leq \ell\leq m$. Since G_i recognizes at least $q_{m,k}/2$ cuts in $\mathbf{Q}_{m,k}$, we have $q_{m,k}/2\leq b_1+\ldots+b_m$.

Since a minimal cut for $(f_{m,k-1}(X_\ell))^*$ is a minimal path for $f_{m,k-1}(X_\ell)$, we have $b_\ell\leq p_{m,k-1}$. Let us say that a bead X_ℓ is poor if $b_\ell<p_{m,k-1}/m$. Let m'' denote the number of poor beads. Then $b_1+\ldots+b_m\leq m''p_{m,k-1}/m+(m-m'')p_{m,k-1}$. Combining these inequalities for $b_1+\ldots+b_m$ and using the relation $q_{m,k}=mp_{m,k-1}$, we obtain $m''\leq m^2/2(m-1)$.

Since $m\geq 3$, $m'+m''<m$, so there is some bead X_ℓ that is neither weak nor poor. This means that G_i

respects at least $q_{m,k-1}/2$ minimal paths for $(f_{m,k-1}(X_\ell))^*$ and recognizes at least $p_{m,k-1}/m$ minimal cuts for $(f_{m,k-1}(X_\ell))^*$.

Let H be the formula obtained from G_i by substituting the variables of $f_{m,k-1}$ for the variables in X_ℓ and substituting 1 for all other variables. not in X_ℓ. Then H^* is a Σ_{k-1}-formula that approximates $f_{m,k-1}$. Since $L(H^*)\le L(G_i)\le 2^{m/2}/2m$, this contradicts the inductive hypotheses. □

We shall conclude this section with some corollaries of Theorem 1. The proofs of these corollaries are omitted from this preliminary version.

Our first corollaries extend the lower bound of Theorem 1 to functions such as transitive closure and clique. This extension goes by way of monotone projections and completeness (see Valiant [17] and Skyum and Valiant [16]).

Let $\text{path}_t(x_{1,2}, \ldots, x_{t-1,t})$ be the Boolean function that assumes the value 1 if the acyclic directed graph with vertices corresponding to the indices {1, ..., t} and edges corresponding to the variables $\{x_{i,j}\}_{1\le i<j\le t}$ has a path of edges corresponding to 1's from vertex 1 to vertex t. We can prove

<u>Proposition 2</u>: The function $f_{m,k}$ is a monotone projection of path_t for $t=4m^{k-1}$.

Combining this with Theorem 1 yields

<u>Corollary 3</u>: For every $k\ge 2$,

$$L_{M\Sigma_k}(\text{path}_t) = \exp\,\Omega(t^{1/(k-1)}).$$

Let $\text{clique}_{s,t}(x_{1,1}, \ldots, x_{s,s})$ be the Boolean function that assumes the value 1 if the s-by-s matrix of 0's and 1's $\{x_{i,j}\}_{1\le i\le s, 1\le j\le s}$ contains a t-by-t principal minor consisting entirely of 1's.

Valiant [18] showed that

$$L_{M\Sigma_3}(\text{clique}_{2t,t}) = \exp\,\Omega(t^{1/2}),$$

and Yao [20] showed that

$$L_{M\Sigma_4}(\text{clique}_{2t,t}) = \exp\,\Omega(t^{\varepsilon})$$

for some unstated value of $\varepsilon>0$. We can prove

<u>Proposition 4</u>: The function $f_{m,k}$ is a monotone projection of $\text{clique}_{s,t}$, where $s=2m^{k-1}$, $t=2^r m^q$ and $k=2q+r$ with $0\le r<2$.

Combining this with Theorem 1 yields

<u>Corollary 5</u>: For every $k\ge 2$,

$$L_{M\Sigma_k}(\text{clique}_{2t,t}) = \exp\,\Omega(t^{1/(k-1)}).$$

Finally, let us mention an application of Theorem 1 to communication complexity. Consider a function f of 2n Boolean variables $\{x_1, \ldots, x_n, y_1, \ldots, y_n\}$ Consider a distributed computation of f by two participants: X, who has access to the variables $\{x_1, \ldots, x_n\}$, and Y, who has access to the variables $\{y_1, \ldots, y_n\}$ (we are considering a fixed partition of the variables). We shall let $C_{k,X}(f)$ (respectively, $C_{k,Y}(f)$) denote the communication complexity of computing f when at most k messages are sent and X (respectively, Y) sends the first message.

<u>Corollary 6</u>: There is a function f of 2n Boolean variables such that $C_{k,X}(f)=O(\log n)$, but $C_{k,Y}(f)=\Omega(n^{1/(k-1)})$.

