$\exists$ a monomial t' and a output function $y_{h_1,h_2\dots h_m}$ with

- $t'tt_1, t'tt_2 \in IM(y_{h_1,h_2\dots h_m})$ but $t't \notin IM(y_{h_1,h_2\dots h_m})$.

Then
$t'tt_1$ contains a prime implicant $(h_1,\dots,h_m,e')$ but
$t't$ does not contain the prime implicant $(h_1,\dots,h_m,e')$.

By construction, it holds $e'=e$.

The same holds with respect to $t'tt_2$.

Altogether, we obtain

- $(h_1,h_2,\dots,h_m,e)$ is a submonomial of $t'tt_1$ and also a submonomial of $t'tt_2$ but not a submonomial of $t't$.

$\Longrightarrow$

t_1 and t_2 have a variable in common.

But this contradicts the definition of A. ∎

A monomial m is <u>useful</u> for an output of f^m_{MN} if m is a submonomial of a prime

implicant of that output.

Lemma 3.3 ⇒

If the function $g = res_\beta(u)$ includes several useful monomials of type ℓ then we can replace g in β by gvt where t is the common part of all useful monomials of type ℓ.

Goal:

Application of Lemma 3.3 without additional cost.

For doing this, we need the following properties:

1) $\vee$-gates are not counted.
2) All monomials of $< m$ variables are given for free; i.e., are given as additional inputs of the network.

A monotone network fulfilling both properties is called a $*$-network. $C^*_{\wedge_m}$ is the associated complexity measure.

Given any $*$-network β for f^m_{MN}, we wish to transform β into a so-called <u>standard</u> $*$-network of the same complexity by applying Lemma 3.3.

For doing this, we consides the gates in β in any topological order. Let u be the current considered gate and $g := res_\beta(u)$.

For $1 \le \ell \le N$ let

$$t_\ell := \begin{cases} 0 & g \text{ contains at most one useful monomial of type } \ell \\ \text{common part of all useful monomials of type } \ell & \text{otherwise} \end{cases}$$

Without additional cost, we replace

$$g \quad \text{by} \quad g \vee t_1 \vee t_2 \vee \dots \vee t_N$$

$\hookrightarrow$

Altogether, we obtain a $*$-network β' for f^m_{MN} such that

(*)
- all functions computed at the ingoing and outgoing edges of the $\wedge$-gates have at most one useful monomial of type ℓ as prime implicant.

Theorem 3.7

Let $m \ge 2$. Then

$$C_{\Omega_m}(f^m_{MN}) \ge C_\wedge(f^m_{MN}) \ge C^*_{\Omega_m}(f^m_{MN}) > \tfrac{1}{2} N \cdot M^m$$

Corollary 3.1

For $n \geq 4$ let $m(n) := \lfloor \log n \rfloor$, $M(n) := 2$, $N(n) := \lfloor \frac{n}{2 \log n} \rfloor$ and $h_n := f^{m(n)}_{M(n)N(n)}$. Then the function h_n depends on at most n variables and has at most n output functions. Furthermore, $C_{\Omega_m}(h_n) = \Omega\left(\frac{n^2}{\log n}\right)$.

Proof:

Exercise ∎

Proof of Theorem 3.7:

Let β be an optimal standard $*$-network for f^m_{MN}.

Goal:

For each $\wedge$-gate v in β, definition of a **value function**

$$c_v : P(M(f^m_{MN})) \longrightarrow [0,1]$$

such that

$$c(v) := \sum_{1 \leq h_1, h_2, \ldots, h_m \leq M} \sum_{1 \leq \ell \leq N} c_v(h_1, h_2, \ldots, h_m, \ell) \leq$$

Properties:

a) $c_v(h_1, h_2, \ldots, h_m, \ell)$ is an estimate of the

contribution of the $\wedge$-gate v to the computation of the prime implicant $(h_1, \dots, h_m, \ell)$.

b) Each gate contributes to all prime implicants at most the value one.

$\Rightarrow$

For an optimal $*$-network β there holds

$$c(\beta) := \sum_{v \text{ } \wedge\text{-gate in } \beta} c(v) \leq C_{\wedge}(\beta) = C^*_{\Omega_m}(f^m_{MN}).$$

This means that $c(\beta)$ is a lower bound for $C^*_{\Omega_m}(f^m_{MN})$. The value function will have the following property:

Claim 1:

$$c(h_1, h_2, \dots, h_m, \ell) := \sum_{v \text{ } \wedge\text{-gate in } \beta} c_v(h_1, h_2, \dots, h_m, \ell) > \frac{1}{2}.$$

Before defining the value function and proving the claim we shall terminate the proof of the theorem.

Claim 1 $\Rightarrow$

$$C^*_{\Omega_m}(f^m_{MN}) \geq \sum_{1 \leq h_1, \dots, h_m \leq M} \sum_{1 \leq \ell \leq N} c(h_1, \dots, h_m, \ell)$$

$$> \frac{1}{2} \cdot N \cdot M^m.$$

Definition of the value function

Let v be an $\wedge$-gate in an optimal standard $*$-network β for f^m_{MN}. Let

g', g'' be the functions of the ingoing edges of v

$g := res_\beta(v)$.

Idea:

The value function c_v assigns to the $\wedge$-gate v a positive value for the prime implicant t in $PIM(f^m_{MN})$ if

- $t \in PIM(g)$ and $t \notin PIM(g')$ or
- $t \in PIM(g)$ and $t \notin PIM(g'')$.

Question:

In these cases, which value > 0 should be chosen?

