MAX-PLANCK-INSTITUT FÜR INFORMATIK

Circuits and Multi–Party Protocols

– technical report No. 104 –

Vince Grolmusz
Max Planck Institute for Computer Science
and
Eötvös University

January 30, 1992

INFORMATIK

Im Stadtwald
66123 Saarbrücken
Germany

Circuits and Multi–Party Protocols

– technical report No. 104 –

Vince Grolmusz
Max Planck Institute for Computer Science
and
Eötvös University

January 30, 1992

Circuits and Multi–Party Protocols

– technical report No. 104 –

Vince Grolmusz
Max Planck Institute for Computer Science
and
Eötvös University

ABSTRACT

We present a multi–party protocol for computing certain functions of an $n \times k$ $0-1$ matrix A. The protocol is for k players, where player i knows every column of A, except column i. *Babai, Nisan* and *Szegedy* [BNS] proved that to compute $GIP(A)$ needs $\Omega(n/4^k)$ bits to communicate. We show that players can count those rows of matrix A which sum is divisible by m, with communicating only $O(mk \log n)$ bits, while counting the rows with sum congruent to 1 $\pmod{m}$ needs $\Omega(n/4^k)$ bits of communication (with an odd m and $k \equiv m \pmod{2m}$). $\Omega(n/4^k)$ communication is needed also to count the rows of A with sum in any congruence class modulo an *even* m.

The exponential gap in communication complexities allows us to prove exponential lower bounds for the sizes of some bounded–depth circuits with MAJORITY, SYMMETRIC and MOD_m gates, where m is an odd – prime or composite – number.

keywords: lower bounds, threshold circuits, ACC–circuits, communication protocols

Address: Max Planck Institute for Computer Science, Im Stadtwald, W–6600 Saarbruecken, GERMANY; email: grolmusz@robin.cs.sb-uni.de

1. INTRODUCTION

The connection between the circuit complexity and the communication complexity plays an important role in the recent literature of the circuit lower bound theory.

The notion of the (2 party) communication complexity was introduced by *Yao* [Y1]. Due to the algebraic characterization of the communication complexity, several strong lower bounds was proved for this model (see [L] for a survey).

Karchmer and Wigderson [KW] extended the original communication model of *Yao*, to compute some relations instead of Boolean functions; then they proved a that the optimal circuit–depth of a Boolean function and the communication complexity of a relation is the same number. This Karchmer–Wigderson theorem was applied to prove an $\Omega(\log^2 n)$ lower bound for the depth of monotone polynomial–sized circuits, computing graph s-t connectivity. *Raz* and *Wigderson* [RW2] used the Karchmer–Wigderson theorem to get linear lower bound for the depth of monotone Boolean circuits, computing graph matching. The proof make use the linear lower bound for the probabilistic communication complexity of the disjointness function of *Kalyanosundaram* and *Snitger* [KS], and *Razborov* [R]. The correspondence between circuits and communication complexity appears also in the work of *Yao* [Y2], *Raz* and *Wigderson* [RW1], and *Szegedy* [S].

The *multi–party communication game*, first examined by *Chandra, Furst* and *Lipton* [CFL], is a generalization of the 2–party communication game. In this game, k players: $P_1, P_2 ..., P_k$ intend to compute the value of $g(A_1, A_2, ..., A_k)$, where $g : \{0,1\}^{kn} \rightarrow \mathbf{N}$ where $\mathbf{N}$ denotes the set of natural numbers, and $A_i \in \{0,1\}^n$, for $i = 1,2,...,k$. Player P_i knows every variable, *except* A_i, for $i = 1,2,...,k$. The players have unlimited computational power, and they communicate with the help of a blackboard, viewed by all players. Only one player may write on the blackboard at a time. The goal is to compute $g(A_1, A_2, ..., A_k)$, such that at the end of the computation, all players know this value. The cost of the computation is the number of bits written on the blackboard for the given $A = (A_1, A_2, ..., A_k) \in \{0,1\}^{nk}$. The cost of a multi–party protocol is the maximum number of bits communicated for any A from $\{0,1\}^{nk}$. The k-party communication complexity, $C^{(k)}(g)$, of a function g, is the minimum of costs of those k-party protocols which compute g.

The theory of the two–party communication games are well developed [L], but much less is known about the multi–party communication complexity of functions. Communicating n bits, P_1 can compute any function of A: P_2 writes down the n bits of A_1 on the blackboard, P_1 reads it, and computes the value $g(A)$ at no cost. The additional cost of diffusing the result $g(A)$ to other players is the binary length of $g(A)$.

Babai, Nisan and Szegedy examined the *Generalized Inner Product* (GIP) function in [BNS].

