Theoretical Computer Science 9 (1979) 83–97

SWITCHING FUNCTIONS WHOSE MONOTONE COMPLEXITY IS NEARLY QUADRATIC

Ingo WEGENER

Fakultät für Mathematik der Universität Bielefeld, Universitätsstraße 1, 4800 Bielefeld 1, Federal Republic of Germany

Communicated by M.S. Paterson
Received November 1977
Revised June 1978

Abstract. A sequence of monotone switching functions $h_n : \{0, 1\}^n \to \{0, 1\}^n$ is constructed, such that the monotone complexity of h_n grows faster than $\Omega(n^2 \log^{-2} n)$. Previously the best lower bounds of this nature were several $\Omega(n^{3/2})$ bounds due to Pratt, Paterson, Mehlhorn and Galil and Savage.

1. Introduction and summary

It is well-known (Paterson [3], Mehlhorn and Galil [2]), that n^3 $\wedge$-gates and $n^3 - n^2$ $\vee$-gates are necessary and sufficient to compute the Boolean matrix product of two $n \times n$ matrices in monotone circuits, i.e. if only $\wedge$- and $\vee$-gates are available. (For $1 \leq i, j \leq n$ let $c_{ij} := \bigvee_{1 \leq l \leq n} a_{il} \wedge b_{lj}$, then

$$C = (c_{ij})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq n}}$$

is the Boolean matrix product of the matrices

$$A = (a_{ij})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq n}} \quad \text{and} \quad B = (b_{ij})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq n}}.)$$

In computing c_{ij} one looks at the ith row of A and at the jth column of B and examines, if these two vectors have a common 1. In order to define more complex monotone functions we generalize the Boolean matrix product of two matrices to a "direct product" of m matrices ($m \in \mathbf{N}$, $m \geq 2$).

Let m $M \times N$ matrices be given:

$$(x^1_{h_1 l})_{\substack{1 \leq h_1 \leq M \\ 1 \leq l \leq N}}, \ldots, (x^m_{h_m l})_{\substack{1 \leq h_m \leq M \\ 1 \leq l \leq N}}.$$

We want to compute an output function for each choice of one row per matrix. If we have chosen the h_1th row of the first matrix, the h_2th row of the second matrix, ... and the h_mth row of the last matrix, we examine, if these rows all together have a common 1.

Now we formalize these considerations:

Definition 1.1. Let $m, M, N \in \mathbb{N}$ and $m \geq 2$. We want to define monotone functions $f^m_{MN}: \{0,1\}^{mMN} \to \{0,1\}^{M^m}$. Therefore we group the mMN variables to m $M \times N$ matrices

$$(x^1_{h_1 l})_{\substack{1 \leq h_1 \leq M \\ 1 \leq l \leq N}}, \ldots, (x^m_{h_m l})_{\substack{1 \leq h_m \leq M \\ 1 \leq l \leq N}}$$

and we denote the M^m output functions by $y_{h_1 \cdots h_m}$ $(1 \leq h_1, \ldots, h_m \leq M)$. Finally we define

$$y_{h_1} \cdots {}_{h_m} := \bigvee_{1 \leq l \leq N} x^1_{h_1 l} x^2_{h_2 l} \cdots x^m_{h_m l}.$$

By investigating these functions it is natural to make the following conjecture: An optimal monotone circuit to compute f^m_{MN} first computes all products $x^1_{h_1 l} \cdots x^m_{h_m l}$ using only $\wedge$-gates and then combines the results using only $\vee$-gates. This conjecture is true for the special case $m = 2$ (Paterson [3], Mehlhorn and Galil [2]), because f^2_{MN} computes the Boolean matrix product of the matrix

$$(x^1_{h_1 l})_{\substack{1 \leq h_1 \leq M \\ 1 \leq l \leq N}}$$

and the transposed matrix of

$$(x^2_{h_2 l})_{\substack{1 \leq h_2 \leq M \\ 1 \leq l \leq N}}.$$

In Section 3 we give upper bounds for the monotone complexity of f^m_{MN} resulting from the above ideas. Then we give a rather surprising result: if we count only the number of $\vee$-gates we often can do much better. That circuit also shows, that it would be difficult to prove good lower bounds for the number of $\vee$-gates in each monotone circuit computing f^m_{MN}.

In Section 4 we present a slight generalization of a replacement rule due to Mehlhorn and Galil [2] and show how this result can be used for monotone circuits computing f^m_{MN}.

In Section 5 we prove a good lower bound for the number of $\wedge$-gates in each monotone circuit computing f^m_{MN}. This lower bound is, for fixed m, optimal up to a constant. We follow the proof of Paterson [3] for the Boolean matrix product and we make use of the results we have derived in Section 4. Beyond that we introduce a new elimination rule and use the following assumption (which is very common in algebraic complexity but was never used before in network complexity): we assume that certain functions (besides the variables and the constants) are given for free.

We look at those gates in a monotone circuit computing f^m_{MN}, where for the first time $x^1_{h_1 1} \cdots x^m_{h_m 1}$ is a prime implicant of the computed function. It is obvious, that these gates are $\wedge$-gates. If at one gate not only $x^1_{h_1 1} \cdots x^m_{h_m 1}$ but also $x^1_{h'_1 1} \cdots x^m_{h'_m 1}$ is for the first time a prime implicant of the computed function $((h_1, \ldots, h_m) \neq (h'_1, \ldots, h'_m))$, then one can alter the circuit, such that the new monotone circuit again computes f^m_{MN}. By the assumption, that certain functions are given for free, the new circuit contains not more $\wedge$-gates than the given circuit. By repeating this procedure we get a monotone circuit computing f^m_{MN} with not more $\wedge$-gates than the given circuit and with the following additional property: if $x^1_{h_1 1} \cdots x^m_{h_m 1}$ is for the first time a prime implicant of the computed function, then this is not true for $x^1_{h'_1 1} \cdots x^m_{h'_m 1}$ $((h_1, \ldots, h_m) \neq (h'_1, \ldots, h'_m))$. The last step is to fix all variables $x^i_{h_i 1}$ $(1 \leq i \leq m, 1 \leq h_i \leq M)$ in such a way, that the following two aims are fulfilled: (1) The remaining circuit computes f^m_{MN-1}. (2) Many of the $\wedge$-gates we have examined before (where some $x^1_{h_1 1} \cdots x^m_{h_m 1}$ is for the first time a prime implicant of the computed function) can be eliminated, because one of the input functions is changed to the constant function 1. This finishes the proof of the lower bound.