To define these values let

$i_1, i_2, \ldots, i_{q'} \in \{1, 2, \ldots, N\}$ be those types such that

$\exists\, t_e \in PIM(f^m_{MN})$ of type i_e with

$t_e \in PIM(g)$ but $t_e \notin PIM(g')$.

Furthermore, let

$j_1, j_2, \dots, j_{q''} \in \{1,2,\dots,N\}$ be those types such that

$\exists\, t_e \in PIM(f^m_{MN})$ of type j_e with $t_e \in PIM(g)$ but $t_e \notin PIM(g'')$

Then we define for $t \in PIM(f^m_{MN})$

$$c'_v(t) := \begin{cases} \frac{1}{2q'} & \text{if } t \in PIM(g) \text{ and } t \notin PIM(g') \\ 0 & \text{otherwise} \end{cases}$$

$$c''_v(t) := \begin{cases} \frac{1}{2q''} & \text{if } t \in PIM(g) \text{ and } t \notin PIM(g'') \\ 0 & \text{otherwise} \end{cases}$$

Then $c_v(t)$ is defined by

$$c_v(t) := c'_v(t) + c''_v(t).$$

Then we obtain because of property (*):

$$c'(v) = \sum_{1 \le h_1, \dots, h_m \le M} \; \sum_{1 \le e \le N} c'_v(h_1, \dots, h_m, e)$$

$$= q' \cdot \frac{1}{2q'} = \frac{1}{2}$$

Note that in an optimal standard *-network, at an $\wedge$-gate at most one prime implicant of each type can have a value > 0.

and also

$$C''(v) = \sum_{1 \le h_1, \dots, h_m \le M} \sum_{1 \le \ell \le N} C''_v(h_1, \dots, h_m, \ell)$$

$$= q'' \cdot \frac{1}{2q''} = \frac{1}{2}.$$

$$\Rightarrow$$

$$C(v) = C'(v) + C''(v) = 1.$$

It remains to prove Claim 1.

Proof of Claim 1:

Consider the prime implicant $t := (h_1, h_2, \dots, h_m, \ell)$ and the corresponding output

$$y_t := y_{h_1 h_2 \dots h_m}.$$

Let $\beta(t)$ be that subnetwork of β which contains the following gates and inputs.

gate v is contained in $\beta(t)$ iff

there is a path P in β from v to the output y_t and t is a prime implicant of all functions computed on P (inclusive $res_{\beta}(v)$).

Additionally, the inputs of the gates in $\beta(t)$ are contained in $\beta(t)$ as well.

Properties:

1, For each input function g of $\beta(t)$ holds $t \notin PIM($

2) t is prime implicant of each function $res_{\beta}(v)$ where v is a gate in $\beta(t)$.

3) Let v be a gate in $\beta(t)$ with both direct predecessors of v are inputs of $\beta(t)$

$\Rightarrow$

v is an $\wedge$-gate (otherwise $t \notin PIM(res_{\beta(t)}(v))$.)

4) If an input of $\beta(t)$ is input of an $\wedge$-gate of $\beta(t)$ then a proper shortening of t is a prime implicant of that input.

Let $s_1, s_2, \ldots, s_D$ be those inputs of $\beta(t)$ which are input of an $\wedge$-gate in $\beta(t)$.

Let $v(i)$ be an $\wedge$-gate in $\beta(t)$ with input s_i.

Let

$$c^*(v(i)) := \begin{cases} c'(v(i)) & \text{if } s_i \text{ is the first input of } v(i) \\ c''(v(i)) & \text{if } s_i \text{ is the second input of } v(i) \end{cases}$$

Properties 1 and 2 $\Rightarrow$ $c^*(v(i)) > 0$.

For $1 \le i \le D$ let $b_i := c^*(v(i))$.

W.l.o.g. we can assume $b_1 \ge b_2 \ge \ldots \ge b_D$.

Note $\sum_{i=1}^{D} b_i > \frac{1}{2} \Rightarrow$ Claim 1.

We shall prove $b_1 + b_2 + \ldots + b_D > \frac{1}{2}$.

Choose $w_i \in PIM(s_i)$ such that a proper prolongation w_i^* of w_i is contained in

$$PIM(res(v(i))) \cap PIM(f_{MN}^m)$$

and the type of w_i^* is different to the types of $w_1^*, w_2^*, \ldots, w_{i-1}^*$.

We can always choose $w_1^* = t$. We distinguish two cases

Case 1:

The choice of w_i according to the rules above is impossible for an $i \leq D$.

$\Rightarrow$

$c^*(v(i))$ is positiv for $\leq (i-1)$ prime implicants

Hence, by the definition of the value function

$$b_i \geq (2(i-1))^{-1}$$

Because of $b_1 \geq b_2 \geq \ldots \geq b_D$, we obtain

$$\sum_{j=1}^{D} b_j \geq i\, b_i \geq i(2(i-1))^{-1} > \frac{1}{2}.$$

Case 2:

The choice of $w_1, w_2, \ldots, w_D$ according to the rules above is possible.

Construction $\Rightarrow$ $w_i \in PIM(s_i)$ for $1 \leq i \leq D$

$\Rightarrow$ $w_1 w_2 \dots w_D \leq s_1 s_2 \dots s_D$.

Construction of $\beta(t)$ $\Rightarrow$

$\forall a$ with $s_i(a) = 1$ for $1 \leq i \leq D$
there holds $y_t(a) = 1$

(this can easily be shown by induction).

$\Rightarrow$

$w_1 w_2 \dots w_D \leq y_t$.