This partially answers a question raised by Papadimitroiu and Sipser [14]. A much more satisfactory answer has been given by Duris, Galil and Schnitger [06]. They show that there is a function f of 2n Boolean variables such that, for <u>any</u> partition of the variables into 2 sets of n variables, $C_{k,X}(f)=O(\log n)$, but $C_{k,Y}(f)=\Omega(n)$.

3. Upper Bounds

In this section we shall show that monotone Boolean functions that have small monotone formulae (with any depth) also have monotone formulae with restricted depth and sub-exponential size. This result shows that the lower bound of Theorem 1 is the best possible.

Theorem 7: For every $k\geq 2$ and every Boolean function f, if $L_M(f)=n$, then

$$L_{M\Sigma_k}(f) = \exp O(n^{1/(k-1)}).$$

The proof of this theorem follows the paradigm of the final proposition of Valiant [18]. We regard a formula as a tree and use a "fragmentation lemma" to break the tree into small pieces. We construct formulae with restricted depth simulating each piece, then combine these into a formula with restricted depth computing the original function.

A binary tree consists of a node called the root, which may have no children (in which case it is a leaf) or which may have two children that are the roots of binary trees (in which case it is an internal node). If T is a binary tree, $\rho(T)$ will denote the root of T and $\lambda(T)$ will denote the number of leaves in T. The following two lemmas generalize the well known (1/3, 2/3)-Lemma of Lewis, Stearns and Hartmanis [08].

Lemma 7.1: For ξ a real number and T a binary tree, if $1\leq\xi\leq\lambda(T)$, then there is a node v in T such that $\xi\leq\lambda(T_v)<2\xi$.

Proof: We proceed by induction on $\lambda(T)$. If $\lambda(T)=1$, then $\xi=1$ and we may take $v=\rho(T)$. If $\lambda(T)\geq 2$, then $\rho(T)$ has two children, say x and y. If $\lambda(T)<2\xi$, then we may again take $v=\rho(T)$. If $\lambda(T)\geq 2\xi$, then $\lambda(T_w)\geq\xi$ for some w in {x, y}. We also have $\lambda(T_w)<\lambda(T)$, so by inductive hypothesis, there is a node v in T_w (and therefore in T) such that $\xi\leq\lambda(T_v)<2\xi$. □

By a forest we shall mean a set of binary trees. If Φ is a forest, $|\Phi|$ will denote the number of trees in Φ. Let T be a binary tree and let W be a set of nodes of T that are neither $\rho(T)$ nor leaves of T. We may decompose T into a forest Φ by splitting each node w in W into two new nodes, one a new leaf with the same parent as w, the other a new root with the same children as w.

Lemma 7.2: For ξ a real number and T a binary tree, if $\xi\geq 2$ and $\lambda(T)\geq 2$, then T can be decomposed into a forest Φ such that $|\Phi|\leq(\lambda(T)+\xi-3)/(\xi-1)$ and, for each tree S in Φ, $\lambda(S)<2\xi$.

Proof: We proceed by induction on $\lambda(T)$. If $\lambda(T)<2\xi$, then we may take $\Phi=\{T\}$, since $\lambda(T)\geq 2$ implies $(\lambda(T)+\xi-3)/(\xi-1)\geq 1$. If $\lambda(T)\geq 2\xi$, then by Lemma 7.1, there is a node v in T such that $\xi\leq\lambda(T_v)<2\xi$. Let T' be the tree obtained from T by substituting a leaf for T_v. Then $\lambda(T')=\lambda(T)-\lambda(T_v)+1$. Since $\lambda(T)\geq 2\xi$ and $\lambda(T_v)<2\xi$, we have $\lambda(T')\geq 2$. Since $\lambda(T_v)\geq\xi\geq 2$, we have $\lambda(T')\leq\lambda(T)-1$. Thus, by inductive hypothesis, T' can be decomposed into a forest Φ' such that $|\Phi'|\leq(\lambda(T')+\xi-3)/(\xi-1)\leq(\lambda(T)-2)/(\xi-1)$ and, for every tree S in Φ', $\lambda(S)<2\xi$. If we take Φ to be the union of Φ' and $\{T_v\}$, then $|\Phi|\leq(\lambda(T)-2)/(\xi-1)+1=(\lambda(T)+\xi-3)/(\xi-1)$, which completes the proof. □

Corollary 7.3: For η a real number and T a binary tree, if $\lambda(T)\geq\eta\geq 6$, then T can be decomposed into a forest Φ such that $|\Phi|\leq 3\lambda(T)/\eta$ and, for every tree S in Φ, $\lambda(S)\leq\eta$.