Notation 1. *Let $\{0,1\}^{n\times k}$ denote the set of all $0-1$ matrices of n rows and k columns. Let $A \in \{0,1\}^{n\times k}$, We shall refer to the i^{th} column of A as A_i, the j^{th} row of A as A^j, and to the i^{th} entry in row j as A_i^j. Let $GIP(A)$ denote the number of the all–1 rows of matrix A, modulo 2.*

In other words, if column A_i is considered to be the characteristic vector of a subset Y_i of a fixed n-element set for $i = 1,2,...,k$, then

$$GIP(A) = | Y_1 \cap Y_2 \cap Y_3 \cap ... \cap Y_k | \text{ mod} 2.$$

Babai, Nisan and *Szegedy* [BNS] gave a lower bound for even that case, when the players compute GIP on *most of the inputs*:

Definition 2. *[BNS] The k–party ε–distributional communicational complexity of a function g, denoted by $C_\varepsilon^{(k)}(g)$, is the minimum number of bits that needed to be exchanged in the worst case, by any k–party protocol which computes g correctly on $1/2+\varepsilon$ fraction of the inputs.*

Theorem 3. *[BNS, Theorem 2]*

$$C_\varepsilon^{(k)}(GIP) = \Omega\left(\frac{n}{4^k} + \log \varepsilon\right).$$

■

Substituting $\varepsilon = 1/2$ in Theorem 3, we get that the multi– party communication complexity of GIP is

$$\Omega\left(\frac{n}{4^k}\right).$$

A protocol in [G] communicates

$$O\left(\frac{n}{2^k}k\right)$$

bits to compute GIP, which shows that the lower bound in Theorem 3 cannot be improved significantly.

One can find several applications of Theorem 3 in [BNS] (e.g. for Turing-machine simulation trade–offs).
Goldmann and *Håstad* [GH] found a surprising application of Theorem 3 to circuit–complexity.

In [GH], depth–3 threshold circuits are considered, with fan– in on the lowest lewel bounded by $k-1$, and it is shown, that the size of that circuits, computing GIP(A), should be exponential in n.

For the significance of this result it is worth mentioning, that no superpolynomial lower bound is known for the sizes of the depth–3 threshold circuits (without fan–in constraint), which compute a function in **NP**.

Our basic strategy for proving exponential lower bounds for circuit sizes is the same as the strategy of *Goldmann* and *Håstad* [GH]: first, it is assumed that a circuit of a given type and size M computes GIP(A). Then we show a k–party protocol, where all the players know the circuit, and which computes the output of the circuit (i.e. GIP(A)), with communicating about $O(\log M)$ bits. From Theorem 3, $O(\log M) \geq n/4^k$, which yields an exponential lower bound to M.

We apply this strategy first to the following families of circuits: Let C' denote a family of depth–4 circuits $C'_{n,k}$, where n, k are positive integers, and $C'_{n,k}$ computes GIP(A) for any $A \in \{0,1\}^{n\times k}$. On the top of $C'_{n,k}$ an unweighted threshold gate T_q is situated; the input wires of T_q is connected to subcircuits $C^{(m_i)}_{i,n,k}$, for $i = 1, 2, ..., z$, where $k \equiv m_i \pmod{2m_i}$ are satisfied, and m_i is odd positive integer, for $i = 1, 2, ..., z$. For each i, $C^{(m_i)}_{i,n,k}$ is a depth–3 circuit, with an arbitrary SYMMETRIC gate at the top (level 3), and MOD_{m_i} gates of fan–in k on level 2. Moreover, the k input wires of MOD_{m_i} gate G are connected to k gates $G_1, G_2, ..., G_k$ of arbitrary type on level 1, where G_j may depend only on the variables of column A_j of matrix A. On level 0, there are the variables A^ℓ_j and their negations.

We prove in section 3:

Theorem 20. *Suppose that members of circuit family C' computes GIP(A). Then the size of $C'_{n,k} \in C'$ is exponential in n.*

Remark. *The constraint of fan–in k on level 2 is not unreasonably strong in the case of function GIP, since a depth–2 circuit, with a PARITY gate (a SYMMETRIC gate) at the top, and AND–gates (one for each row of A) of fan–in k on level 1 computes GIP(A) with size $n+1$. Theorem 20 shows, that if we exchange the AND-gates on level 1 to MOD_k gates, (substituting $m = k$ in Theorem 20), for an odd k, then these gates are 1 exactly when all of their input-wires are 0 or all of them is 1. This "small" change blows up the size of the circuit exponentially, even when a MAJORITY gate is allowed to put above the symmetric gates.*

By our knowledge, this is the first non-trivial lower bound result for circuits containing MOD_m gates for composite m. Results of *Razborov* [R1] and *Smolensky* [Sm] gives exponential lower bounds when m is prime.

