Electronic Colloquium on Computational Complexity - Reports Series 1995 - available via:
ECCC
TR95-009
FTP: ftp.eccc.uni-trier.de:/pub/eccc/
WWW: http://www.eccc.uni-trier.de/eccc/
Email: ftpmail@ftp.eccc.uni-trier.de with subject 'help eccc'

A Note on Realizing Iterated Multiplication by small depth Threshold Circuits

Matthias Krause
Lehrstuhl Informatik II
Universität Dortmund
D44267 Dortmund

February 2, 1995

Abstract

It will be shown that decomposition via Chinese Remaindering does not yield polynomial size depth 3 threshold circuits for iterated multiplication of n n–bit numbers. This result will be achieved by proving that, in contrast to multiplication of two n–bit numbers, powering, division, and other related problems, the resulting subproblems, iterated multiplication modulo $polylog(n)$–bit numbers, do not have polynomial size approximation schemes over the set of all threshold functions. We use a lower bound argument based on probabilistic communication complexity.

1 Introduction

In the last years there has been proved a lot of interesting results on the power of small depth threshold circuits [A89,HMPST87,GHK92,Y90,BHKS92]. A main observation is that depth 3 threshold circuits are surprisingly powerful. So, it was shown by *Allender* [A89] that AC_0–functions can be realized by depth 3 threshold circuits of nearly polynomial size. This could still be improved by *Yao* who proved that this is true even for ACC–functions [Y90]. Another group of results concerns efficient small depth realizations of arithmetic operations. For example, *addition* and *comparison* of two n–bit numbers can be done by depth 2 threshold circuits with polynomially many edges [B90,AB91,BHKS92]. For other basic operations such as *multiple addition*, *sorting*, *multiplication*, *squaring*, *powering*, and *division* of n–bit numbers there are known depth 3 threshold circuits with polynomially many edges [BHKS92,H93]. A certain eyecatching exception is *iterated multiplication*, the multiplication of n n–bit numbers, for which the best known polynomial size threshold circuits have depth 4 [BHKS92].

This paper was initiated by several unsuccessful trials made at several places to construct more efficient threshold circuits for *iterated multiplication*. Is it really possible to construct depth 3 polynomial size circuits for this problem? We give a negative answer of the following type. The main and up to now only successful strategy for getting small depth realizations of arithmetic operations is to decompose the problem via Chinese Remaindering and handle the resulting subproblems in parallel [BHKS92,H93]. Using methods based on probabilistic communication complexity we show that in contrast to *multiple addition, multiplication* and *division* in the case of *iterated multiplication* this strategy does not lead to polynomial size depth 3 threshold circuits.

Observe that, unless there is a significant breakthrough in circuit lower bounds, an exhaustive negative answer cannot be expected because we don't know any method for proving even superlinear lower bounds on the size of depth 3 threshold circuits.

The paper is organized as follows. For making this article self–contained in section 2 we review the main techniques developed in [H93,BHKS92] for designing small depth threshold realizations, including the concepts of approximability and linear representations of Boolean functions. In section 3 we describe a connection between 1–approximability of multi–output functions and probabilistic communication complexity, which is the basis for our lower bound results presented in section 4.

Still one technical remark. For sake of shortness at many places in the text there will occur phrases like "Let $g : \{0,1\}^n \longrightarrow \{0,1\}$ be a Boolean function, where $C(g) \in n^{O(1)}$ ..." instead of "Let $g = (g_n : \{0,1\}^n \longrightarrow \{0,1\})_{n \in \mathbb{N}}$ be a sequence of Boolean functions, where $C(g_n) \in n^{O(1)}$ ...". Please do not get confused by this.

2 Preliminaries

2.1 Different notations of threshold realizations, and approximability

The basic processing elements of threshold circuits are Boolean threshold gates, T_r^n, producing one if and only if at least r of the n input bits are one. The complexity class $TC_{0,k}$, $k \geq 1$, contains all problems having depth k threshold circuits with polynomially many edges. A Boolean function $t : \{0,1\}^n \longrightarrow \{0,1\}$ is called a threshold function if t can be realized by one single threshold gate.