In Section 6 we use this result to define a sequence of monotone switching functions $h_n : \{0, 1\}^n \to \{0, 1\}^n$, whose monotone complexity grows faster than $\Omega(n^2 \log^{-2} n)$. Thus the largest known lower bound for the monotone complexity of a series of explicitly defined monotone functions is considerably increased.

Pratt [4] raised the following question: what is the greatest loss of economy a circuit designer may incur in implementing monotone functions using only $\wedge$- and $\vee$-gates? He examined for the Boolean matrix product the quotient of the monotone complexity and the complexity over the complete basis $\{\wedge, \vee, -\}$. We show in Section 7, that for many other functions this quotient is at least as large as the lower bound for the Boolean matrix product proved by Pratt, which is the largest known lower bound of this kind.

Firstly we give in Section 2 some definitions and notations we shall use later. The reader who is familiar with the ideas, which are connected with the theory of monotone circuits, may omit the Definitions 2.1–2.9.

2. Definitions and notations

Definition 2.1. Let g be a monotone function. $C_{\{\wedge,\vee\}}(g)$ is the monotone complexity of g, i.e. the complexity of g over the basis $\{\wedge, \vee\}$. $C^{\wedge}_{\{\wedge,\vee\}}(g)(C^{\vee}_{\{\wedge,\vee\}}(g))$ is the minimal number of $\wedge$-gates ($\vee$-gates) in each monotone circuit computing g.

In the following we denote the variables by $x_1, \ldots, x_n$.

Definition 2.2. A function t, which is the product of some variables is called a monom:

$$t(x_1, \ldots, x_n) = \bigwedge_{1 \leq j \leq m} x_{i_j} \quad (i_1, \ldots, i_m \in \{1, \ldots, n\}).$$

The empty product is the constant function 1.

Definition 2.3. The monom t is an implicant of the monotone function g, if $t \leq g$, i.e. for any $\alpha \in \{0, 1\}^n t(\alpha) = 1$ implies $g(\alpha) = 1$. Let $\mathrm{I}(g)$ be the set of all implicants of g.

Definition 2.4. The implicant t of the monotone function g is a prime implicant of g, if for all monoms t' the following is true: $t \leq t'$ and $t \neq t' \Rightarrow t' \notin \mathrm{I}(g)$. Let $\mathrm{PI}(g)$ be the set of all prime implicants of g.

Definition 2.5. For each monotone function g it is true, that

$$g(x_1, \ldots, x_n) = \bigvee_{t \in \mathrm{PI}(g)} t(x_1, \ldots, x_n).$$

This representation of g is called the monotone disjunctive normal form of g (MDNF(g)).

We have defined $y_{h_1 \cdots h_m}$ in Definition 1.1. by its MDNF.
Now we give similar definitions by interchanging the roles of $\wedge$ and $\vee$.

Definition 2.6. A function u is called a sum of some variables, if

$$u(x_1, \ldots, x_n) = \bigvee_{1 \leq j \leq m} x_{i_j} \quad (i_1, \ldots, i_m \in \{1, \ldots, n\}).$$

The empty sum is the constant function 0.

Definition 2.7. The sum u is a clause of the monotone function g, if $g \leq u$, i.e. in order that $g(\alpha) = 1$ it is necessary, that $u(\alpha) = 1$ ($\alpha \in \{0, 1\}^n$). Let $\mathrm{Cl}(g)$ be the set of all clauses of g.

Definition 2.8. The clause u of the monotone function g is a prime clause of g, if for all sums u' the following is true: $u' \leq u$ and $u' \neq u \Rightarrow u' \notin \mathrm{Cl}(g)$. Let $\mathrm{PC}(g)$ be the set of all prime clauses of g.

Definition 2.9. For each monotone function g it is true, that

$$g(x_1, \ldots, x_n) = \bigwedge_{u \in \mathrm{PC}(g)} u(x_1, \ldots, x_n).$$

This representation of g is called the monotone conjunctive normal form of g (MCNF(g)).

For the proof of the main lemma in Section 5 we shall need the following definitions of the length of a monotone function and the length of a gate of a monotone circuit.

Definition 2.10. The length of a monom

$$t(x_1, \ldots, x_n) = \bigwedge_{1 \leq j \leq m} x_{i_j} \quad (i_1, \ldots, i_m \text{ distinct})$$

is defined by $\mathrm{L}(t) = m$.

Definition 2.11. The length of a monotone function g is defined by

$$L(g)=\sum_{t\in PI(g)} L(t),$$

the number of variables in its MDNF.

Definition 2.12. Let G be a gate of the monotone circuit S, G_1 and G_2 the two direct predecessors of G in S (inputs of gates) and s_1 and s_2 the functions, which are computed at G_1 and G_2. Then $L(G):=L(s_1)+L(s_2)$ is the length of the gate G.

The last definition is unusual, but it turns out to be useful.

Notation. (1) Let t be the monom

$$t(x_1,\ldots,x_n)=\bigwedge_{1\le j\le m} x_{i_j},$$

then t will denote also the set $\{x_{i_j}\mid 1\le j\le m\}$.

(2) Let $f, g:\mathbf{N}\to\mathbf{R}^+$.

$$f=O(g):\Leftrightarrow \exists\, c\in\mathbf{R}^+, N_0\in\mathbf{N}\;\forall\, n\ge N_0\quad f(n)\le cg(n).$$

$$f=\Omega(g):\Leftrightarrow \exists\, c\in\mathbf{R}^+, N_0\in\mathbf{N}\;\forall\, n\ge N_0\quad f(n)\ge cg(n).$$

3. Upper bounds for the monotone complexity of f^m_{MN}

By examining the MDNF of $y_{h_1\cdots h_m}$ we obtain the following monotone circuit computing f^m_{MN}. Take M^2N $\wedge$-*gates to compute* $x^1_{h_1l}x^2_{h_2l}$ $(1\le h_1,\ h_2\le M,\ 1\le l\le N)$. Then take M^3N $\wedge$-gates to compute $x^1_{h_1l}x^2_{h_2l}x^3_{h_3l}$ $(1\le h_1,\ h_2,\ h_3\le M,\ 1\le l\le N)$ and so on. $N\sum_{2\le i\le m}M^i$ $\wedge$-gates are sufficient to compute $\bigwedge_{1\le i\le m}x^i_{h_il}$ for $(h_1,\ldots,h_m)\in\{1,\ldots,M\}^m$ and $l\in\{1,\ldots,N\}$. Using these functions $N-1$ $\vee$-gates are sufficient to compute each function $y_{h_1\cdots h_m}$, therefore $(N-1)M^m$ $\vee$-gates are sufficient to compute the function f^m_{MN}.