When the modulus is 2, we can get a result without unnatural restrictions:

Theorem 21. *Suppose that family C'' of depth–3 circuits $C''_{n,k}$ computes GIP(A) for any $A \in \{0,1\}^{n\times k}$, where*

–$C''_{n,k}$ has an unweighted threshold gate at the top,

–$MOD_{2^{k-1}}$ gates on the second, and MOD_2 gates on the first level,

–variables A^j_i with their negations on level 0.

Then the size of $C''_{n,k}$ is exponential in n. ▮

The key step in the proofs of Theorems 20 and 21 are the constructions of some protocols, which computes the output of a circuit with few communicated bits. Considering these protocols, one can find several very interesting exponential gaps between the communication complexities of *extremely* closely related functions. To describe our results, we define two complexity classes:

Definition 4. *Let* $G = \{g_{n,k} \mid n,k \in \mathbf{N},\ g_{n,k} : \{0,1\}^{n\times k} \to \mathbf{N}\}$, *where* $\mathbf{N}$ *denotes the set of natural numbers. We say that a* G *is multi–party easy if* $\exists c > 0$ *such that for all* $g_{n,k} \in G$ $C^{(k)}(g_{n,k}) \leq 2^{ck} \log n$. *Let* **ME** *denote the family of all multi–party easy sets. We say that* G *is multi–party hard, if* $\exists c' > 0$ *such that for all* $g_{n,k} \in G$ $C^{(k)}(g_{n,k}) \geq n2^{-c'k}$. *Let* **MH** *denote the family of all multi–party hard sets.*

Theorem 3 shows that GIP is in **MH**.

In Section 2 we show several surprising theorems about the membership in the classes **MH** and **ME**, and these theorems will be the basis of proving the circuit results:

Theorem 11. *Let* m *be an odd, positive integer, let* $0 \leq \ell \leq m-1$, *and* $k \equiv m + 2\ell \pmod{2m}$. *Let* $A \in \{0,1\}^{n\times k}$. *Then the number of those rows of* A *which are congruent to* $\ell \pmod{m}$, *is in* **ME**.

With $\ell = 0$ we get that the number of rows divisible by m is in **ME**. However, not every congruence–class can be counted easily, even with the assumptions of Theorem 11:

Corollary 13. *Let* m *be odd, and* $k \equiv m \pmod{2m}$. *Then the number of rows congruent to 1* $\pmod{m}$ *is in* **MH**.

For even m, congruence–class counting is hard:

Theorem 12. *Let* $A \in \{0,1\}^{n\times k}$, *and let* m *be an even positive integer. Then to compute the number of that rows of* A, *which are congruent to* $\ell \pmod{m}$ *is in* **MH**, *for any integer* ℓ.

If $m = 2$, at least a modular result is easy:

Theorem 14. *The function, which is defined to be the number of even rows of* A, mod 2^{k-1}, *is in* **ME**.

From Theorem 3, the number of the all–1 rows is in **MH**.

Corollary 15. *Let* k *be an odd positive integer. The function which gives the number of the all–0 rows plus the number of the all–1 rows of* A *is in* **ME**.

2. The protocol

Definition 5. *Let $A \in \{0,1\}^{n\times k}$, and let $m, z \in \mathbf{N}$. Suppose that $1 \le j \le n$. We say that row A^j is congruent to $z \pmod m$, iff*

$$\sum_{i=1}^{k} A_i^j \equiv z \pmod m.$$

We say that row A^j is divisible by m if it is congruent to $0 \pmod m$.

The goal of the players in protocol **MOD m** is to compute the number of the rows of A in every congruency–class, mod m.

Notation 6. *We denote the elements of vector space $\mathbf{N}^m$ by small–case greek letters, and we index their coordinates from 0 through $m-1$.*

Definition 7. *Let $A \in \{0,1\}^{n\times k}$ and $m \in \mathbf{N}$. Let*

$$\delta^{(m)}(A) = (\delta_0, \delta_1, \ldots, \delta_{m-1})$$

denote a vector where δ_i is the number of that rows of A, which are congruent to $i \pmod m$. Let $v \in \{0,1\}^k$, then $CT(v, A)$ denotes the number of that rows of A, which are equal to v. Let $\mathbf{0} = (0,0,\ldots,0) \in \{0,1\}^k$, and $\mathbf{1} = (1,1,\ldots,1) \in \{0,1\}^k$.

The fundamental strategy of the players in protocol **MOD m** is the following: Player P_i $(1 \le i \le k)$ assumes that column i of A, A_i is the all–1 vector. P_1 communicates the number of rows in separate congruency– classes, and then P_2 corrects him in case of that rows, which begin with 0, instead of the assumed 1. Then P_3 corrects P_2, in case of that rows, which begins with two zeros, and so on, until P_k comes. Then P_k corrects P_{k-1} in case of that rows which begins with $k-1$ zeros. The protocol makes errors only in the case of that rows, for which *neither of the assumptions* were satisfied: the rows with k 0's. Every other row will be counted correctly: since at least one player's assumption was right, he saw the row entirely, and counted it to the proper congruency–class, corrected the errors of the others.