Proof: Let $\xi=\eta/2$. Since $\xi\geq 3$, $(\zeta+\xi-3)/\zeta$ is a non-increasing function of ζ. Since $\lambda(T)\geq 2\xi$, we have $(\lambda(T)+\xi-3)/\lambda(T)\leq(2\xi+\xi-3)/2\xi=3(\xi-1)/2\xi$. This implies that $(\lambda(T)+\xi-3)/(\xi-1)\leq 3\lambda(T)/2\xi$. Thus applying Lemma 7.2 yields a forest Φ such that $|\Phi|\leq 3\lambda(T)/2\xi=3\lambda(T)/\eta$ and, for every tree S in Φ, $\lambda(S)<2\xi=\eta$. □

For $k\geq 2$ and $n\geq 6$, let $A_k(n)$ denote the maximum of $L_{M\Sigma_k}(f)$ over all monotone Boolean functions f such that $L_M(f)\leq n$. Since $L_{M\Pi_k}(f)=L_{M\Sigma_k}(f^*)$ and $L_M(f^*)=L_M(f)$, $A_k(n)$ is also the maximum of $L_{M\Pi_k}(f)$ over all f such that $L_M(f)\leq n$.

Proposition 7.4: For all $k\geq 3$ and real numbers $n\geq m\geq 6$,

$$A_k(n) \leq (3n/2m)2^{3n/m} A_{k-1}(m).$$

Proof: Let f be a monotone Boolean function such that $L_M(f)\leq n$ and $L_{M\Sigma_k}(f)=A_k(n)$. It will suffice to show that

$$L_{M\Sigma_k}(f) \leq (3n/2m)2^{3n/m} A_{k-1}(m).$$

If f is a function of the variables X, then f is computed by some binary tree T with $\lambda(T)\leq n$, where the internal nodes of T are labelled with the operations "conjunction" and "disjunction" and the leaves of T are labelled with variables from X. By Corollary 7.3, T may be decomposed into a forest Φ such that $|\Phi|\leq 3n/m$ and, for every tree S in Φ, $\lambda(S)<m$.

Define a total order on Φ in such a way that, for any trees S and S' in Φ, if $\rho(S')$ is a descendant of $\rho(S)$ in T, then S'<S. This order has a maximal element R, and $\rho(R)=\rho(T)$. For every S in Φ, define the segment $\Phi(S)$ to be the set of all S' in Φ such that S'<S.

For every S in $\Phi(R)$, define a new Boolean variable y_S, and for every vertex in T that splits into a leaf and a root $\rho(S)$ in Φ, label the leaf with the variable y_S. Let Y_S denote the set of variables $y_{S'}$ for all S' in $\Phi(S)$.

For every S in Φ, let g_S be the monotone Boolean function of the variables X and Y_S computed by S. (Of course, g_S might not actually depend upon all of these variables.) Since $\lambda(S)<m$, $L_M(g_S)\leq m$ and thus

$$L_{M\Pi_{k-1}}(g_S) \leq A_{k-1}(m).$$

For every subset Ψ of $\Phi(R)$, let $g_{S,\Psi}$ be the function of the variables X obtained from g_S by substituting 1 for those variables $y_{S'}$ in Y_S such that S' belongs to Ψ and substituting 0 for all other variables in Y_S. Clearly,

$$L_{M\Pi_{k-1}}(g_{S,\Psi}) \leq A_{k-1}(m).$$

For every subset Ψ of $\Phi(R)$, let f_Ψ be the conjunction of $g_{R,\Psi}$ and those functions $g_{S,\Psi}$ for which S belongs to Ψ. Since $|\Psi|+1\leq|\Phi|\leq 3n/m$, we have

$$L_{M\Pi_{k-1}}(f_\Psi) \leq (3n/m)A_{k-1}(m).$$

It is routine to verify that f is the disjunction of those functions f_Ψ for which Ψ is a subset of $\Phi(R)$. Since there are $2^{|\Phi(R)|}\leq 2^{2n/m-1}$ such subsets,

$$L_{M\Sigma_k}(f) \leq (3n/2m)2^{3n/m} A_{k-1}(m),$$

which completes the proof. □

Corollary 7.5: For $k\geq 2$ and $n\geq 36$,

$$A_k(n) \leq (3/2)^{k-2}n2^{3(k-1)n^{1/(k-1)}}.$$

Proof: We proceed by induction on k. If k=2, disjunctive normal form shows that

$$A_2(n) \leq n2^n \leq n2^{3n}.$$

If $k\geq 3$, take $m=n^{(k-2)/(k-1)}\geq 6$. By inductive hypothesis,

$$A_{k-1}(m) \leq (3/2)^{k-3}m2^{3(k-2)m^{1/(k-2)}}.$$

Thus, by Proposition 7.4,

$$A_k(n) \leq (3n/2m)2^{3n/m} (3/2)^{k-3}m2^{3(k-2)m^{1/(k-2)}} = (3/2)^{k-2}n2^{3(k-1)n^{1/(k-1)}},$$

which completes the proof. □

The same method of proof can be used to show that functions that have small planar monotone circuits have monotone formulae with restricted depth and sub-exponential size.