Summarizing we proved

Theorem 3.1. (i) $C_{\{\wedge,\vee\}}(f^m_{MN})\le N\sum_{2\le i\le m}M^i+(N-1)M^m$,

(ii) $C^{\wedge}_{\{\wedge,\vee\}}(f^m_{MN})\le N\sum_{2\le i\le m}M^i$,

(iii) $C^{\vee}_{\{\wedge,\vee\}}(f^m_{MN})\le (N-1)M^m$.

This realization of f^m_{MN} is not the shortest one. The number of $\wedge$-gates can be reduced by grouping the variables first to pairs ($x^1_{h_1l}x^2_{h_2l}$ and $x^3_{h_3l}x^4_{h_4l}$ and $x^5_{h_5l}x^6_{h_6l}$ and so on), then taking groups of 4, 8, 16, ... elements. But the largest term of the number of $\wedge$-gates used remains NM^m. One can show that this improvement reduces the number of $\wedge$-gates for $M>1$ at most by a factor of 2. Therefore we shall not discuss this improvement in detail.

We guess that the bounds of Theorem 3.1(i) and (ii) are essentially optimal. In the following we shall show, that for $m>2$ the bound of Theorem 3.1(iii) is not optimal. We derive the monotone conjunctive normal form of the functions $y_{h_1\cdots h_m}$. If a prime clause has the value 0, the function itself has the value 0 (Definition 2.9). Therefore each prime implicant must be equal to 0. That means that each prime clause contains at least one variable of each prime implicant.

Now let us look at a sum, which arises if one shortens the prime clause. Such a sum is not a clause. Therefore we can conclude: A prime clause is a sum with the following two properties:

(i) it contains at least one variable of each prime implicant.

(ii) each shortening of it fulfils not property (i).

Different prime implicants of $y_{h_1\cdots h_m}$ contain only different variables (Definition 1.1 is the MDNF of $y_{h_1\cdots h_m}$). Therefore a sum is a prime clause of $y_{h_1\cdots h_m}$ iff it contains exactly one variable of each prime implicant of $y_{h_1\cdots h_m}$, that means $\mathrm{PC}(y_{h_1\cdots h_m})=\{x^{i_1}_{h_{i_1}1}\vee x^{i_2}_{h_{i_2}2}\vee\cdots\vee x^{i_N}_{h_{i_N}N}\mid 1\leq i_1,\ldots,i_N\leq m\}$. Therefore the MCNF $(y_{h_1\cdots h_m})$ is

$$y_{h_1\cdots h_m}=\bigwedge_{(i_1,\ldots,i_N)\in\{1,\ldots,m\}^N} x^{i_1}_{h_{i_1}1}\vee\cdots\vee x^{i_N}_{h_{i_N}N}.$$

Take m^2M^2 $\vee$-gates to compute all $x^{i_1}_{h_11}\vee x^{i_2}_{h_22}$ $(1\leq i_1,\ i_2\leq m,\ 1\leq h_1,\ h_2\leq M)$. Following up these ideas it is clear, that

$$\sum_{2\leq i\leq N}(mM)^i=\frac{(mM)^{N+1}-mM}{mM-1}-mM-1\leq\frac{(mM)^{N+1}}{mM-1}$$

$\vee$-gates are sufficient to compute all prime clauses of all $y_{h_1\cdots h_m}$.

Therefore we proved

Lemma 3.2.

$$C^{\vee}_{\{\wedge,\vee\}}(f^m_{MN})\leq\frac{m^{N+1}M^{N+1}}{mM-1}.$$

Exactly as in the case of the number of necessary $\wedge$-gates we can easily improve this result by a small amount. Beyond that we have computed some redundant functions like $x^1_{h_11}\vee x^1_{h_22}$, if $h_1\neq h_2$, since no lengthening of this function is a prime clause of $y_{h_1\cdots h_m}$. In the interesting cases this possibility of improving Lemma 3.2 is unimportant. Therefore we again omit a detailed discussion of this improvement.

The upper bound of Lemma 3.2 does better than the upper bound of Theorem 3.1(iii) for fixed m, $m>N$ and M sufficiently large. In the general case we can decompose $y_{h_1\cdots h_m}$ in the following way: For $k\in\{1,\ldots,\lceil N/(m-1)\rceil-1\}$ we define

$$y^k_{h_1\cdots h_m}:=\bigvee_{(k-1)(m-1)+1\leq l\leq k(m-1)} x^1_{h_1l}\cdots x^m_{h_ml}$$

then the functions $y^k_{h_1\cdots h_m}(1 \le h_1, \ldots, h_m \le M)$ together form the function f^m_{Mm-1}. Finally we define for

$$k = \left\lceil \frac{N}{m-1} \right\rceil \quad y^k_{h_1\cdots h_m} := \bigvee_{(k-1)(m-1)+1 \le l \le N} x^1_{h_1 l} \cdots x^m_{h_m l}.$$

The functions

$$y^k_{h_1\cdots h_m}\left(1 \le h_1, \ldots, h_m \le M, k = \left\lceil \frac{N}{m-1} \right\rceil\right)$$

form the function f^m_{Mp}, where

$$p := N - \left(\left(\left\lceil \frac{N}{m-1} \right\rceil - 1\right)(m-1)\right).$$

It is easy to see, that

$$y_{h_1\cdots h_m} = \bigvee_{1 \le k \le \lceil N/(m-1) \rceil} y^k_{h_1\cdots h_m}.$$

For computing the functions $y^k_{h_1\cdots h_m}$ for each fixed k we make use of the upper bound of Lemma 3.2. Afterwards $\lceil N/(m-1) \rceil - 1$ $\vee$-gates are sufficient for computing each of the M^m functions $y_{h_1\cdots h_m}$.