Now we present a more detailed description of the protocol, together with its analysis. (The protocol itself is typesetted in `typewriter font`, while the analytical remarks are in roman)

Protocol MOD m

`P`$_1$ `begins the communication.`

Since P_1 assumes that the first column of A is the all–1 vector, P_1 is assumed to know the entire input, so he can communicate any function of it.

P_1 first communicates α_0, the number of those rows, which are congruent to 0 $\pmod m$, second α_1, the number of rows, congruent to 1 $\pmod m$, ..., and last α_{m-1}, the number of rows, congruent to $m-1 \pmod m$.

So P_1 communicates vector

$$\alpha = (\alpha_0, \alpha_1, ..., \alpha_{m-1})$$

of length $O(m \log n)$. Let us note that

$$\sum_{\ell=0}^{m-1} \alpha_\ell = n.$$

P_1 correctly counts that rows, which begins with a 1, but if a row begins with a 0, and P_1 counted it to α_ℓ then correctly it would have been counted to $\alpha_{(\ell-1) \bmod m}$.

P_2 communicates next.

Since P_1 already advertised vector α, the task of P_2 is only to correct the errors made by P_1. P_2 knows where P_1 made an error: those rows begin with 0.

Suppose that row A^j begins with a 0, and P_2

--- using his assumption that A_2 is the all–1 vector ---

sees that A^j is congruent to $\ell \pmod m$.

P_2 knows, that P_1 assumed that the first entry of A^j is 1, and assumes that the second entry in A^j is also 1, so P_2 assumes that P_1 counted erroneously A^j to that rows, which are congruent to $\ell + 1 \pmod m$.

P_2 subtracts 1 from the number $\alpha_{\ell+1 \pmod m}$ and adds 1 to α_ℓ. P_2 repeats this for all rows, beginning with 0, but communicates only the vector--sum of the corrections:

$$\beta^{(2)} = (\beta_0^{(2)}, \beta_1^{(2)}, ..., \beta_{m-1}^{(2)}),$$

where $\beta_i^{(2)}$ the number of those rows which begin with 0 and P_2 sees them to be congruent to i, minus the number of those rows, which begin with 0 and P_2 sees them to be congruent to $i-1 \pmod m$.

Note that

$$\sum_{\ell=0}^{m-1} \beta_\ell^{(2)} = 0,$$

and $\beta^{(2)}$ can be communicated with $O(m \log n)$ bits.

P_3, after that $P_4, ..., P_{i-1}$ communicates $(i \le k)$, and

P_i communicates next.

The task of P_i is to correct errors, committed by P_{i-1}. Until now, all of the rows were counted correctly, which contain at least one bit 1 in the first $i-1$ positions.

P_i deals only with rows which begin with $i - 1$ zeros. Suppose that a row, A^j, begins with $i-1$ zeros, and P_i sees it to be congruent to $\ell \pmod m$.

Then P_i assumes that P_{i-1} has seen A^j to be congruent with $\ell+1$, so he corrects P_{i-1}. However, so far P_{i-1} have corrected $P_{i-2}, P_{i-3}, ..., P_1$ with an assumption that $A^j_{i-1} = 1$, but P_i knows that $A^j_{i-1} = 0$, so P_i should also *correct the corrections* of P_{i-1}.

Let P_i communicate

$$\beta^{(i)} = (\beta_0^{(i)}, \beta_1^{(i)}, ..., \beta_{m-1}^{(i)}),$$

the vector--sum of the correction vectors.

Since P_i knows the strategy of the other players, and assumes to know the whole input, he can simulate their computation, and can correct their errors. So P_i computes $\beta^{(i)}$, and can communicate it with $O(m \log n)$ bits. Let us note again, that

$$\sum_{\ell=0}^{m-1} \beta_\ell^{(i)} = 0.$$

When P_k has communicated $\beta^{(k)}$, all players compute -- privately -- the vector--sum

$$\gamma = \alpha + \sum_{i=2}^{k} \beta^{(i)}.$$

End of protocol MOD m

The players of this protocol uses $O(mk \log n)$ bits of communication.

Let us observe that if no row of A is equal to $\mathbf{0}$, then

$$\gamma = \delta^{(m)}(A),$$

since every row is correctly counted by one player, and that player corrected all the previous errors, for that row.

Notation 8. *Let*

$$\Pi = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & 0 & \dots & 0 & 0 \end{pmatrix}$$

the $m \times m$ cyclic–right–shift permutation–matrix.

Lemma 9.

(1) $$\gamma = \delta^{(m)}(A) + CT(\mathbf{0}, A)(\mu - \nu)$$

where $\nu = (1, 0, 0, ..., 0)$, *and* $\mu = \nu - \nu(I - \Pi)^k$.