Definition 2.1 *We say that $f : \{0,1\}^n \longrightarrow \{0,1\}$ has a threshold representation over the set G of Boolean functions over $\{0,1\}^n$ if f can be realized by a depth two circuit consisting of gates performing functions from G at the bottom level and a threshold gate at the top. The number of edges is called the weight of the representation.*

Clearly, $f \in TC_{0,k}$ if and only if f has a polynomial weight threshold representation over $TC_{0,k-1}$. It will be convenient to work with the following different (but equivalent)

notation.

Lemma 2.1 **(i)** *The function f has a threshold representation over G if and only if there are functions $g_1, \ldots, g_s \in G$, real numbers $r, w_1, \ldots, w_s$ and a distance parameter $\delta > 0$ such that for all inputs x*

$$f(x) = 1 \implies \sum_{k=1}^{s} w_k g_k(x) \geq r + \delta$$

$$f(x) = 0 \implies \sum_{k=1}^{s} w_k g_k(x) \leq r - \delta.$$

(ii) *The function f has a threshold representation of polynomial weight over G if and only if there is a representation as in (i), and s, δ^{-1}, as well as $|r|, |w_1|, \ldots, |w_s|$ are polynomially bounded in n.* □

For constructing small depth threshold circuits it has been proved very usefull to investigate approximability of functions over a given basis.

Definition 2.2 *The function f is called G–approximable if for any $\epsilon > 0$ there are $g_1, \ldots, g_s \in G$, $w_1, \ldots, w_s \in \mathbb{R}$, where s and $w = \sum_{k=1}^{s} |w_k|$ are polynomially bounded in n and ϵ^{-1}, such that for all x $|f(x) - \sum_{k=1}^{s} w_k g_k(x)| \leq \epsilon$.*

It can easily be proved that the G–approximability of f provides a polynomial weight threshold representation of f over G. Our special interest is devoted to $TC_{0,k}$–approximability which is usually called *k–approximability*. The special effect of saving depth is due to the following fact.

Lemma 2.2 *Suppose we are given G–approximable functions $h_1, \ldots, h_m$, where m is polynomially bounded in n. Further suppose that f has a polynomial weight threshold representation over $\{h_1, \ldots, h_m\}$. Then f has a polynomial weight threshold representation over G.*

Proof: By definition there are $\delta > 0$ and reals $r, w'_1, \ldots, w'_m$, where δ^{-1} and $w' = \sum_{l=1}^{m} |w'_l|$ are polynomially bounded in n such that for all inputs x

$$f(x) = 1 \implies \sum_{l=1}^{m} w'_l h_l(x) \geq r + \delta \quad \text{and} \quad f(x) = 0 \implies \sum_{l=1}^{m} w'_l h_l(x) \leq r - \delta.$$

Now, for each l, $1 \leq l \leq m$, take a polynomial weight threshold representation of h_l over G which approximates h_l with error $\frac{\delta}{2w'}$. Combining these approximators according to $w'_1, \ldots, w'_m$ yields a polynomial weight threshold representation of f over G. □

2.2 Linear representations of Boolean functions

Definition 2.3 *A Boolean function $f : \{0,1\}^n \longrightarrow \{0,1\}$ is called to have a linear representation of length q if there is a linear mapping $l : \mathbb{R}^n \longrightarrow \mathbb{R}$ and a set of pairwise disjoint intervals $[a_1, b_1], [a_2, b_2], \ldots, [a_q, b_q]$ of the real line such that for all inputs x*

$$f(x) = 1 \quad \Longleftrightarrow \quad l(x) \ \in \ \bigcup_{j=1}^{q} [a_j, b_j].$$

Observe that the linear mapping $l(x_1, \ldots, x_n) = \sum_{i=1}^{n} 2^i x_i$ induces a linear representation for each Boolean function f. The function f is symmetric if and only if it has a linear representation with respect to $l(x_1, \ldots, x_n) = \sum_{i=1}^{n} x_i$. Threshold functions have linear representations of length 1. The known constructions of efficient threshold circuits of very small depth make use of the following observation [BHKS92,H93].