All variables in w_i are of type ℓ_i and $\ell_1, \ell_2, \dots, \ell_D$ are pairwise different.

Each w_i is a proper shortening of w_i^* and $c^*(v(i))(w_i^*) > 0$.

$\Rightarrow$

w_i contains $\leq m-1$ variables

$\Rightarrow$

$w_1 w_2 \dots w_D \notin IM(y_t)$

a contradiction

Hence, Case 2 cannot occur.

■

3.4 The Boolean convolution

References

- Nicholas Pippenger, Leslie G. Valiant, Shifting graphs and their applications, JACM 23 (1976), 423 – 432.
- Edmund A. Lamagna, The complexity of monotone networks for certain bilinear forms, routing problems, sorting and merging, IEEE Trans. Comput. 28 (1979), 773 – 782.
- Norbert Blum, An $\Omega(n^{4/3})$ lower bound on the monotone network complexity of the n^{th} degree convolution, TCS 36 (1985), 59 – 69.
- Jürgen Weiß, An $n^{\frac{3}{2}}$ lower bound on the monotone network complexity of the Boolean convolution, Information and Control 59 (1983), 184 – 188.
- M. I. Grinchuk, I. S. Sergeev, Thin circulant matrices and lower bounds on the complexity of some Boolean operators, arXiv: 1701.08557v [cs. CC] 30. Jan 2017.

Let $A = \{a_0, a_1, \dots, a_{n-1}\}$, $B = \{b_0, b_1, \dots, b_{n-1}\}$ be two disjoint sets of n variables. Then we define the n-th degree convolution C_n as follows:

$C_n = (c_0, c_1, \dots, c_{2n-2}) : \{0,1\}^{2n} \longrightarrow \{0,1\}^{2n-1}$, where

$$c_k := \bigvee_{i+j=k} a_i b_j \qquad 0 \le k \le 2n-2.$$

All sets of monotone functions considered so far (Boolean sums, Boolean matrix multiplication, generalized Boolean matrix multiplication) have disjunctive properties that Boolean convolution does not have. To formalize this, we need some notations.

Let $A = \{a_0, a_1, \dots, a_p\}$ and $B = \{b_0, b_1, \dots, b_q\}$ be two disjoint sets of variables. A monotone function

$$f = (f_1, f_2, \dots, f_m) : A \cup B \to \{0,1\}^m$$

is <u>bilinear</u> if each prime implicant of f consists of one variable from A and one variable from B. Then, f is a set of <u>bilinear forms</u>.

<u>Example</u>:

Boolean matrix multiplication and also the Boolean convolution are bilinear.

◆

We can extend this definition to multilinear forms (i.e., we have more than two disjoint sets of variables) in the obvious way.

f is a set of <u>semidisjoint</u> bilinear forms if f has the following properties:

1) $PIM(f_i) \cap PIM(f_j) = \emptyset$ for $1 \leq i < j \leq m$.

2) For $1 \leq k \leq m$, each variable in $A \cup B$ is contained in at most one prime implicant in $PIM(f_k)$.

Note that Boolean matrix multiplication and also Boolean convolution are semi-disjoint.

For $a_i b_j \in PIM(f)$, we define the following subset $PIM_{ij}(f)$ of $PIM(f)$ inductively:

a) $a_i b_j \in PIM_{ij}(f)$,

b) $a_{i'} b_{j'} \in PIM_{ij}(f) \Rightarrow$
$a_{i''} b_{j'} \in PIM_{ij}(f) \; \forall \, a_{i''} b_{j'} \in PIM(f)$
and also
$a_{i'} b_{j''} \in PIM_{ij}(f) \; \forall \, a_{i'} b_{j''} \in PIM(f)$.

A set f of <u>disjoint</u> bilinear forms is a semi-disjoint set of bilinear forms which also fulfills the following third property:

3) For $1 \leq k \leq m$, $0 \leq i, j \leq n-1$,
$|\, PIM_{ij}(f) \cap PIM(f_k) \,| \leq 1$.

Boolean matrix multiplication is disjoint. Boolean convolution is not disjoint.

Exercise

a) Give a formal definition of multilinear forms. Extend the definitions of semi-disjointness and disjointness to multilinear forms.

b) Show that the generalized Boolean matrix multiplication is a set of disjoint multilinear forms.

c) Show that Boolean convolution is not a set of disjoint bilinear forms.

Boolean sums are (k,k)-disjoint. Boolean matrix multiplication is a set of disjoint bilinear forms. The generalized Boolean matrix multiplication is a set of disjoint multilinear forms. The Boolean convolution has not such disjointness property. The sets of variables upon which two functions $f_k, f_{k'} \in C_n$ depend can be almost equal which is not the case for the sets of functions mentioned above.

As a consequence, the assumptions of Theorem 3.3 do not hold for the Boolean convolution such that this theorem cannot be applied.

Now, we shall prove a general lower bound for the monotone network complexity of semidisjoint bilinear forms.

Theorem 3.8

Let f be a semidisjoint bilinear form. Let r_i be the number of prime implicants which contain the variable a_i. Then

$$C_{\wedge,\vee}(f) \geq \sum_{i=1}^{p} r_i^{1/2}$$

Corollary 3.1

The monotone network complexity of the Boolean n-th degree convolution is $\geq n^{3/2}$.

Proof:

This is a direct consequence of Theorem 3.8 since $r_i = n$ for $0 \leq i \leq n-1$.

■

Proof of Theorem 3.8:

Let β be an optimal Ω_m-network for f. Note that after replacing a_0 by 0, we obtain a subfunction of f which is semidisjoint as well. Moreover, the values r_i, $i > 0$ do not change.