Proof. In protocol **MOD m** players count correctly all the rows, except those, which are equal to **0**. In fact, they never count the **0**–rows, since no player's assumption is compatible with **0**. Player P_i for each row **0** compute some vector $\mu^{(i)}$, which they add up to μ at the end:

$$\mu = \sum_{i=1}^{k} \mu^{(i)},$$

instead of the correct $\nu = (1, 0, 0, ..., 0)$, this shows the correctness of equation (1).

Our remaining task is to compute μ.

P_1 counts **0** to rows, congruent to $1 \pmod m$, so he adds the following $\mu^{(1)}$ to its communicated vector α, for each row **0**:

$$\mu^{(1)} = (0, 1, 0, ..., 0).$$

P_2 also counts **0** to rows, congruent to $1 \pmod m$, and he assumes, that P_1 counted the row to the rows, congruent to $2 \pmod m$. So P_2 adds

$$\mu^{(2)} = (0, 1, 0, ..., 0) - (0, 0, 1, 0, ..., 0) = \mu^{(1)} - \mu^{(1)}\Pi = \mu^{(1)}(I - \Pi)$$

to its $\beta^{(2)}$, where I denotes the $m \times m$ unit–matrix.

Now let $2 \le i \le k - 1$, and suppose that

(2). $$\mu^{(i)} = \mu^{(1)}(I - \Pi)^{i-1}$$

We state that P_{i+1} communicates $\mu^{(i)}$, the same corrections to $P_1, P_2, ..., P_{i-1}$ as P_i has communicated, since P_i assumes that bit i is the only 1–bit in the row, while P_{i+1} assumes that bit $i + 1$ is the only 1–bit in the row, and these assumptions are equivalent, from the viewpoints of $P_1, P_2, ..., P_{i-1}$, so when P_i and P_{i+1} correct them, they must communicate the same number.

However, P_{i+1} corrects P_i, too. P_{i+1} assumes that P_i sees one more bit than himself, so P_{i+1} assumes that P_i has computed the correction–vectors for $P_1, P_2, ..., P_{i-1}$ as himself, but with a circular right–shift. So to correct P_i, P_{i+1} should subtract $\mu^{(i)}\Pi$ from $\mu^{(i)}$:

$$\mu^{(i+1)} = \mu^{(i)} - \mu^{(i)}\Pi = \mu^{(1)}(I - \Pi)^i.$$

We have got that

$$\mu = \sum_{i=1}^{k} \mu^{(i)} = \mu^{(1)}((I-\Pi)^0 + (I-\Pi)^1 + ... + (I-\Pi)^{k-1}).$$

Using that $\mu^{(1)} = \nu\Pi$,

$$\mu = \nu\Pi((I-\Pi)^0 + (I-\Pi)^1 + ... + (I-\Pi)^{k-1}) \tag{3}$$

Multiplying both sides of (3) from right by $(I-\Pi) - I = -\Pi$:

$$-\mu\Pi = \nu\Pi((I-\Pi)^k - I),$$

since Π commutes with its powers,

$$-\mu\Pi = \nu((I-\Pi)^k - I)\Pi. \tag{4}$$

Multiplying both sides of (4) with $-\Pi^{-1}$, from right:

$$\mu = \nu - \nu(I-\Pi)^k,$$

and this equation proves the theorem. ∎

Lemma 10.

$$\delta^{(m)}(A) = \gamma - CT(\mathbf{0}, A)\theta,$$

where $\theta = (\theta_0, \theta_1, ..., \theta_{m-1})$, *and*

$$\theta_j = \sum_{\substack{0 \le i \le k \\ i \equiv j \pmod m}} (-1)^i \binom{k}{i}.$$

Proof. From the binomial theorem,

$$(I-\Pi)^k = \binom{k}{0}I - \binom{k}{1}\Pi + \binom{k}{2}\Pi^2 - ... + (-1)^k \binom{k}{k}\Pi^k.$$

Since $\Pi^m = I$, we can write

$$(I-\Pi)^k = \sum_{\ell=0}^{m-1} \Pi^\ell \left(\sum_{\substack{0 \le i \le k \\ i \equiv \ell \pmod m}} (-1)^i \binom{k}{i} \right), \tag{5}$$

It is easy to see, if a matrix is multiplied by ν from the left, the result is the first row of the matrix. When a row-vector is multiplied by Π the effect is the circular right–shift of the coordinates; this also holds for the first rows of the powers of Π: the first row of I is $1, 0, ..., 0$, the first row of Π is $0, 1, 0, ..., 0$, the first row of Π^2 is $0, 0, 1, 0, ..., 0, ...$, the first row of Π^{m-1} is $0, ..., 0, 1$.