Therefore we proved

Theorem 3.3. *Let*

$$p := N - \left(\left(\left\lceil \frac{N}{m-1} \right\rceil - 1\right)(m-1)\right),$$

then

$$C^{\vee}_{\{\wedge,\vee\}}(f^m_{MN}) \le \left(\left\lceil \frac{N}{m-1} \right\rceil - 1\right)M^m + \left(\left\lceil \frac{N}{m-1} \right\rceil - 1\right)\frac{m^m M^m}{mM-1} + \frac{m^{p+1}M^{p-1}}{mM-1}.$$

For fixed m and large M we improved the upper bound of Theorem 3.1(iii) from approximately NM^m to about $NM^m/(m-1)$. (This is naturally no improvement if $m=2$ (Paterson [3], Mehlhorn and Galil [2])). This circuit, which computes f^m_{MN}, shows some of the difficulties which arise if one tries to prove good lower bounds for $C^{\vee}_{\{\wedge,\vee\}}(f^m_{MN})$.

In the following we shall consider only the number of $\wedge$-gates. Our main result is

$$C_{\{\wedge,\vee\}}(f^m_{MN}) \ge C^{\wedge}_{\{\wedge,\vee\}}(f^m_{MN}) \ge N\lceil (2/m)M^m \rceil.$$

4. Transformations of monotone circuits, which preserve the property of computing f^m_{MN}

The following theorem is a slight generalization of a theorem, which was proved by Mehlhorn and Galil [2].

Theorem 4.1. *Let S be a monotone circuit computing $g_1, \ldots, g_l$. Let G be a gate of S, where s is computed. Let t, t_1 and t_2 be monoms satisfying the following properties:*

(i) $tt_1 \in \mathrm{I}(s)$,

(ii) $tt_2 \in \mathrm{I}(s)$,

(iii) $\forall_{1 \leq j \leq l} \forall_{\tilde{t}\ \mathrm{monom}} / \tilde{t}tt_1, \tilde{t}tt_2 \in \mathrm{I}(g_j) \Rightarrow \tilde{t}t \in \mathrm{I}(g_j)$.

Let S' be produced from S by computing $s \vee t$ at a new gate G' and replacing some of the edges which leave G, by edges leaving G'. Then the monotone circuit S' computes $g_1, \ldots, g_l$.

Proof. Suppose, that the theorem is false. Then there exists $j \in \{1, \ldots, l\}$, such that g_j is not computed in S'. Let G^* be the gate of S, where g_j is computed. Let g^* be the monotone function, which is computed at G^* as a gate of S', $g^* \neq g_j$. By monotonicity and $s \leq s \vee t$ we conclude $g_j \leq g^*$. Therefore we can choose $\alpha \in \{0, 1\}^n$: $g_j(\alpha) = 0$ and $g^*(\alpha) = 1$. Since the results of S and S' are not the same if the input is α, we conclude (by monotonicity) $s(\alpha) = 0$ and $t(\alpha) = 1$. Let $t^* \in \mathrm{PI}(g^*)$ with the property: $t^*(\alpha) = 1$.

If $t^*t \in \mathrm{I}(g_j)$, we can conclude $g_j(\alpha) = 1$, which is a contradiction. Because of (iii) it remains to show: $t^*tt_i \in \mathrm{I}(g_j)$ $(i = 1, 2)$. If for $\gamma \in \{0, 1\}^n t^*tt_i(\gamma) = 1$ (for $i = 1$ or $i = 2$), we deduce $g^*(\gamma) = 1$ (since $t^* \in \mathrm{PI}(g^*)$) and $s(\gamma) = 1$ (since $tt_i \in \mathrm{I}(s)$) and therefore S and S' compute the same if the input is γ. That means $g_j(\gamma) = g^*(\gamma) = 1$ and therefore $t^*tt_i \in \mathrm{I}(g_j)$.

Now we use this general result for the special function f^m_{MN}. In doing so we use the following notation: x^i_{0l} $(1 \leq i \leq m, 1 \leq l \leq N)$ means the constant 1.

Lemma 4.2. *Let S be a monotone circuit computing f^m_{MN}. Let S be the monotone function, which is computed at the gate G of S and let s have the following properties: For some $l \in \{1, \ldots, N\}$, $i_1, \ldots, i_m, j_1, \ldots, j_m \in \{0, 1, \ldots, M\}$*

$$s_1 := \bigwedge_{1 \leq k \leq m} x^k_{i_k l} \in \mathrm{I}(s) \quad \textit{and} \quad s_2 := \bigwedge_{1 \leq k \leq m} x^k_{j_k l} \in \mathrm{I}(s).$$

Let $s' := \bigwedge_{k \in A} x^k_{i_k l}$, where $A := \{1 \leq k \leq m \mid i_k = j_k\}$. Finally let S' be produced from S by computing $s \vee s'$ at a new gate G' and replacing some of the edges, which leave G, by edges leaving G'. The monotone circuit S' computes f^m_{MN}.

Proof. If we define

$$t := s', \qquad t_1 := \bigwedge_{k \notin A} x^k_{i_k l}, \qquad t_2 := \bigwedge_{k \notin A} x^k_{j_k l},$$

then $tt_1, tt_2 \in \mathrm{I}(s)$. We observe that by definition of A t_1 and t_2 do not have a common variable. If the lemma is false, there exist (Theorem 4.1) a monom $\tilde{t}$ and an output function $y_{h_1 \cdots h_m}$, for which $\tilde{t}tt_1, \tilde{t}tt_2 \in \mathrm{I}(y_{h_1 \cdots h_m})$ and $\tilde{t}t \notin \mathrm{I}(y_{h_1 \cdots h_m})$. Since $\tilde{t}tt_1 \in \mathrm{I}(y_{h_1 \cdots h_m})$, there exists an $l' \in \{1, \ldots, N\}$: $\bigwedge_{1 \le i \le m} x^i_{h_i l'} \subseteq \tilde{t}tt_1$. Since $\tilde{t}t \notin \mathrm{I}(y_{h_1 \cdots h_m})$:

$$\{x^i_{h_i l'} \mid 1 \le i \le m\} \cap t_1 = \{x^i_{h_i l'} \mid 1 \le i \le m\} \cap \{x^k_{i_k l} \mid k \notin A\} \ne \emptyset \Rightarrow l = l'.$$

Therefore $\bigwedge_{1 \le i \le m} x^i_{h_i l} \subseteq \tilde{t}tt_1$ and similarly $\bigwedge_{1 \le i \le m} x^i_{h_i l} \subseteq \tilde{t}tt_2$. Again since

$$\tilde{t}t \notin \mathrm{I}(y_{h_1 \cdots h_m}), \quad \bigwedge_{1 \le i \le m} x^i_{h_i l} \not\subseteq \tilde{t}t \Rightarrow \exists\, i' \in \{1, \ldots, m\}: x^{i'}_{h_{i'} l} \notin \tilde{t}t \Rightarrow t_1 \quad \text{and} \quad t_2$$

have the variable $x^{i'}_{h_{i'} l}$ in common.