Lemma 2.3 *If $f : \{0,1\}^n \longrightarrow \{0,1\}$ has a linear representation of polynomial length then f is 1–approximable.*

Proof: Fix a linear mapping l and a related set of intervals, say $[a_1, b_1]$,$[a_2, b_2]$, ... $[a_q, b_q]$, realizing f as in Definition 2.3 where additionally $q \in n^{O(1)}$. Observe that f can be written as

$$f(x) \ = \ \sum_{i=1}^{q} (t_i(x) + t'_i(x)) - q,$$

where the unbounded weight threshold functions t_i and t'_i test whether $l(x) \geq a_i$ and $l(x) \leq b_i$, respectively. It has been shown in [GHR92] that arbitrary threshold functions are 1–approximable. In [GK93] there has been given an explicit construction of those approximators. Combining $\frac{\epsilon}{2q}$–approximators for all t_i, t'_i with the above exact representation gives an ϵ–approximator for f. Hence, f is 1–approximable. □

2.3 Realizing operations by Chinese Remaindering

Realizing an operation $F : \{0,1\}^n \longrightarrow \{0,1\}^m$, where $m \in n^{O(1)}$, by Chinese Remaindering means the following. Fix p–bit numbers $q_1 \ldots, q_r$ and set $Q = \prod_{j=1}^{r} q_j$ such that $p \in O(\log(n))$, $Q \geq 2^m$, $\log(Q) \in n^{O(1)}$, and that for all $i \neq j \in \{1, \ldots, r\}$ the greatest common divisor of q_i and q_j is one. The corresponding Chinese Remaindering Transformation $CRT : \{0,1\}^m \longrightarrow (\{0,1\}^p)^r$ is defined

$$CRT(y) \ = \ (y \ mod \ q_1, \ldots, y \ mod \ q_r).$$

Given $F, q_1 \ldots, q_r, Q$ as above let $F^{(j)}(x) \ = \ F(x) \ mod \ q_j$.

Lemma 2.4 *If for all $j = 1, \ldots, r$ each output bit of $F^{(j)}$ is k–approximable then all output bits of F are $(k+1)$–approximable.*

Proof: It is sufficient to show that $CRT^{-1} : (\{0,1\}^p)^r \longrightarrow \{0,1\}^m$ the inverse transformation of Chinese Remaindering, has a linear representation of polynomial length. Such an representation can be obtained using the well–known fact that CRT^{-1} can be written as

$$CRT^{-1}(y^1, \ldots, y^r) \;=\; \sum_{j=1}^{r} e_j y^j \; mod \; Q,$$

where $e_1, \ldots, e_r$ are fixed numbers from $\{1, \ldots, q-1\}$, the so–called orthogonal idempotents, characterized by the property $e_i \; mod \; q_j = \delta_{i,j}$. □

For the sake of illustration we recall the proof in [H93] of 2–approximability of the multiplication of two n bit numbers $mult_n : \{0,1\}^n \times \{0,1\}^n \longrightarrow \{0,1\}^{2n}$.

Lemma 2.5 *Let $n, m = 2n, Q, q_1, \ldots, q_r$ be defined as above. Then for all $j = 1, \ldots, r$ $mult_n^{(j)}$ is 1–approximable.*

Proof: For all n–bit numbers $x = (x_{n-1}, \ldots, x_0), y = (y_{n-1}, \ldots, y_0) \in \{0,1\}^n$ define

$$l(x,y) \;=\; \sum_{i=0}^{n-1} a_i x_i \;+\; n q_j \left(\sum_{i=0}^{n-1} a_i y_i \right),$$

where for all $i = 1, \ldots, n$, $a_i = 2^i \; mod \; q_j$. It can easily be checked that l induces a linear representation for each output bit of $mult_n^{(j)}$ and, as $a_0, \ldots, a_n, q_j$ have polynomial size, the length of this representation is polynomially bounded. □

How is the situation for $it.mult_n : (\{0,1\}^n)^n \longrightarrow \{0,1\}^{n^2}$, the multiplication of n n–bit numbers? The best known realization of $it.mult_n(x^1, \ldots, x^n)$, which shows that $it.mult_n$ is 3–approximable, requires that $q_1, \ldots, q_r$ are prime numbers. Further we need in advance for each $j = 1 \ldots r$ a number u_j having multiplicative order $q_j - 1$ in $\mathbb{F}_{q_j}^*$. The first step is to compute in parallel $\log_{u_j} x^i$, $i = 1 \ldots n$, $j = 1 \ldots r$, this is 1–approximable. Computing from this $it.mult_n(x^1, \ldots, x^n) \; mod \; q_j$ via adding the discrete logarithms is also 1–approximable. We obtain 2–approximability of all $it.mult_n^{(j)}$ and 3–approximability of $it.mult_n$.