Theorem 8: For every k≤2, if f has a planar monotone circuit of size n, then

$$L_{M\Sigma_{2k-1}}(f) = \exp O(n^{2^{k-1}/(2^k-1)}).$$

The proof, which is omitted in this preliminary version, is similar to that of Theorem 7, except that the "fragmentation lemma" is obtained from the planar separator theorem of Lipton and Tarjan [09], [10]. Theorems 7 and 8 have analogues for non-monotone functions, formulae and circuits; these analogues have virtually identical proofs.

4. References

[01] M. Ajtai, "Σ_1^1-Formulae on Finite Structures", Ann. Pure and Appl. Logic, 24 (1983) 1-48.

[02] A. V. Aho, J. D. Ullman and M. Yannakakis, "On Notions of Information Transfer in VLSI Circuits", ACM Symp. on Theory of Computing, 15 (1983) 133-139.

[03] R. B. Boppana, "Threshold Functions and Bounded Depth Monotone Circuits", ACM Symp. on Theory of Computing, 16 (1984).

[04] A. K. Chandra, S. J. Fortune and R. J. Lipton, "Unbounded Fan-In Circuits and Associative Functions", ACM Symp. on Theory of Computing, 15 (1983) 52-60.

[05] A. K. Chandra, S. J. Fortune and R. J. Lipton, "Lower Bounds for Constant Depth Monotone Circuits for Prefix Functions", Internat Colloq. on Automata, Languages and Programming, 10 (1983) 109-117.

[06] P. Duris, Z. Galil and G. Schnitger, "Lower Bounds on Communication Complexity", ACM Symp. on Theory of Computing, 16 (1984).

[07] M. Furst, J. B. Saxe and M. Sipser, "Parity, Circuits and the Polynomial Time Hierarchy", IEEE Symp. on Foundations of Computer Science, 22 (1981) 260-270.

[08] P. M. Lewis, R. E. Stearns and J. Hartmanis, "Memory Bounds for Recognition of Context-Free and Context-Sensitive Languages", IEEE Conf. on Switching Theory and Logical Design, 6 (1965) 191-202.

[09] R. J. Lipton and R. E. Tarjan, "A Separator Theorem for Planar Graphs", SIAM J. Appl. Math., 36 (1979) 177-189.

[10] R. J. Lipton and R. E. Tarjan, "Applications of a Planar Separator Theorem", SIAM J. Comp., 9 (1980) 615-627.

[11] O. B. Lupanov, "Implementing the Algebra of Logic Functions in Terms of Bounded Depth Formulas in the Basis &, , -", Sov. Phys. Dokl., 6 (1961)107-108.

[12] O. B. Lupanov, "On the Realization of Functions of Logical Algebra by Formulae of Finite Classes (Formulae of Limited Depth) in the Basis &, , -", Prob. Cyb., 6 (1961) 1-14.

[13] O. B. Lupanov, "Effect of the Depth of Formulas on Their Complexity", Cybernetics, 2 (1970) 62-66.

[14] C. H. Papadimitriou and M. Sipser, "Communication Complexity", ACM Symp. on Theory of Computing, 15 (1982) 196-200.

[15] M. Sipser, "Borel Sets and Circuit Complexity", ACM Symp. on Theory of Computing 15 (1983) 61-69.

[16] S. Skyum and L. G. Valiant, "A Complexity Theory Based on Boolean Algebra", IEEE Symp. on Foundations of Computer Science, 22 (1981) 244-253.

[17] L. G. Valiant, "Completeness Classes in Algebra", ACM Symp. on Theory of Computing, 11 (1979) 249-261.

[18] L. G. Valiant, "Exponential Lower Bounds for Restricted Monotone Circuits", ACM Symp. on Theory of Computing, 15 (1983) 110-117.

[19] A. C. Yao, "Some Complexity Questions Related to Distributive Computing", ACM Symp. on Theory of Computing, 11 (1979) 209-213.

[20] A. C. Yao, "Lower Bounds by Probabilistic Arguments", IEEE Symp. on Foundations of Computer Science, 24 (1983) 420-428.