This contradiction proves the lemma.

5. Lower bounds for the monotone complexity of f^m_{MN}

Definition 5.1. (i) For $(h_1, \ldots, h_m) \in \{1, \ldots, M\}^m$ we denote by $Q_{h_1 \cdots h_m}$ the set of monotone functions, for which $\bigwedge_{1 \le i \le m} x^i_{h_i 1}$ is a prime implicant.

(ii) Let S be a monotone circuit computing f^m_{MN} and $(h_1, \ldots, h_m) \in \{1, \ldots, M\}^m$. We denote by $S(Q_{h_1 \cdots h_m})$ the set of gates of S with the property, that the function which is computed at G is an element of $Q_{h_1 \cdots h_m}$, but the functions which are computed at the two direct predecessors of G (gates or inputs) are not elements of $Q_{h_1 \cdots h_m}$.

As indicated in the introduction we shall make the assumption that certain functions are given for free. In this chapter we shall consider only monotone circuits S with the property (*):

(*) Besides the variables $x^i_{h_i l}$ $(1 \le i \le m, 1 \le h_i \le M, 1 \le l \le N)$ also all monoms of less than m variables are inputs of the circuit.

Note: For $m = 2$ this assumption is empty.

$C^*_{\{\wedge,\vee\}}$, $C^{*\wedge}_{\{\wedge,\vee\}}$ and $C^{*\vee}_{\{\wedge,\vee\}}$ are the complexity measures $C_{\{\wedge,\vee\}}$, $C^{\wedge}_{\{\wedge,\vee\}}$ and $C^{\vee}_{\{\wedge,\vee\}}$ restricted to circuits with the property (*). It is easy to see that for all monotone functions s

$$C^*_{\{\wedge,\vee\}}(s) \le C_{\{\wedge,\vee\}}(s), \qquad C^{*\wedge}_{\{\wedge,\vee\}}(s) \le C^{\wedge}_{\{\wedge,\vee\}}(s) \quad \text{and} \quad C^{*\vee}_{\{\wedge,\vee\}}(s) = C^{\vee}_{\{\wedge,\vee\}}(s).$$

Let S be a monotone circuit with property (*) computing f^m_{MN}. All inputs are not elements of any $Q_{h_1 \cdots h_m}$. On the other side $y_{h_1 \cdots h_m} \in Q_{h_1 \cdots h_m}$. Therefore for all $(h_1, \ldots, h_m) \in \{1, \ldots, M\}^m$: $S(Q_{h_1 \cdots h_m}) \ne \emptyset$. The aim of the following considerations is to give lower bounds to

$$\#\Big(\bigcup_{(h_1, \ldots, h_m) \in \{1, \ldots, M\}^m} S(Q_{h_1 \cdots h_m})\Big)$$

and to eliminate as many as possible of these gates by fixing the variables $x^i_{h_i 1}$ ($1 \le i \le m$, $1 \le h_i \le M$). At the same time these variables should be fixed in such a way that the remaining circuit computes f^m_{MN-1}.

The following lemma shows that we need consider only $\wedge$-gates.

Lemma 5.2. *$G \in S(Q_{h_1 \cdots h_m}) \Rightarrow G$ is an $\wedge$-gate.*

Proof. Let s be the function computed at G and s_1 and s_2 the functions computed at the two direct predecessors of G. $\bigwedge_{1 \le i \le m} x^i_{h_i 1}$ is a prime implicant of s but not of s_1 or s_2. If G is an $\vee$-gate, $s = s_1 \vee s_2$ and $\mathrm{PI}(s) \subseteq \mathrm{PI}(s_1) \cup \mathrm{PI}(s_2)$ yields a contradiction.

Main lemma 5.3. *There exists a monotone circuit S with property (*) and the following properties:*

(i) *S computes f^m_{MN}.*
(ii) *The number of $\wedge$-gates of S is $C^{*\wedge}_{\{\wedge,\vee\}}(f^m_{MN})$.*
(iii) $\forall_{G \text{ gate of } S} \; \#\{(h_1, \ldots, h_m) \in \{1, \ldots, M\}^m \mid G \in S(Q_{h_1 \cdots h_m})\} \le 1$.

Proof. We start with a monotone circuit S with property (*), (i) and (ii). The existence of S is clear. If (iii) is fulfilled, the lemma is proved. Otherwise let $G_1, \ldots, G_{C^{*\wedge}_{\{\wedge,\vee\}}(f^m_{MN})}$ be the $\wedge$-gates of S. We assume that the $\wedge$-gates are in their topological order, i.e. for $i < k$ there is no directed path from G_k to G_i. We have shown (Lemma 5.2) that these gates are the only gates, where (iii) may not hold.

$J := \min\{1 \le k \le C^{*\wedge}_{\{\wedge,\vee\}}(f^m_{MN}) \mid \text{(iii) is not fulfilled for } G_k\}$. Our aim is to build another monotone circuit S' with property (*), (i) and (ii), such that either for $G'_1, \ldots, G'_J$ (the leading J $\wedge$-gates of S') property (iii) is fulfilled or for $G'_1, \ldots, G'_{J-1}$ property (iii) is fulfilled and $\mathrm{L}(G'_J) < \mathrm{L}(G_J)$ (see Definition 2.12). If this assertion is proved, we may proceed in the same way and shall obtain after a finite number of steps (since $\mathrm{L}(G) \in \mathbb{N}_0$) a monotone circuit with the desired properties.