The main result of this paper is that for all p–bit numbers q, where $p \in O(\log(n))$, which have a prime factor not smaller than 5, $it.mult_n \; mod \; q$ is not 1–approximable, not depending on how the binary code of the output is chosen. Consequently, realizing $it.mult$ by Chinese Remaindering does not lead to polynomial size depth three circuits.

3 1–Approximability and Probabilistic Communication Complexity

This section is devoted to the proof of the following statement.

Theorem 1 *Let $g : \{0,1\}^n \longrightarrow \{0,1\}^m$ be a 1–approximable Boolean operation with $m = O(\log(n))$. Further let $h : \{0,1\}^m \longrightarrow \{0,1\}$ be arbitrarily fixed. Then the probabilistic communication complexity of $f = h \circ g : \{0,1\}^n \longrightarrow \{0,1\}$ with respect to an arbitrarily fixed partition of the set of input variables is at most $O(m \log(n))$.*

We will work with the following definition of one–way probabilistic communication protocols, for more details see [HR88,K91]. Suppose we are given a Boolean function $f = f(x_1, \ldots, x_n, y_1, \ldots, y_n)$ in a fixed distributed form. Communication protocols for f refer to a pair (P_0, P_1) of processors of unbounded computational power which want to cooperate in computing $f(x,y)$ under the restriction that P_0 only knows the left input half x, and P_1 only knows the right input half y. A protocol Π of length k works as follows. In dependence of x and a private random string r, processor P_0 computes a message $M = M(x,r) \in \{0,1\}^k$ and sends it to P_1. Then processor P_1 decides in dependence of w and y deterministically whether to accept or to reject the input. Π computes the function f with advantage ϵ if for all inputs (x,y) it holds

$$f(x,y) = 1 \quad \Longrightarrow \quad \text{Prob}[\Pi \text{ accepts } (x,y)] \ \geq \ \frac{1}{2} + \epsilon.$$

$$f(x,y) = 0 \quad \Longrightarrow \quad \text{Prob}[\Pi \text{ accepts } (x,y)] \ \leq \ \frac{1}{2} - \epsilon.$$

Let us fix a number $\epsilon = \frac{1}{3}2^{-4m}$. By definition, there are numbers $s, w, W \in n^{O(1)}$, threshold functions $T_1, \ldots, T_s : \{0,1\}^n \times \{0,1\}^n \longrightarrow \{0,1\}$ of weight $\leq W$, and real functions $g'_j = \sum_{k=1}^{s} w_{j,k} T_k$, $j = 1, \ldots, m$, fulfilling for all j,k $|w_{j,k}| \leq w$, such that $\left|g_j(x,y) - g'_j(x,y)\right| \leq \epsilon$ for all inputs $x, y \in \{0,1\}^n$ and all j, $1 \leq j \leq m$.
Further observe that there is a uniquely defined multilinear mapping

$$h' = \sum_{\alpha \in \{0,1\}^m} \lambda_\alpha x^\alpha : \ \mathbb{R}^{\,m} \longrightarrow \ \mathbb{R}$$

which coincides with h on $\{0,1\}^m$. Using *Cramer*'s rule it can be derived that all coefficients λ_α are integers between -2^m and 2^m.

We define $f' : \{0,1\}^n \times \{0,1\}^n \longrightarrow \mathbb{R}$ by $f' = h' \circ g'$.

Observe that for all $x, y \in \{0,1\}^n$ $|f(x,y) - f'(x,y)| = |h'(z) - h'(z')|$, where $z = (g_1(x,y), \ldots, g_m(x,y))$ and $z' = (g'_1(x,y), \ldots, g'_m(x,y))$.