$\Rightarrow$

It suffices to prove that after setting a_0 to 0, at least $r_0^{1/2}$ gates have been eliminated.

Let s_0 denote the number of functions f_i with $f_i = a_0 b_j$ for any j.

$\Rightarrow$

Setting a_0 to 0 eliminates these s_0 $\wedge$-gates where these outputs are computed.

Since f is semidisjoint, these gates cannot be used for the computation of other outputs

<u>Claim</u>

Setting a_0 to 0 eliminates at least $(r_0 - s_0)^{\frac{1}{2}}$ $\vee$-gates.

Note that this claim implies that setting a_0 to 0 eliminates at least $r_0^{1/2}$ gates.

To prove the claim, we consider exactly those functions f_k which depend on a_0 and which contain at least two prime implicants.

Let

$$P = v_0, v_1, \ldots, v_m$$

be a path from a_0 to f_k.

$\Rightarrow$

The path P contains at least one v-gate v_e with

$$a_i b_j \leq res_B(v_e)$$

for an $i \neq 0$ and any $j \in \{0,1,\ldots,n-1\}$.

Otherwise, because of the semidisjointness of f, the function f_k could not contain two prime implicants.

Exercise

Show that the property "P contains no v-gate v_e with $a_i b_j \leq res_B(v_e)$ for an $i \neq 0$ and some j" implies that f_k does not contain two prime implicants.

The first such an v-gate on P is called suitable for P. Let

$$\mathcal{P} := \{ P \mid P \text{ is a path from } a_0 \text{ to an output } f_k \text{ which depends on } a_0 \text{ and contains } \geq 2 \text{ prime implicants} \}.$$

Let

$$V^* := \{ \text{v-gate } v \mid v \text{ is suitable for a path } P \in \mathcal{P} \}.$$

Consides any $v \in V^*$. By construction, for each gate w between a_0 and v, each prime implicant p of $res_B(w)$ has the following property:

P contains at least one of the following monomials as a submonomial:

- a_0,
- the conjunction of two variables in A,
- the conjunction of two variables in B.

$\Rightarrow$

After setting a_0 to 0 and an eventual application of Theorem 3.2, $res_\beta(w)$ can be replaced by 0.

$\Rightarrow$

Setting a_0 to 0 yields for each gate $v \in V^*$ that one input can be replaced by 0.

$\Rightarrow$

After setting a_0 to 0, each gate $v \in V^*$ can be eliminated.

Claim: $|V^*| \geq (r_0 - s_0)^{\frac{1}{2}}$

To prove this claim, let

$$V^* = \{v_1, v_2, \ldots, v_q\}.$$

Then there are

$$i_1, i_2, \ldots, i_q \text{ and } j_1, j_2, \ldots, j_q$$

with

$$a_{i_\ell} b_{j_\ell} \leq res_\beta(v_\ell) \qquad 1 \leq \ell \leq q.$$

After setting

$$a_{i_\ell}, b_{j_\ell} \text{ to } 1 \quad \text{for } 1 \le \ell \le q,$$

all gates in V^* compute the constant 1.

$\Rightarrow$

No output f_k of f with ≥ 2 prime implicants depends on a_0.

Consider any such an output f_k and choose s such that

$$a_0 b_s \in \mathrm{PIM}(f_k).$$

After the assignment above, for the function f_k' computed at that output node, the following holds:

$$f_k' = 1 \quad \text{or} \quad b_s \in \mathrm{PIM}(f_k').$$

In the second case

$$a_{i_\ell} b_s \in \mathrm{PIM}(f_k)$$

for an $\ell \in \{1, 2, \dots, q\}$.

Since $i_\ell \neq 0$, this would contradict the semidisjointness of f. Hence,

$$f_k' = 1 \text{ and } a_{i_\ell} b_{j_t} \in \mathrm{PIM}(f_k)$$

for some $1 \le \ell, t \le q$.

This holds for all $r_0 - s_0$ output nodes depending on a_0 and having ≥ 2 prime implicants.

Since f is semidisjoint, prime implicants of different output nodes are different.

At most q^2 distinct prime implicants can be constructed using q variables in A and q variables in B.

$\Rightarrow$

$$q^2 \geq r_0 - s_0$$

$\Rightarrow$

$$|V^*| \geq (r_0 - s_0)^{\frac{1}{2}}.$$

Now, the assertion can be proved using induction.

■

Using Theorem 3.8, we have obtained an $\Omega(n^{3/2})$ lower bound for the number of $\vee$-gates in any monotone network which computes the n-th degree convolution. To get a lower bound for the number of $\wedge$-gates, we need some other techniques.

We start with an optimal monotone network β_0 computing C_n. We have no knowledge about the structure of β_0. To get knowledge, we

transform the network into a normal form network $\mathcal{B}_1$ which computes a number of subfunctions of C_n. For doing this, we split each output of a gate into several parts. We do this in such a manner that after the transformation the following normal form property holds:

- On every path P leading from an input node u with $op(u) = b_r \in \mathcal{B}$ to an output node there exists a node w such that:
  a) the direct successor of w on the path P is an $\wedge$-gate or the output node and
  b) $\exists\, b_s \in \mathcal{B}$, $b_s \neq b_r$ and $\exists\, A_S \subseteq A$, $|A_S| \geq 2 \cdot q$ such that

$$b_s \wedge \left(\bigvee_{a_j \in A_S} a_j \right) \leq res_{\mathcal{B}}(w).$$

The normal form transformation enlarges the number of $\wedge$-gates at most by the factor 4. During the transformation, we count some $\wedge$-gates. After the termination of the normal form transformation, we have counted $\left\lfloor \frac{1}{2}\left(\frac{n^2}{q} + n\right) \right\rfloor$ $\wedge$-gates in $\mathcal{B}_0$ and we are done or at least $\frac{n^2}{2}$ products $a_i b_j$ are still computed at the output gates in $\mathcal{B}_1$ which compute the subfunction $\bar{c}_{i+j}$ of c_{i+j}.