Now we are going to prove the assertion:

We conclude (from the definition of J) the existence of $(h_1, \ldots, h_m)$, $(h'_1, \ldots, h'_m) \in \{1, \ldots, M\}^m$: $(h_1, \ldots, h_m) \ne (h'_1, \ldots, h'_m)$ and $G_J \in S(Q_{h_1 \cdots h_m}) \cap S(Q_{h'_1 \cdots h'_m})$. Let s denote the function computed at G_J and let G^1 and G^2 be the two direct predecessors of G_J, where s_1 and s_2 are computed. Therefore $s = s_1 s_2$. By Definition 5.1 it follows, that

$$\bigwedge_{1 \le i \le m} x^i_{h_i 1}, \quad \bigwedge_{1 \le i \le m} x^i_{h'_i 1} \in \mathrm{PI}(s), \qquad \bigwedge_{1 \le i \le m} x^i_{h_i 1}, \quad \bigwedge_{1 \le i \le m} x^i_{h'_i 1} \notin \mathrm{PI}(s_1) \cup \mathrm{PI}(s_2).$$

Furthermore

$$\bigwedge_{1\le i\le m} x^i_{h_i1},\quad \bigwedge_{1\le i\le m} x^i_{h'_i1} \in \mathrm{I}(s_1)\cap \mathrm{I}(s_2).$$

Proof. Suppose for example $\bigwedge_{1\le i\le m} x^i_{h_i1} \notin \mathrm{I}(s_1)$. Fix $x^i_{h_i1}=1$, all other variables fix to 0, s_1 computes 0, therefore s computes 0, which is a contradiction ($\bigwedge_{1\le i\le m} x^i_{h_i1} \in \mathrm{PI}(s)$).

For monotone functions $\tilde{s}$ the following is true: $m\in \mathrm{I}(\tilde{s}) \Rightarrow \exists\ m'\subseteq m\ m'\in \mathrm{PI}(\tilde{s})$. Therefore there exist $t_1, t'_1 \in \mathrm{PI}(s_1)$ and $t_2, t'_2\in \mathrm{PI}(s_2)$:

$$t_1 \subsetneq \bigwedge_{1\le i\le m} x^i_{h_i1},\qquad t'_1 \subsetneq \bigwedge_{1\le i\le m} x^i_{h'_i1},\qquad t_2 \subsetneq \bigwedge_{1\le i\le m} x^i_{h_i1},\qquad t'_2 \subsetneq \bigwedge_{1\le i\le m} x^i_{h'_i1}.$$

We claim, that $t_1t_2 = \bigwedge_{1\le i\le m} x^i_{h_i1}$. "$\subseteq$" is obvious. "$\supseteq$": Suppose $t_1t_2=\bigwedge_{i\in A} x^i_{h_i1}$ for some $A\subsetneq\{1,\dots,m\}$. Let $x^i_{h_i1}=1$ for $i\in A$, then $t_1=1$ and $t_2=1\Rightarrow s_1=1$ and $s_2=1\Rightarrow s=1$ in contradiction to $\bigwedge_{1\le i\le m} x^i_{h_i1}\in \mathrm{PI}(s)$. Similarly $t'_1t'_2=\bigwedge_{1\le i\le m} x^i_{h'_i1}$. We again use the following notation: For $1\le i\le m$, $1\le l\le N$ x^i_{0l} means 1. There exist $k_{i,1}, k_{i,2}\in\{0, h_i\}$ and $k'_{i,1}, k'_{i,2}\in\{0, h'_i\}$:

$$t_1=\bigwedge_{1\le i\le m} x^i_{k_{i,1}1},\qquad t'_1=\bigwedge_{1\le i\le m} x^i_{k'_{i,1}1},\qquad t_2=\bigwedge_{1\le i\le m} x^i_{k_{i,2}1},$$

$$t'_2=\bigwedge_{1\le i\le m} x^i_{k'_{i,2}1}.$$

Case 1: $t_1\ne t'_1$. Let $A:=\{1\le j\le m\mid k_{j,1}=k'_{j,1}\}\Rightarrow A\subsetneq\{1,\dots,m\}$. Therefore $s':=\bigwedge_{i\in A} x^i_{k_{i,1}1}$ is an input of the circuit, because it is a monom of less than m variables. Let S' be produced from S by computing with one additional $\vee$-gate G' and no additional $\wedge$-gate $s_1\vee s'$ and by replacing the edge $G^1\to G_J$ by the edge $G'\to G_J$. We have shown (Lemma 4.2), that S' computes f^m_{MN}. Because of the construction of S', S' fulfils property (*), (i) and (ii). The $\wedge$-gates of S' will be labelled

$$G'_1,\dots, G'_{C^{*\wedge}_{\{\wedge,\vee\}}(f^m_{MN})},$$

such that G_i corresponds to G'_i in a natural way. Property (iii) remains correct for $G'_1,\dots,G'_{J-1}$. Perhaps property (iii) is fulfilled for G'_J too. In any case we shall show: $\mathrm{L}(G'_J)<\mathrm{L}(G_J)$.

$$\mathrm{L}(G'_J)=\mathrm{L}(s_1\vee s')+\mathrm{L}(s_2)\quad\text{and}\quad \mathrm{L}(G_J)=\mathrm{L}(s_1)+\mathrm{L}(s_2).$$

If $\mathrm{PI}(s_1)=\{t_1, t'_1, g_1,\dots,g_r\}$, then $\mathrm{PI}(s_1\vee s')\subseteq\{s', g_1,\dots,g_r\}$, since $s'\subseteq t_1$ and $s'\subseteq t'_1$. Moreover $t_1\ne t'_1$, so we can conclude

$$\mathrm{L}(s_1\vee s')<\mathrm{L}(s_1)\Rightarrow \mathrm{L}(G'_J)<\mathrm{L}(G_J).$$

In this case the assertion is proved.

Case 2: $t_1=t'_1$.

$$t_1=t'_1=\bigwedge_{i\in A} x^i_{h_i1}=\bigwedge_{i\in A} x^i_{h'_i1}\quad\text{for some } A\subsetneq\{1,\dots,m\}.$$

Since $(h_1,\dots,h_m)\neq(h'_1,\dots,h'_m)$, there exists $j\in\{1,\dots,m\}$: $h_j\neq h'_j$ and $j\notin A$. Since $t_1t_2=\bigwedge_{1\leq i\leq m}x^i_{h_i1}$, we conclude $x^j_{h_j1}\in t_2$ and similarly $x^j_{h'_j1}\in t'_2$.

Therefore $t_2\neq t'_2$ and we can prove the assertion in the same way as in Case 1. The proof of the assertion completes the proof of the lemma.