Consequently,

$$|f(x,y) - f'(x,y)| \ \leq \ 2^{2m} \max_{\alpha \in \{0,1\}^m} |z^\alpha - z'^\alpha| \ \leq$$

$$\leq 2^{2m} \max\{2\epsilon, (1+\epsilon)^m - (1-\epsilon)^m\} \ \leq \ 2^{2m} 2\epsilon m (1+\epsilon)^{m-1} \ \leq \epsilon 2^{4m}.$$

By definition of ϵ we obtain that for all $x, y \in \{0,1\}^n$

$$|f(x,y) - f'(x,y)| \ \leq \ \frac{1}{3}. \tag{1}$$

Applying the distributive law to $f' = h' \circ g'$ and simplifying the expression appropriately the function f' can be written as

$$f' = \sum_{J \subseteq \{1,\ldots,s\}, |J| \leq m} v_J U_J, \tag{2}$$

where $U_J : \{0,1\}^n \times \{0,1\}^n \longrightarrow \{0,1\}$ are defined as $U_J(x,y) = \bigwedge_{j \in J} T_j(x,y)$.
Observe that for all $J \subseteq \{1,\ldots,s\}$ it holds that

$$|v_J| \in n^{O(m)}, \tag{3}$$

and that by (1) f' defines a threshold representation of f.
It is now quite straightforward (but somewhat technical) to construct from (1), (2), and (3) a probability distribution R on the functions U_J and some $\epsilon^* > 0$, where $\epsilon^{*-1} \in n^{O(m)}$, such that for all $x, y \in \{0,1\}^n$

$$Prob_R[f(x,y) = U_J(x,y)] \geq \frac{1}{2} + \epsilon^* \tag{4}$$

(Observe that those U_J for which $v_J < 0$ have to be replaced by their negation.)
Now we are ready to define an efficient probabilistic communication protocol Π for f. Suppose that for all j, $1 \leq j \leq s$, the threshold function T_j is defined as

$$T_j(x,y) = 1 \iff \sum_{i=1}^{n} a_{i,j} x_i + \sum_{i=1}^{n} b_{i,j} y_i \geq c_j,$$

where $a_{i,j}, b_{i,j}, c_j$ are integers fulfilling $\sum_{i=1}^{n} |a_{i,j}| + \sum_{i=1}^{n} |b_{i,j}| + |c_j| \leq W$.
Given the input x to player P_0 and y to player P_1, P_0 chooses randomly according to R a function $U_J = T_{j_1} \wedge \ldots \wedge T_{j_l}$, $l \leq m$, and sends the message

$$j_1 \# \sum_{i=1}^{n} a_{i,j_1} x_i \# \ \ldots \ \# j_l \# \sum_{i=1}^{n} a_{i,j_l} x_i.$$

Player P_1 can now compute $T_{j_1}(x,y), \ldots, T_{j_l}(x,y)$ and, consequently, $U_J(x,y)$. P_1 accepts if $U_J(x,y) = 1$ and rejects otherwise. It can easily be checked that this protocol computes f with advantage ϵ^*. The length of the protocol is bounded by $O(m(\log(s) + \log(W)) = O(m \log(n))$. $\Box$

4 Iterated multiplication modulo small integers is not 1–approximable

This section is devoted to the proof of

Lemma 4.1 *Let $q \geq 5$ be an $O(\log(n))$–bit prime number. Then there is no output representation of $it.mult_n$ mod q of length $\log^{O(1)} n$ such that all output bits are 1–approximable.*

Using this lemma it is quite straightforward to show

Theorem 2 *Let r be an $O(\log(n))$–bit number having a prime factor not smaller than 5. Then there is no output representation of $it.mult_n$ mod r of length $\log^{O(1)} n$ such that all output bits are 1–approximable.*

Proof: Suppose there is a prime number $q \geq 5$, a natural number $r = r'q$ and an output representation

$$it.mult_n \ mod \ r : (\{0,1\}^n)^n \longrightarrow \{0,1\}^m,$$

$m = \log^{O(1)} n$, for $it.mult_n \ mod \ r$ such that each output bit is 1–approximable. By Theorem 1, this 1–approximator can be used to construct a 1–approximator for each output bit of $it.mult_n \ mod \ q$. This contradicts Lemma 4.1. □

We will prove Lemma 4.1 by using the following lower bound result from [HR88] on the probabilistic communication complexity of ip_n^2 defined by

$$ip_n^2(x_1, \ldots, x_n, y_1, \ldots, y_n) = x_1y_1 \oplus \ldots \oplus x_ny_n.$$

Lemma 4.2 *For all $\epsilon > 0$ fulfilling $\epsilon^{-1} \in 2^{\log^{O(1)} n}$ it holds that the length of any probabilistic communication protocol computing ip_n^2 with advantage ϵ is $\Omega(n)$.* □