$\Rightarrow$

$\exists\, a_i$ such that at least q products $a_i \cdot b_j$ are computed at those output nodes.

Now we first set a_i to 1 and then we set successively all q b_ℓ's to 1. We prove that after every fixing of a b_ℓ, at least $\frac{1}{2}q$ $\wedge$-gates are eliminated.

$\Rightarrow$

In total, $\geq \lfloor \frac{1}{2} q^2 \rfloor$ $\wedge$-gates are eliminated.

To see this let us consider the computation graph which computes the product $a_i \cdot b_\ell$ at the output node for $\bar{c}_{i+\ell}$.

On every path P from the input node u with $op(u) = b_\ell$ to the output node g computing $\bar{c}_{i+\ell}$ consider the node w of the normal form property.

Assume there are less than $\frac{1}{2}q$ $\wedge$-gates.

$\Rightarrow$

$m < q$ pairwise distinct such nodes w are in the computation graph.

Normal form property $\Rightarrow$

$\exists\, b_{s_1}, b_{s_2}, \ldots, b_{s_m} \in B$, $b_{s_j} \neq b_\ell$ $1 \leq j \leq m$ and

$\exists\ A_1, A_2, \ldots, A_m \subseteq A$, $|A_j| \geq 2q$ $1 \leq j \leq m$ such that

$$a_i \bigwedge_{j=1}^{m} \left(b_{s_j} \wedge \left(\bigvee_{a_t \in A_j} a_t \right) \right) \leq res_{B_1}(g).$$

Definition of n-th degree convolution $\Rightarrow$

$\forall\ b_s \in B$ there exists at most one $a_t \in A$ with

$$a_t b_s \leq c_{i+e}.$$

Hence, for $1 \leq j \leq m$, it holds that

$\exists\ a_{j'} \in A_j$ such that $a_{j'} \bigwedge_{r=1}^{m} b_{s_r} \not\leq c_{i+e}$

and hence,

$$a_i \bigwedge_{j=1}^{m} a_{j'} \bigwedge_{r=1}^{m} b_{s_r} \not\leq c_{i+e}.$$

By construction,

$$a_i \bigwedge_{j=1}^{m} a_{j'} \bigwedge_{r=1}^{m} b_{s_r} \leq res_{B_1}(g).$$

This contradicts that $\overline{c}_{i+e} = res_{B_1}(g)$ is a subfunction of c_{i+e}.

$\Rightarrow$

$\geq \frac{1}{2} q$ $\wedge$-gates exist in the computation graph.

By setting a_i and b_e to 1, all these $\wedge$-gates are eliminated.

Now we shall give the lower bound proof.

The construction of β_1

Let β_0 be a monotone network computing C_n with minimal number of $\wedge$-gates. Let $0 < o_1 < \frac{1}{2}n$.

Beginning at the input nodes of β_0, we shall construct β_1 successively. In each step, we consider a node u in β_0, the direct predecessors of which were constructed in β_1 before.

$u \rightsquigarrow$ small network δ_u with output nodes u' and u''.

The input nodes of δ_u are the output nodes of δ_v and δ_w where v and w are the direct predecessors of u in β_0.

For $0 \leq k \leq 2n-2$, the node in β_0 which computes c_k is denoted by c_k.

An $\wedge$-gate g with $pred(g) = \{g_1, g_2\}$ is called a $(*)$-type-gate if

$$op(g_1) \in B \text{ and } res_{\beta_0}(g_2) = \bigvee_{a_j \in A'} a_j$$

where $\emptyset \neq A' \subseteq A$.

The network β_1 is constructed such that the following holds:

i) $res_{B_1}(u') \vee res_{B_1}(u'') \leq res_{B_0}(u)$.

ii) If $\exists\, b_s \in B$, $A_S \subseteq A$ maximal, $A_S \neq \emptyset$ such that $b_s\left(\bigvee_{a_j \in A_S} a_j\right) \leq res_{B_1}(u')$ then $|A_S| \geq 2q$.

iii) On every path P leading from a node u with $op(u) = b_r \in B$ to an $\wedge$-gate g which is not a $(*)$-type-gate or to the node u'' there exists a node w with $\exists\, b_s \in B$, $b_s \neq b_r$ and $\exists\, A_S \subseteq A$, $|A_S| \geq 2 \cdot q$ such that

$$b_s \wedge \left(\bigvee_{a_j \in A_S} a_j\right) \leq res_{B_1}(w).$$

Remark

Property i) means that the output nodes of δ_u compute only subfunctions of $res_{B_0}(u)$. Property iii) ensures that, after the construction of B_1, the normal form property introduced above holds.

Now we shall construct δ_u. We distinguish three cases.

Case 1: u is an input node of B_0.