Now we consider a monotone circuit S with property (*), (i), (ii) and (iii). Since

$$S(Q_{h_1\cdots h_m})\neq\emptyset:\quad \#\Big(\bigcup_{(h_1,\dots,h_m)\in\{1,\dots,M\}^m} S(Q_{h_1\cdots h_m})\Big)\geq M^m.$$

We examine a gate $G\in S(Q_{h_1\cdots h_m})$. Similarly to the proof of Lemma 5.3 there exist $t_1\in \mathrm{PI}(s_1)$ and $t_2\in\mathrm{PI}(s_2)$:

$$t_1\subsetneq\bigwedge_{1\leq i\leq m}x^i_{h_i1},\qquad t_2\subsetneq\bigwedge_{1\leq i\leq m}x^i_{h_i1}\quad\text{and}\quad t_1t_2=\bigwedge_{1\leq i\leq m}x^i_{h_i1}.$$

Now we try to fix the variables $x^i_{h_i1}$ $(1\leq i\leq m, 1\leq h_i\leq M)$ to fulfil the aims we have discussed at the beginning of this chapter. We distinguish m cases:

Case k: $(1\leq k\leq m)$ $x^k_{j1}:=0$ $(1\leq j\leq M)$ and $x^i_{j1}:=1$ $(1\leq j\leq M, 1\leq i\leq m, i\neq k)$. The remaining circuit $\tilde S_k$ computes, instead of $y_{h_1\cdots h_m}$

$$\tilde y_{h_1\cdots h_m}:=\bigvee_{2\leq l\leq N}x^1_{h_1l}\cdots x^m_{h_ml}$$

and therefore the function f^m_{MN-1}. G can be eliminated in $\tilde S_k$, if $x^k_{h_k1}\notin t_1$ or $x^k_{h_k1}\notin t_2$. In either case t_1 or t_2 becomes 1, therefore s_1 or s_2 becomes 1 and one input of G becomes constant.

We claim that t_1 and t_2 contain at most $m-2$ equal variables. This is easy to see. $\mathrm{L}(t_1)\leq m-1$ and $\mathrm{L}(t_2)\leq m-1$. If t_1 and t_2 have more than $m-2$ variables in common, we conclude $t_1=t_2$. So $t_1t_2=t_1\subsetneq\bigwedge_{1\leq i\leq m}x^i_{h_i1}$, which is a contradiction. (*Note*: If $m=2$, t_1 and t_2 have no common variable.)

Therefore G is eliminated in at least two of the m cases above. We can conclude that there is a $k\in\{1,\dots,m\}$, such that fixing the variables $x^k_{j1}:=0$ and $x^i_{j1}:=1$ $(i\neq k)$ causes the elimination of at least $\lceil 2m^{-1}M^m\rceil$ $\wedge$-gates.

Summarizing we have proved $C^{*\wedge}_{\{\wedge,\vee\}}(f^m_{MN})\geq C^{*\wedge}_{\{\wedge,\vee\}}(f^m_{MN-1})+\lceil 2m^{-1}M^m\rceil$. Thus we have proved

Theorem 5.4. *For all $m, M, N\in\mathbb{N}$, where $m\geq 2$,*

$$C_{\{\wedge,\vee\}}(f^m_{MN})\geq C^{\wedge}_{\{\wedge,\vee\}}(f^m_{MN})\geq C^{*\wedge}_{\{\wedge,\vee\}}(f^m_{MN})\geq N\lceil 2m^{-1}M^m\rceil.$$

Let us consider the very special case of the Boolean matrix product of two $M\times M$-matrices: $m=2$ and $M=N$. Then $N\lceil 2m^{-1}M^m\rceil=M^3$ and Corollary 5.5 is a direct consequence of Theorem 5.4.

Corollary 5.5. *M^3 $\wedge$-gates are necessary to compute the Boolean matrix product of two $M\times M$-matrices.*

This result was proved earlier by Paterson [3] and Mehlhorn and Galil [2]. Pratt [4] proved the necessity of $\frac{1}{2}M^3$ $\wedge$-gates.

Remark. Savage [5] extended the results of Pratt, Paterson and Mehlhorn and Galil and examined the monotone complexity of disjoint monotone bilinear forms. One can easily extend the above considerations in order to evaluate the monotone complexity of disjoint monotone multilinear forms, whose definition is a natural generalization of the Savage definition of disjoint monotone bilinear forms. The functions $y_{h_1 \cdots h_m}$ $(1 \leq h_1, \ldots, h_m \leq M)$ form a set of disjoint monotone multilinear forms. (For details see Wegener [7].)

6. Switching functions whose monotone complexity is nearly quadratic

By using Theorem 5.4 it is easy to deduce our main result, which was indicated in the abstract.

We shall introduce the following notation, which we shall use in Section 7 too. If $g:\{0,1\}^n \to \{0,1\}^m$ is defined, we also denote the following function $g':\{0,1\}^{n+n'} \to \{0,1\}^{m+m'}$ $(n', m' \in \mathbb{N}_0)$ with g:

$$\forall_{1 \leq i \leq m} \; g'_i(x_1, \ldots, x_{n+n'}) := g_i(x_1, \ldots, x_n), \qquad \forall_{m < i \leq m+m'} \; g'_i \equiv 0.$$

Definition 6.1. For $n \geq 4$ let $m(n) := \lfloor \log n \rfloor$, $M(n) := 2$, $N(n) := \lfloor n/(2 \log n) \rfloor$ and $h_n := f^{m(n)}_{M(n)N(n)}(h_n : \{0,1\}^n \to \{0,1\}^n)$.

Theorem 6.2. *The monotone functions h_n are explicitly defined and $\Omega(n^2 \log^{-2} n) = C_{\{\wedge,\vee\}}(h_n) = O(n^2 \log^{-1} n)$.*

Proof. The theorem is a direct consequence of Theorem 5.4 and Theorem 3.1.

This is a considerable improvement of the previously largest known lower bounds for a series of explicitly defined monotone functions, which are $\Omega(n^{3/2})$ bounds for example for the Boolean matrix product.

7. Monotone complexity – complexity over a complete basis

Definition 7.1. Let m be even and fixed. For $n \in \mathbb{N}$ let

$$M(n) := \max\{k \in \mathbb{N} \mid mk^m \leq n\} \quad \text{and} \quad g_n^m := f^m_{M(n)M(n)^{m-1}}(g_n^m : \{0,1\}^n \to \{0,1\}^n).$$

We shall show the following theorem, which shows the importance of the negation. Pratt [4] proved this theorem for the special case $m = 2$.