From (5) we got:

$$\nu(I - \Pi)^k = (\theta_0, \theta_1, ..., \theta_{m-1}) = \theta, \tag{6}$$

where

$$\theta_j = \sum_{\substack{0 \le i \le k \\ i \equiv j \pmod m}} (-1)^i \binom{k}{i}.$$

Lemma 9 together with (6) implies Lemma 10. ∎

Theorem 11. *Let m be an odd, positive integer, let $0 \le \ell \le m - 1$, and $k \equiv m + 2\ell \pmod{2m}$. Let $A \in \{0,1\}^{n \times k}$. Then the number of those rows of A which are congruent to $\ell \pmod m$, is in* **ME**.

Proof. By Lemma 10,

$$\theta_\ell = \sum_{\substack{0 \le i \le k \\ i \equiv \ell \pmod m}} (-1)^i \binom{k}{i} = \sum_{\substack{0 \le i \le k \\ i \equiv \ell \pmod m \\ i \text{ even}}} \binom{k}{i} - \sum_{\substack{0 \le i \le k \\ i \equiv \ell \pmod m \\ i \text{ odd}}} \binom{k}{i} =$$

$$= \sum_{\substack{0 \le i \le k \\ i \equiv \ell \pmod m \\ i \text{ even}}} \binom{k}{i} - \sum_{\substack{0 \le i \le k \\ i \equiv \ell \pmod m \\ i \text{ odd}}} \binom{k}{k-i} = \sum_{\substack{0 \le i \le k \\ i \equiv \ell \pmod m \\ i \text{ even}}} \binom{k}{i} - \sum_{\substack{0 \le j \le k \\ j \equiv \ell \pmod m \\ j \text{ even}}} \binom{k}{j} = 0,$$

since k is odd, and $k - i \equiv \ell \pmod m$.

So, $\gamma_\ell = \delta_\ell^{(m)}(A)$, and since protocol **MOD m** computes γ in **ME**, we are done. ∎

Theorem 12. *Let $A \in \{0,1\}^{n \times k}$, and let m be an even positive integer. Then to compute the number of that rows of A, which are congruent to $\ell \pmod m$ is in* **MH**, *for any integer ℓ.*

Proof. We may assume that $0 \le \ell \le m - 1$. From Lemma 10,

$$\delta_\ell^{(m)} = \gamma_\ell - CT(\mathbf{0}, A)\theta_\ell, \tag{7}$$

and

$$\theta_\ell = \sum_{\substack{0 \le i \le k \\ i \equiv \ell \pmod m}} (-1)^i \binom{k}{i} \ne 0$$

since every summand is of the same sign. k players, who compute $\delta_\ell^{(m)}$ with communicating c bits can compute $CT(\mathbf{0}, A)$ with communicating $c + O(km \log n)$ bits, using protocol **MOD m**, and equation (7). However, Theorem 3 shows (interchanging the roles of bits 1 and 0 in its proof), that computing $CT(\mathbf{0}, A)$ needs $\Omega(n/4^k)$ bits to communicate, and since any player can compute θ without any communication, we are done. ∎

Corollary 13. *Let m be odd, and $k \equiv m \pmod{2m}$. Then the number of rows congruent to 1 $\pmod m$ is in* **MH**.

Proof. As in the proof of Theorem 12, we need to prove that $\theta_1 \neq 0$. Let us suppose that $\theta_0 = \theta_1 = 0$. Using Lemma 10, and the Pascal-triangle equality for binomial coefficients, we get:

$$\binom{k+1}{1} + \binom{k+1}{2m+1} + \binom{k+1}{4m+1} + \dots + \binom{k+1}{2sm+1} =$$

$$= \binom{k+1}{m+1} + \binom{k+1}{3m+1} + \dots + \binom{k+1}{(2s+1)m+1},$$

where $k = (2s+1)m$, and we assume that s is even. From here:

$$\binom{k+1}{1} - \binom{k+1}{0} + \binom{k+1}{2m+1} - \binom{k+1}{2m} + \dots + \binom{k+1}{sm+1} - \binom{k+1}{sm} =$$

$$\binom{k+1}{m+1} - \binom{k+1}{m} + \binom{k+1}{3m+1} - \binom{k+1}{3m} + \dots + \binom{k+1}{(s-1)m+1} - \binom{k+1}{(s-1)m}.$$

Every difference, counting from right to left, at the left side of the previous equation is strictly greater than the appropriate difference at the same position at the right side, so the equation cannot be true, which proves our statement. The proof is similar for odd s. ∎

Let $A \in \{0,1\}^{n \times k}$. A row of A is called *even*, if it is divisible by 2. Theorem 12 shows, that the number of even rows of A is in **MH**. However:

Theorem 14. *The function, which is defined to be the number of even rows of A,* mod 2^{k-1}, *is in* **ME**.