Then δ_u consists of the nodes

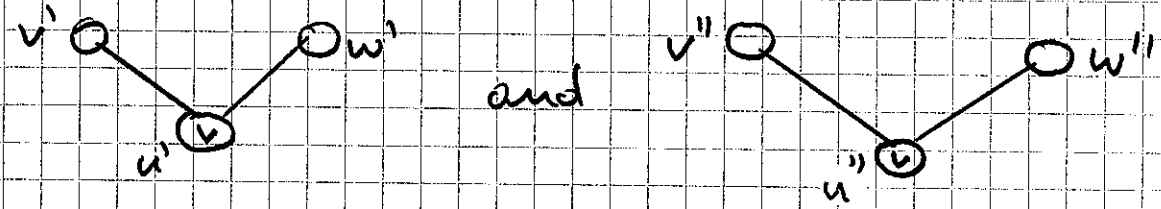
$$\begin{cases} u' \text{ with } op(u') = op(u) \\ u'' \text{ with } op(u'') = 0 \end{cases} \quad \text{if } op(u) \in B$$

$$\begin{cases} u' \text{ with } op(u') = 0 \\ u'' \text{ with } op(u'') = op(u) \end{cases} \quad \text{if } op(u) \in A$$

Obviously, conditions i), ii) and iii) hold after this construction

Case 2: u is an $\vee$-gate with $pred(u) = \{v, w\}$.

Then δ_u is constructed by



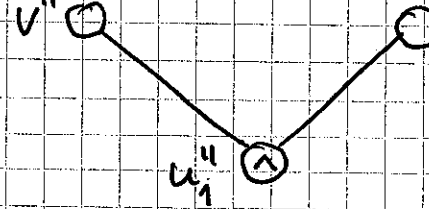
For this construction, we need no $\wedge$-gate. Since properties i), ii) and iii) hold for δ_v and δ_w, these properties also hold for δ_u.

Case 3: u is an $\wedge$-gate with $pred(u) = \{v, w\}$.

We have to realize $v'w'$, $v'w''$, $v''w'$ and $v''w''$.

Step 1: Realization of $v''w''$

Construct



For the realization of the other three products, we have to take care that the properties ii) and iii) are not destroyed after the construction.

Hence, for v' and w', respectively we distinguish two cases according to whether the following property is fulfilled or not.

We say for a node $g \in \{v', w'\}$ that g is <u>bipotent</u> if

$\exists\, b_s \in B, A_s \subseteq A$ and $b_r \in B, b_r \neq b_s, A_r \subseteq A$ such that

$$b_s \wedge \Big(\bigvee_{a_j \in A_s} a_j\Big) \vee b_r \wedge \Big(\bigvee_{a_j \in A_r} a_j\Big) \leq res_{B_1}(g)$$

<u>Remark:</u>

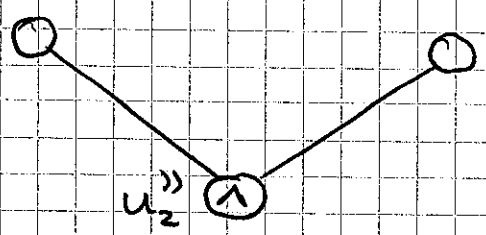
Since property ii) holds for δ_v and δ_w, $|A_s| \geq 2q$ and $|A_r| \geq 2q$.

If a node $g \in \{v', w'\}$ is not bipotent then either $res_{B_1}(g) = 0$ or $res_{B_1}(g) \leq b_r \wedge \Big(\bigvee_{a_j \in A_r} a_j\Big)$ for $b_r \in B$ and $A_r \subseteq A$.

<u>Step 2:</u> Realization of $v'w'$.

a) v' and w' are bipotent.

Construct



b) At least one of v' and w' is not bipotent.

$\Rightarrow$

There is at most one $b_s \in B$ such that

$$b_s \wedge \left(\bigvee_{a_j \in A_s} a_j \right) \leq res_{B_1}(v'w') \quad \text{for } A_s \subseteq A.$$

If no such $b_s \in B$, $A_s \subseteq A$ exist then by the structure of C_n and Theorem 3.2, we can replace $v'w'$ by 0 without changing the functions which are computed and we need not realize the product $v'w'$.

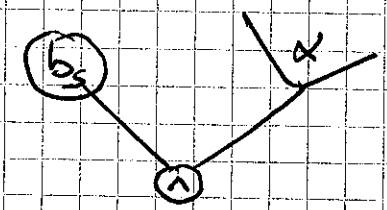
Let $b_s \in B$, $A_s \subseteq A$ maximal such that

$$b_s \wedge \left(\bigvee_{a_j \in A_s} a_j \right) \leq res_{B_1}(v'w').$$

We distinguish two cases.

i) $|A_s| \geq 2q$

Then we construct



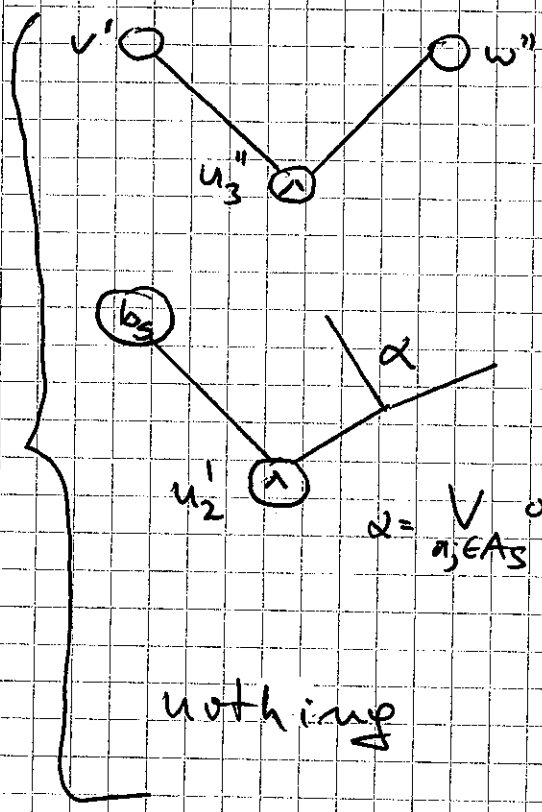
where α is a network which computes $\bigvee_{a_j \in A_s} a_j$ using only $(|A_s| - 1)$ $\vee$-gates.

ii) $|A_s| < 2q$.