Consider now the function $f : (\{0,1\}^n)^n \longrightarrow \{0,1\}$, over n n–bit numbers $x^{(1)}, \ldots, x^{(n)}$ defined by

$$f(x^{(1)}, \ldots, x^{(n)}) = 1 \quad \Longleftrightarrow \quad \prod_{i=1}^n x^{(i)} \text{ is a quadratic non–residue mod } q.$$

We construct a partition of the n^2 input variables of f into subsets U and V such that for any $\epsilon > 0$, $\epsilon^{-1} \in 2^{\log^{O(1)} n}$ the length of any probabilistic communication protocol computing f with advantage ϵ is $\Omega(n)$. By Theorem 1 this proves Lemma 4.1.

We do that by defining a *rectangular reduction* of ip_n^2 to f, i.e. we define mappings $l : \{0,1\}^n \longrightarrow \{0,1\}^U$ and $r : \{0,1\}^n \longrightarrow \{0,1\}^V$ such that either $ip_n^2(x,y) = f(l(x), r(y))$ for all $x, y \in \{0,1\}^n$, or $ip_n^2(x,y) = \neg f(l(x), r(y))$ for all $x, y \in \{0,1\}^n$. This is sufficient as l and r translate each protocol for f into a protocol of the same length for ip_n^2. Before defining U, V, l, r observe the following facts.

Lemma 4.3 *For all primes $q \geq 5$ there is a natural number $a, 1 \leq a \leq q-3$, fulfilling*

$$\left(\frac{a}{q}\right) = \left(\frac{a+1}{q}\right) \neq \left(\frac{a+2}{q}\right)$$

Proof: As usual, $\left(\frac{a}{q}\right)$ denotes the *Legendre* symbol defined as

$$\left(\frac{a}{q}\right) = \begin{cases} 0, & \text{if } q \text{ divides } a \\ 1, & \text{if } a \text{ is a quadratic residue modulo } p \\ -1, & \text{if } a \text{ is a quadratic non–residue modulo } p \end{cases}$$

The multiplicative group $\mathbb{F}_q^*$ can be written as $\{1, \omega, \omega^2, \ldots, \omega^{q-1}\}$ for some $\omega \in \mathbb{F}_q^*$. Clearly, $a' \in \mathbb{F}_q^*$ is a quadratic residue modulo p if and only if $\log_\omega a'$ is even.

Consequently, it is sufficient to prove the existence of a, $1 \leq a \leq q-2$, such that $\left(\frac{a}{q}\right) = \left(\frac{a+1}{q}\right)$. But this follows straightforwardly from the fact that $\left(\frac{1}{q}\right) = \left(\frac{4}{q}\right) = 1$. □

We get l, r by defining mappings $\bar{l} : \{0,1\} \longrightarrow \{0,1\}$ and $\bar{r} : \{0,1\} \longrightarrow \{0,1,\ldots,2^n-1\}$ and setting

$$(l,r)(x_1,\ldots,x_n,y_1,\ldots,y_n) = (\bar{l}(x_1)+2\bar{r}(y_1),\ldots,\bar{l}(x_n)+2\bar{r}(y_n)).$$

The corresponding partition (U,V) is given by putting all last bits of the n–bit input numbers $x^{(1)},\ldots,x^{(n)}$ into U and the remaining input bits into V.

The mapping $\bar{l}$ will be the identity, i.e., $\bar{l}(b) = b$ for $b \in \{0,1\}$, and $\bar{r}$ will be defined as $\bar{r}(0) = A$ and $\bar{r}(1) = A+1$ for some appropriate number A, $1 \leq A \leq p-1$.

Now observe that $f(x^{(1)},\ldots,x^{(n)}) = 1$ if and only if there is no i, $1 \leq i \leq n$, such that q divides $x^{(i)}$ and the number of those i with $\left(\frac{x^{(i)}}{q}\right) = -1$ is odd.

Consequently, the mappings l and r will do their job if A has one of the following properties.

(a) It holds $\left(\frac{\bar{l}(0)+2\bar{r}(0)}{q}\right) = \left(\frac{\bar{l}(0)+2\bar{r}(1)}{q}\right) = \left(\frac{\bar{l}(1)+2\bar{r}(0)}{q}\right) = 1$ and $\left(\frac{\bar{l}(1)+2\bar{r}(1)}{q}\right) = -1$.