Proof. Protocol **MOD m**, with $m = 2$, computes vector

$$\gamma = \delta^{(2)}(A) + CT(\mathbf{0}, A) \left(\sum_{\substack{0 \le i \le k \\ i \text{ even}}} \binom{k}{i}, - \sum_{\substack{0 \le i \le k \\ i \text{ odd}}} \binom{k}{i} \right) = \delta^{(2)}(A) + CT(\mathbf{0}, A)(2^{k-1}, -2^{k-1}).$$

The first coordinate of γ is congruent to $\delta_1^{(2)} \pmod{2^{k-1}}$, and this proves the statement. ∎

From Theorem 3, the number of the all–1 rows is in **MH**.

Corollary 15. *Let m be an odd positive integer. The function which gives the number of the all–0 rows plus the number of the all–1 rows of A is in* **ME**.

Proof. *Let $m = k$ and $\ell = 0$ in Theorem 11.* ■

3. Circuits with mod m gates

Definition 16. *Let C^* be a family of depth–3 circuits $C_{n,k}^{(m)}$, where n and k are positive integers, m is odd and positive, and $k \equiv m \pmod{2m}$ is also satisfied. Moreover*

– the input of $C_{n,k}^{(m)}$ is A for $A \in \{0,1\}^{n\times k}$,

– on the bottom level (level 0) situated the variables A_j^i, with their negations;

– on the top (level 3), there is a symmetric gate,

– there are MOD_m gates of fan–in k on the second level;

– the k input wires of MOD_m gate G are connected to k gates of arbitrary type $G_1, G_2, ..., G_k$, situated on the first level, where G_i may depend only on the variables of column A_i of matrix A.

Theorem 17. *Suppose that members of the circuit family C^* computes GIP(A). Then the size of $C_{n,k}^{(m)}$ is exponential in n.*

Proof. Let us consider circuit $C_{n,k}^{(m)}$, computing GIP(A), $A \in \{0,1\}^{n\times k}$. Let us consider k players, such that player i knows every column of A, except column i, for $i = 1,2,...,k$, and suppose that all the players know circuit $C_{n,k}^{(m)}$. On the top of the circuit there is a symmetric gate, and the output of that gate depends only on the number of MOD_m gates, evaluated to 1, on level 2.

Players will collectively compute the number of MOD_m gates, evaluated to 1. To do this, first they – individually, without any communication – build a matrix B. B has k columns, and each row of it corresponds to one of the MOD_m gates of the circuit; suppose that row B^i corresponds to a MOD_m gate G, and G has input–gates $G_1, G_2, ..., G_k$ on the first level. Then let B_j^i be equal to the output of G_j.

Since G_j depends only on the variables of column j of A, Player j knows all the columns of B, except column j, B_j.

Let us observe that B^i is divisible by m exactly when G is evaluated to 1.

Let the size of $C_{n,k}^{(m)}$ be N. B has at most N rows. From Theorem 11, protocol **MOD m** computes the number of rows B, divisible by m, with communicating

$$O(mk \log N)$$

bits, and Theorem 3 shows that to compute GIP(A) the players should communicate

$$\Omega\left(\frac{n}{4^k}\right)$$

bits, so

$$O(mk\log N) = \Omega\left(\frac{n}{4^k}\right)$$

or

$$N \geq \exp\left(c\frac{n}{4^k mk}\right).$$

∎

The next theorem does not have unnatural restrictions, but we can prove it only with modulus 2:

Theorem 18. *Suppose that family C^{**} of depth–2 circuits $C_{n,k}$ computes GIP(A) for $A \in \{0,1\}^{n\times k}$, where*

– $C_{n,k}$ has a $MOD_{2^{k-1}}$ gate at the top, and MOD_2 gates at the first level.

Then the size of $C_{n,k}$ is exponential in n.

Proof. Suppose that there are N MOD_2 gates in $C_{n,k}$. As in the proof of Theorem 17, players build a matrix $B \in \{0,1\}^{n\times k}$ in the following way: each MOD_2 gate G is corresponded to a row of B, B^i, such that entry B^i_j is the mod 2 sum of that input–variables of G, which are also in column j of A. Let us observe that G is evaluated to 1 iff the sum of B_i is even. Using Theorem 14, the result follows. ∎

With standard techniques of [HMPST] and [GH], we can generalize Theorems 17 and 18:

Definition 19. *Let C' denote a family of depth–4 circuits $C'_{n,k}$, where n, k are positive integers, and $C'_{n,k}$ computes GIP(A) for any $A \in \{0,1\}^{n\times k}$. On the top of $C'_{n,k}$ an unweighted threshold gate T_q is situated; the input wires of T_q is connected to subcircuits $C^{(m_i)}_{i,n,k}$, for $i = 1, 2, ..., z$, where $C^{(m_i)}_{i,n,k}$ has the same form as $C^{(m)}_{n,k}$ with $m = m_i$ in Definition 16.*