Then we do not realize the product $v'w'$. By the structure of C_n, we destroy the com-

putation of $< 2q$ prime implicants of C_n.

Step 3 Realization of $v'w''$



if v' is bipotent

if v' is not bipotent and $\exists\, b_S \in B$, $A_S \subseteq A$ maximal with $|A_S| \geq 2q$ such that $b_S \wedge \left(\bigvee_{a_j \in A_S} a_j\right) \leq res_{B_1}(v'w'')$

otherwise

Step 4: Realization of $v''w'$

Analogous to that for $v'w''$. Produce result in u_4'' or u_3'.

If in the construction above

u_j', $j \in \{1,2,3\}$ (u_j'', $j \in \{1,2,3,4\}$, resp.)

do not exist then construct

u_j' with $op(u_j') = 0$ (u_j'' with $op(u_j'') = 0$, resp.).

Realization of u' and u'':

$$u' := \bigvee_{j \in \{1,2,3\}} u_j', \quad u'' := \bigvee_{j \in \{1,2,3,4\}} u_j''.$$

For the construction of δ_u, we need at most four $\wedge$-gates.

Exercise

Prove that after the construction of δ_u the properties i), ii) and iii) are fulfilled for δ_u.

If we destroy the computation of some prime implicants of C_n then, by construction, for the number N of these prime implicants we have

$$N \leq t \cdot 2^q$$

where $(4-t)$ $\wedge$-gates are used for the realization of the four products $v'w'$, $v'w''$, $v''w'$ and $v''w''$.

The following lemma characterizes the network β_1.

Lemma 3.3

In β_1, the following properties are fulfilled:

(1) For all node $u \in \beta_0$ and the output nodes u', u'' of δ_u, the following holds:

a) $res_{\beta_1}(u') \vee res_{\beta_1}(u'') \leq res_{\beta_0}(u)$.

b) If $\exists\, b_S \in B$, $A_S \subseteq A$ maximal, $A_S \neq \emptyset$ such that $b_S \wedge \left(\bigvee_{a_j \in A_S} a_j \right) \leq res_{\beta_1}(u')$ then $|A_S| \geq 2^q$.

(2) For $0 \leq k \leq 2^n - 2$, the output node c'_k computes 0.

(3) On every path P leading from a node h with $op(h) = b_r \in B$ to an $\wedge$-gate g which is not a $(*)$-type-gate or to C''_k, $k \in \{0, 1, \ldots, 2n-2\}$, there exists a node w such that

$\exists\, b_s \in B$, $b_s \neq b_r$ and $\exists\, A_s \subseteq A$, $|A_s| \geq 2q$ such that

$$b_s \wedge \Big(\bigvee_{a_j \in A_s} a_j \Big) \leq res_{\beta_1}(w).$$

(4) $0 \leq L_\wedge(\beta_1) \leq 4\, C^{\wedge}_{\Sigma_m}(C_n) - m$ where $L_\wedge(\beta_1)$ is the number of $\wedge$-gates in β_1 and at most $m \cdot 2q$ prime implicants of C_n have been destroyed.

Proof:
From the construction of β_1, (1) and (3) follow directly. Assertion (2) follows from (1) b) and the structure of C_k, $0 \leq k \leq 2n-2$. As observed above, for each $\wedge$-gate which is not used for the construction of δ_n, at most $2q$ prime implicants have been destroyed. Hence, assertion (4) follows. ∎

Using the network β_1, we shall prove the following theorem.

Theorem 3.9

$$C^{\wedge}_{\Sigma_m}(C_n) \geq \left\lfloor \frac{1}{8} \min\left\{\left(\frac{n^2}{q} - n\right), q^2\right\} \right\rfloor.$$

Setting $q := n^{2/3}$, we obtain the following corollary.

Corollary 3.2

$$\hat{C}_{-2m}(C_n) \geq \left\lfloor \frac{1}{8}\left(n^{4/3} - n\right) \right\rfloor.$$

Proof:

If in (4) of Lemma 3.3 $m \geq \frac{1}{2}\left(\frac{n^2}{q} - n\right)$ then the lower bound is proved. Otherwise, at least $n \cdot q$ prime implicants of C_n remain.

$\Rightarrow$

$\exists\, a_i \in A$, $\exists\, \tilde{B} \subseteq B$ with $|\tilde{B}| = q$ such that

$$a_i b_e \leq res_{B_1}(C''_{i+e}) \quad \forall\, b_e \in \tilde{B}.$$

Let $\tilde{B} = \{b_{e_1}, b_{e_2}, \ldots, b_{e_q}\}$.

We first fix a_i to 1 and eliminate all superfluous gates.

Observation

Fixing a_i to 1 does not destroy the normal form property since, if $a_i \in A_S$, the set A_S grows into the whole set A after fixing a_i to 1.

Then successively we set each $b_{e_j} \in \tilde{B}$ to 1 and eliminate all superfluous gates.

Since fixing an input variable to 1 does not affect the property that one function implies another function, during this process the normal form property is not

destroyed.