This is equivalent to $\left(\frac{2A}{q}\right) = \left(\frac{2A+1}{q}\right) = 1$ and $\left(\frac{2A+2}{q}\right) = -1$ and means that l, r reduces ip_n^2 to f.

(b) It holds $\left(\frac{\bar{l}(0)+2\bar{r}(0)}{q}\right) = \left(\frac{\bar{l}(0)+2\bar{r}(1)}{q}\right) = \left(\frac{\bar{l}(1)+2\bar{r}(0)}{q}\right) = -1$ and $\left(\frac{\bar{l}(1)+2\bar{r}(1)}{q}\right) = 1$.

This is equivalent to $\left(\frac{2A}{q}\right) = \left(\frac{2A+1}{q}\right) = -1$ and $\left(\frac{2A+2}{q}\right) = 1$ and means that l, r reduces ip_n^2 to f, if n is odd, or l, r reduces ip_n^2 to $\neg f$ if n is even.

By Lemma 4.3 there is some a, $1 \leq a \leq q-3$, such that $\left(\frac{a}{q}\right) = \left(\frac{a+1}{q}\right) \neq \left(\frac{a+2}{q}\right)$. It is easy to check that $A = ca \mod q$ where c denotes the multiplicative inverse of 2 in $\mathbb{F}_q^*$, matches all our requirements. □

Acknowledgment

I would like to thank Claudia Bertram, Thomas Hofmeister, Ingo Wegener, and Stephan Waack for helpful discussions.

References

[A89] Allender,E.: *A note on the power of threshold circuits*, Proceedings der 30. IEEE Symposium FOCS, 1989, 580–584.

[AB91] Alon,N.,J.Bruck: *Explicit constructions of depth–2 majority circuits for comparison and addition*, Technical Report RJ 8300 (75661) of the IBM Almaden Research Center, San Jose, 1991.

[B90] Bruck,J. *Harmonic analysis of polynomial threshold functions*, SIAM Journal of Discrete Mathematics, 3, Nr. 22, 1990, 168–177.

[BHKS92] Bruck,J.,Th.Hofmeister,Th.Kailath,K.Y.Siu, *Depth efficient networks for division and related problems.* Technical Report 1992, to appear in IEEE Transactions on Information Theory.

[GHR92] Goldmann,M.,J.Håstad,A.A.Razborov: *Majority Gates versus general weighted threshold gates*, J. of Computational Complexity 2 (1992), 277–300.

[GK93] Goldmann,M.,M.Karpinski: *Simulating Threshold Circuits by Majority Circuits.* Proc. 25th ACM Conference STOC, 1993.

[HMPST87] Hajnal,A.,W.Maass, P.Pudlák, M.Szegedy, G.Turán: *Threshold circuits of bounded depth,* Proc. 28th IEEE Conf. FOCS, 1987, 99–110.

[HR88] Halstenberg,B.,R.Reischuk *Relations between communication complexity classes* Proc. of the 3. IEEE Structure in Complexity Theory Conference, 1988, 19–28.

[H93] Hofmeister,Th. *Depth–efficient threshold circuits for arithmetic functions* in: *Theoretical Advances in Neural Computation and Learning* eds. Roychowdhury et.al., Kluwer Academic Publishers, ISBN 0–7923–9478–X.

[HHK91] Hofmeister,Th.,W.Hohberg,S.Köhling: *Some notes on threshold circuits and multiplication in depth 4* IPL 39 (1991) 219–225.

[K91] Krause,M. *Geometric Arguments yield better bounds for threshold circuits and distributed computing* Proc. of the 6. IEEE Structure in Complexity Theory Conference, 314–322.

[KW91] Krause,M.,S.Waack, *Variation ranks of communication matrices and lower bounds for depth two circuits having symmetric gates with unbounded fan-in*, Proc. 32th IEEE Conference FOCS, 1991, 777–787.

[RT92] Reif,J.H.,S.R.Tate *On threshold circuits and polynomial computation* SIAM Journal of Computing, Vol.21,Nr.5,pp.896–908, 1992

[Y90] Yao,A.C.: *On ACC and Threshold Circuits,* Proc. 31th IEEE Conference FOCS, 1990, 619–628.