Theorem 20. *Suppose that members of circuit family C' computes GIP(A). Then the size of $C'_{n,k} \in C'$ is exponential in n.*

Proof. If $C'_{n,k}$ of size N computes GIP(A) then – by Lemma 2 of [GH] or Lemma 3.3. of [HMPST] – at least one of the depth–3 subcircuits computes GIP(A) or 1–GIP(A) correctly on at least

$$\frac{1}{2} + \frac{1}{2N}$$

fraction of the inputs. Theorem 17 shows that the output of that depth–3 subcircuit can be computed with $O(m_i k \log N)$ communication. From Theorem 3, with $\varepsilon = 1/2N$:

$$O(m_i k \log N) = \Omega\left(\frac{n}{4^k} - \log N\right),$$

and this completes the proof. ∎

Theorem 21. *Suppose that family C'' of depth–3 circuits $C''_{n,k}$ computes GIP(A) for any $A \in \{0,1\}^{n\times k}$, where*

–$C''_{n,k}$ has an unweighted threshold gate at the top,

–$MOD_{2^{k-1}}$ gates on the second, and MOD_2 gates on the first level,

–variables A_i^j with their negations on level 0.

Then the size of $C''_{n,k}$ is exponential in n.

Proof. The proof follows from Theorems 18 and 3, exactly like Theorem 20 from Theorems 17 and 3. ∎

4. Open Problems

– Only 0–1 matrices can be handled by our protocol **MOD m**. We would get more attractive circuit applications if, for example, the number of rows, divisible by 3, had been computed in **ME**, for a matrix with entries 0, 1 and 2. This would show that any circuit of depth 3, with a threshold gate at the top, arbitrary symmetric gates (e.g. MOD_2) gates at level 2 and MOD_3 gates at level 1 need exponential size to compute GIP.

– All of our protocols are oblivious in the sense that the communication and the numbers communicated by the players do not depend on the messages, communicated earlier. It is not clear, if the non–oblivious communication is stronger or not than the oblivious communication in this model.

– In our protocol the players play only one round – everybody speaks at most once. Are the two– or more–round protocols stronger than the one–round ones?

REFERENCES

[BNS] L. Babai, N. Nisan, M. Szegedy: Multiparty Protocols and Pseudorandom Sequences, Proc. 21st ACM STOC, 1989, pp. 1-11.

[CFL] A. K. Chandra, M. L. Furst, R. J. Lipton: Multi-party Protocols, Proc. 15th ACM STOC, 1983, pp. 94–99.

[G] V. Grolmusz: The BNS Lower Bound for Multi–Party Protocols is Nearly Optimal, to be appeared in "Information and Computation".

[GH] M. Goldmann, J. Håstad: On the Power of Small–Depth Threshold Circuits, 31st IEEE FOCS, 1990, pp. 610–618.

[HMPST] A. Hajnal, W. Maass, P. Pudlak, M. Szegedy, G. Turán: Threshold Circuits of Bounded Depth, Proc. 28th IEEE FOCS, 1987, pp. 99–110.

[KS] B. Kalyanosundaram, G. Snitger: The Probabilistic Communication Complexity of Set Intersection, Proc. Structure in Complexity Theory, 1987, pp. 41–49.

[KW] M. Karchmer, A. Wigderson: Monotone Circuits for Connectivity Require Super–Logarithmic Depth, Proc. 20th ACM STOC, 1988, pp.

[L] L. Lovász: Communication Complexity: A Survey, Technical Report, CS–TR–204–89, Princeton University, 1989.

[R] A. A. Razborov: On the Distributional Complexity of Disjointness, preprint

[R1] A. A. Razborov: Lower Bounds on the Size of Bounded Depth Networks Over a Complete Basis with Logical Addition, (in Russian), Mat. Zametki, 41 (1987), 598–607

[RW1] R. Raz, A. Wigderson: Probabilistic Communication Complexity of Boolean Relations. Proc. 30th IEEE FOCS, 1989, pp.

[RW2] R. Raz, A. Wigderson: Monotone Circuits for Matching Require Linear Depth. 22nd ACM STOC, pp. 287–292.

[S] M. Szegedy: Functions with Bounded Symmetric Communication Complexity and Circuits with MOD m Gates, Proc. 22nd ACM STOC, pp. 278–286.

[Sm] R. Smolensky, Algebraic Methods in the Theory of Lower Bounds for Boolean Circuit Complexity, Proc. 19th ACM FOCS, pp. 77-82, (1987).

[Y1] A.C. Yao: Some Complexity Questions Related to Distributive Computing, Proc. 11th ACM STOC, 1979, pp. 209–213.

[Y2] A.C. Yao: Circuits and Local Computation, Proc. 21st ACM STOC, 1989, pp. 186–196

[Y3] A. C. Yao: On ACC and Threshold Circuits, 31st IEEE FOCS, 1990, pp. 619–627.