Now we prove that in each step in which we set a $b_{e_j} \in \tilde{B}$ to 1, at least $\frac{1}{2}q$ $\wedge$-gates are eliminated.

$\Rightarrow$

After the termination of this process, we have eliminated at least $\frac{1}{2}q^2$ $\wedge$-gates and the theorem is proved.

Assume, we have constructed the monotone network β_2 from β_1 by setting

$$a_i, b_{e_1}, b_{e_2}, \dots, b_{e_{r-1}}, \quad 1 \le r < q$$

to 1.

Claim

After setting b_{e_r} to 1, at least $\frac{1}{2}q$ more $\wedge$-gates can be eliminated.

Proof of claim

Since $a_i b_{e_1} b_{e_2} \cdots b_{e_{r-1}} \notin \operatorname{res}_{\beta_1}(C''_{i+e_r})$ and $a_i b_{e_r} \le \operatorname{res}_{\beta_1}(C''_{i+e_r})$ the following holds:

i) $\operatorname{res}_{\beta_2}(C''_{i+e_r}) \neq 1$ and

ii) $b_{e_r} \le \operatorname{res}_{\beta_2}(C''_{i+e_r})$.

Let h be the node in β_2 with $op(h) = b_{e_r}$. We consider all paths $P_1, P_2, \dots, P_s$ with

a) start node h and end node c''_{i+e_r} and

b) $b_{e_r} \leq res_{\beta_2}(v)$ for all nodes v on P_j, $1 \leq j \leq s$

Since $b_{e_r} \leq res_{\beta_2}(c''_{i+e_r})$, at least one such a path exists.

Obviously, setting b_{e_r} to 1 eliminates all $\wedge$-gates on the paths $P_1, P_2, \ldots, P_s$.

If on these paths $\frac{1}{2}q$ $\wedge$-gates exist, we are done. Assume that less than $\frac{1}{2}q$ $\wedge$-gates exist.

Property b) $\Rightarrow$

All $\wedge$-gates on these paths are not (*)-type-gates.

Normal form property $\Rightarrow$

For every path leading from the node h to the first $\wedge$-gate g on a path P_j, $1 \leq j \leq s$ (or to c''_{i+e_r} if no $\wedge$-gate on P_j exists) there is a node w such that

$\exists\, b_{t_j} \in B$, $b_{t_j} \neq b_{e_r}$, $\exists\, A_j \subseteq A$ with $|A_j| \geq 2q$ such that

$$b_{t_j} \wedge \Big(\bigvee_{a_d \in A_j} a_d \Big) \leq res_{\beta_2}(w)$$

and $g \in Suc(w)$ ($c''_{i+e_r} \in Suc(w)$, resp.)

$\Rightarrow$

$$\bigwedge_{j=1}^{s} \left(b_{t_j} \wedge \left(\bigvee_{a_d \in A_j} a_d \right) \right) \leq res_{B_2}(C''_{i+e_r})$$

and if no $\wedge$-gate on P_j exists then

$$b_{t_j} \wedge \left(\bigvee_{a_d \in A_j} a_d \right) \leq res_{B_2}(C''_{i+e_r}).$$

Assume that for all P_j, $1 \leq j \leq s$, an $\wedge$-gate on P_j exists. (Otherwise, the same proof with $s=1$ works).

Since less than $\frac{1}{2}q$ $\wedge$-gates exist on $P_1, P_2, \ldots, P_s$, less than q of the b_{t_j}, $1 \leq j \leq s$ can be pairwise distinct. Let

$$B' := \{ b_{t_1}, b_{t_2}, \ldots, b_{t_s}, b_{e_1}, b_{e_2}, \ldots, b_{e_{r-1}} \}.$$

Note that $b_{e_r} \notin B'$. Then we have

$$a_i \bigwedge_{b_j \in B'} b_j \bigwedge_{j=1}^{s} \left(\bigvee_{a_d \in A_j} a_d \right) \leq res_{B_2}(C''_{i+e_r}).$$

Since $\forall\, b_j \in B'$ there exists at most one $a_p \in A$ with $a_p b_j \leq C_{i+e_r}$

(namely $p = (i+e_r) - j$ if $(i+e_r) - j \geq 0$),

$|B'| < 2q$ and $|A_j| \geq 2q$, $1 \leq j \leq s$, the following holds:

$$\exists\, a_{p_j} \in A_j \text{ such that } a_{p_j} \bigwedge_{b_d \in B'} b_d \notin C_{i+e}$$

and hence,

$$a_i \bigwedge_{b_d \in B'} b_d \bigwedge_{j=1}^{s} a_{P_j} \not\leq C_{i+e_r}.$$

But by construction

$$a_i \bigwedge_{b_d \in B'} b_d \bigwedge_{j=1}^{s} a_{P_j} \leq res_{B_1}(C''_{i+e_r})$$

and by Lemma 33, property (1) a), $res_{B_1}(C''_{i+e_r})$ is a subfunction of C_{i+e_r},

a contradiction.

$\Rightarrow$

On $P_1, P_2, \ldots, P_s$ at least $\frac{1}{2}$ of $\wedge$-gates exist.

Hence, the claim and therefore Theorem 3.9 is proved.

□

Gronchuk and Sergeev have improved the lower bound for the number of $\vee$-gates to $\Omega\left(\frac{n^2}{\log^5 n}\right)$.

They use the fact that the Boolean convolution can be reduced to Boolean cyclic convolution which can be reduced to certain Boolean sums which are related to circulant matrices (see also Jukna, pp. 386 - 390).