Theorem 7.2. *For m even and fixed*

$$C_{\{\wedge,\vee\}}(g_n^m)/C_{\{\wedge,\vee,-\}}(g_n^m) = \Omega(n^{(1/2)\log_2 8/7}\log^{-2} n).$$

Proof. It is an easy consequence of Theorem 5.4, that $C_{\{\wedge,\vee\}}(g_n^m) = \Omega(n^{2-m^{-1}})$. So it remains to show, that

$$C_{\{\wedge,\vee,-\}}(g_n^m) = O(n^{(1/2)\log_2 7+1/2-m^{-1}}\log^2 n),$$

since $2-m^{-1}-1/2\log_2 7-1/2+m^{-1} = 3/2-1/2\log_2 7 = 1/2\log_2 8/7$. Strassen [6] has shown, that $O(n^{\log_2 7})$ real additions, subtractions and multiplications are sufficient to compute the matrix product of two $n \times n$-matrices with elements in $\mathbf{R}$. Fischer and Meyer [1] used this result and have shown, that $O(n^{\log_2 7}\log^2 n)$ gates of the basis $\{\wedge, \vee, -\}$ are sufficient to compute the Boolean matrix product of two $n \times n$-matrices.

Now we present a circuit over $\{\wedge, \vee, -\}$, which fulfils the desired properties. First one realizes for all $1 \le h_1, \ldots, h_{m/2} \le M(n)$ and $1 \le l \le M(n)^{m-1}$ $\bigwedge_{1\le i\le m/2} x^i_{h_i l}$ and for all $1 \le h_{m/2+1}, \ldots, h_m \le M(n)$ and $1 \le l \le M(n)^{m-1}$ $\bigwedge_{m/2<i\le m} x^i_{h_i l}$. For this $2(m/2-1)M(n)^{m/2}M(n)^{m-1}$ $\wedge$-gates are sufficient. Now the task to compute g_n^m is the task to compute the Boolean matrix product of the $M(n)^{m/2} \times M(n)^{m-1}$-matrix

$$Z^1 = (z^1_{(h_1,\ldots,h_{m/2})l} = \bigwedge_{1\le i\le m/2} x^i_{h_i l})_{\substack{1\le h_1,\ldots,h_{m/2}\le M(n)\\ 1\le l\le M(n)^{m-1}}}$$

and the $M(n)^{m-1} \times M(n)^{m/2}$-matrix

$$Z^2 = (z^2_{l(h_{m/2+1},\ldots,h_m)} = \bigwedge_{m/2<i\le m} x^i_{h_i l})_{\substack{1\le l\le M(n)^{m-1}\\ 1\le h_{m/2+1},\ldots,h_m\le M(n)}}.$$

This is true, because the result of the matrix product is the $M(n)^{m/2} \times M(n)^{m/2}$-matrix

$$Y = (y_{(h_1,\ldots,h_{m/2})(h_{m/2+1},\ldots,h_m)} = \bigvee_{1\le l\le M(n)^{m-1}} \bigwedge_{1\le i\le m} x^i_{h_i l})$$

$(1 \le h_1, \ldots, h_{m/2} \le M(n), 1 \le h_{m/2+1}, \ldots, h_m \le M(n))$.

For computing this matrix product we decompose each of the matrices Z^1 and Z^2 to $M(n)^{m/2-1}M(n)^{m/2} \times M(n)^{m/2}$-matrices, whereby for the jth partial matrix we take those variables, for which $l \in \{(j-1)M(n)^{m/2}+1, \ldots, jM(n)^{m/2}\}$. We multiply the jth partial matrix of Z^1 with the jth partial matrix of Z^2 (using the results of Fischer and Meyer and Strassen) with $O((M(n)^{m/2})^{\log_2 7}(\log M(n)^{m/2})^2)$ gates of the basis $\{\wedge, \vee, -\}$. All together $O(M(n)^{m/2-1}M(n)^{(m/2)\log_2 7}(\log M(n)^{m/2})^2)$ gates are sufficient. As result we get $M(n)^{m/2-1}M(n)^{m/2} \times M(n)^{m/2}$-matrices $Y_1, \ldots, Y_{M(n)^{m/2-1}}$. It is easy to see, that $Y = Y_1 \vee \cdots \vee Y_{M(n)^{m/2-1}}$. (If $A = (a_{ij})$ and

$B = (b_{ij})$ we define $A \vee B = (a_{ij} \vee b_{ij})$.) Therefore another $M(n)^m (M(n)^{m/2-1} - 1)$ $\vee$-gates are sufficient to compute Y. Combining the results we get

$$\begin{aligned} C_{\{\wedge,\vee,-\}}(g_n^m) &= O(2(m/2-1)M(n)^{m/2}M(n)^{m-1} \\ &\quad + M(n)^{m/2-1}M(n)^{(m/2)\log_2 7}(\log M(n)^{m/2})^2 \\ &\quad + M(n)^m (M(n)^{m/2-1} - 1)) \\ &= O((M(n)^m)^{(1/2)\log_2 7 + 1/2 - m^{-1}}(\log M(n)^m)^2) \\ &= O(n^{(1/2)\log_2 7 + 1/2 - m^{-1}} \log^2 n), \end{aligned}$$

since m is fixed and since $M(n)^m \leqslant nm^{-1} \leqslant n$.

Thus the complexity gap between monotone circuits and circuits over a complete basis is for the functions g_n^m (m even and fixed) at least as large as the well-known gap for the Boolean matrix product.

References

[1] M.J. Fischer and A.R. Meyer, Boolean matrix multiplication and transitive closure, *IEEE Conference Record of the Twelfth Annual Symposium on Switching and Automata Theory* (1971) 129–131.

[2] K. Mehlhorn and Z. Galil, Monotone switching circuits and Boolean matrix product, *Computing* **16** (1976) 99–111.

[3] M.S. Paterson, Complexity of monotone networks for Boolean matrix product, *Theoret. Comput. Sci.* **1** (1975) 13–20.

[4] V.R. Pratt, The power of negative thinking in multiplying Boolean matrices, *SIAM J. Comput.* **4** (1975) 326–330.

[5] J.E. Savage, *The Complexity of Computing* (John Wiley, New York, 1976).

[6] V. Strassen, Gaussian elimination is not optimal, *Numer. Math.* **13** (1969) 354–356.

[7] I. Wegener, Boolesche Funktionen, deren monotone Komplexität fast quadratisch ist, Dissertation, Universität Bielefeld (1978).