Electronic Colloquium on Computational Complexity - Reports Series 1996 - available via:

ECCC
TR96-017

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'

The "Log Rank" Conjecture for Modular Communication Complexity

Christoph Meinel
FB IV - Informatik
Universität Trier
D–54286 Trier

Stephan Waack
Institut für Numerische
und Angewandte Mathematik
Georg–August–Universität Göttingen
D–37027 Göttingen

Abstract

The "log rank" conjecture consists in the question how exact the deterministic communication complexity of a problem can be determinied in terms of algebraic invarants of the communication matrix of this problem. In the following, we answer this question in the context of modular communication complexity. We show that the modular communication complexity can be exactly characterised in terms of the logarithm of a certain rigidity function of the communication matrix. Thus, we are able to exactly determine the modular communication complexity of several problems, such as, e.g., set disjointness, comparability, and undirected graph connectivity. From the obtained bounds for the modular communication complexity we can conclude exponential lower bounds on the size of depth two circuits having arbitary symmetric gates at the bottom level and a MOD_m–gate at the top.

Area: Computational Complexity

Keywords: Communication Protocols, Modular Acception Modes, Depth Two Circuits Having Arbitary Symmetric Gates in the Bottom Level, and a MOD_m–Gate at the Top, Invertible Incidence Functions, Undirected Graph Connectivity.

Introduction

In the basic model of communication complexity the number of bits is investigated which have to be exchanged in order to enable two processors $\mathcal{P}_1$ and $\mathcal{P}_2$ (with unlimited computational power) which have access to input data from different finite sets S_1 and S_2 to compute functions $f : S_1 \times S_2 \to \{0,1\}$. The deterministic communication complexity $\mathrm{CC}(f)$ of such an f is the minimum number of bits the processors have to exchange for the worst case input. This model, first introduced by Yao [19], has many applications in different branches of complexity theory and has been studied in many papers. (See [6] for a survey.)

For the communication matrix $\mathcal{M}^f_{s_1,s_2} = f(s_1, s_2)$ of the problem f given in distributed form we have the following inequality,

$$\log_2\left(\mathrm{rk}_{\mathbf{R}}\left(\mathcal{M}^f\right)\right) \leq \mathrm{CC}(f),$$

due to Mehlhorn and Schmidt [8]. The question, which is called the "log rank" conjecture (see [7]), is whether this lower bound is sharp. Considering this conjecture several authors have obtained interesting separation results which show gaps between $CC(f)$ and $\log_2\left(\mathrm{rk}_{\mathbf{R}}\left(\mathcal{M}^f\right)\right)$. Recently, Raz and Spieker [12] proved that there is a nonconstant gap. (We review this result shortly in Section 4.)

We shall study an analogous problem in the case of modular communication complexity. The investigation of the modular communication complexity is motivated by circuits of finite depth having MOD_m–gates (see [20], and Section 1, Theorem 1). We succeed in characterizing the modular communication complexity in terms of the logarithm of a certain rigidity function of the communication matrix. The consideration of such regidity functions came into prominence in the last few years. Lower bounds on rigidity functions of explicit matrices proved useful in algebraic circuit complexity theory (see [9], [15]), for branching programs (see [1]) and threshold circuits (see [4]), and for communication complexity (see [11]). Lokam [5] unifies and strengthens many of these results.

We adopt and weaken a rigidity function of [4] which we call then rigidity rank of a matrix, abbreviated $\mathrm{rrk}_m M$. First, we show that the modular communication complexity MOD_m-$\mathrm{CC}(f)$ is equal to $\Theta(\log_2 \mathrm{rrk}_m \mathcal{M}^f)$ (see Section 2). In contrast to the variation rank of [4], which could only be computed for the identity matrix, we determine the rigidity rank in many more cases (see Sections 3 and 4). Thus, we can determine exactly the modular communication complexity of the underlying problems (e.g., set disjointness, comparability, undirected graph connectivity). Due to the close connection between modular communication complexity and the size of depth two circuits having arbitary symmetric gates in the bottom level, and a MOD_m–gate at the top (see Theorem 1), finally, we can conclude some exponential lower bounds on the circuit size for these problems.

1 The Computational Model and its Motivation

Assume that we are given a function $f : S_1 \times S_2 \to \{0, 1\}$ in distributed form, where the S_i are finite sets. *A communication protocol P of two processors $\mathcal{P}_1$ and $\mathcal{P}_2$ of length L* computing a function f is, as usual, defined as follows. In order to compute $f(s_1, s_2)$, the processor $\mathcal{P}_i$ has the element $s_i \in S_i$ as input, for $i = 1, 2$. The one processor is not allowed to read the input of the other directly. They have to communicate via a common communication tape. The computation of the whole structure is going on in *rounds*. Starting with $\mathcal{P}_1$, the processors write alternatingly bits on the communication tape. These bits depend on the input available to the processor which is to move and on the bits already written on the communication tape before. If the last bit written onto the communication tape is "1" or "0", the computation is called *accepting* or *rejecting*, respectively. Under the usual assumption of prefix–freeness

for messages, the co-operative computation can be thought of as to be a Boolean string, the string of the bits communicated. The length of the string is the *communication complexity* of the computation. Since we consider the worst–case complexity in this paper, we assume without loss of generality that all computations of a protocol which influence the final output are of equal length, say L.

If the processors are nondeterministic, the outcome of one computation is, of course, not identical to the final vote. The output of a protocol P, for a given input (s_1, s_2), depends only on the numbers $\mathcal{M}^{P,acc}_{s_1,s_2}$ and $\mathcal{M}^{P,rej}_{s_1,s_2}$ of accepting and rejecting computations performed by the protocol accessing this input. How are these numbers, considered as matrices, related to the length of the protocol? The answer is given by the following

Lemma 1 *If P is a protocol of length L, and if R is any semiring, then we have the inequalities*

$$\begin{aligned} \mathrm{rk}_R(\mathcal{M}^{P,acc}) &\leq 2^{L-1}, \\ \mathrm{rk}_R(\mathcal{M}^{P,rej}) &\leq 2^{L-1}. \end{aligned}$$

The proof can be done in a straightforward way (see, e.g., [2]). As usual, *the R–rank of a $N_1 \times N_2$–matrix A over R*, which we denote by $\mathrm{rk}_R A$, is defined to be the minimal number K such that $A = B \cdot C$, where B is a $N_1 \times K$–matrix and C is a $K \times N_2$–matrix over R.

This dependency between the matrices $\mathcal{M}^{P,acc}$ and $\mathcal{M}^{P,rej}$ and the final outcome of the protocol is formally given by a function $\mu : \mathbb{N}^2 \to \{0,1\}$, the *counting acception mode*. More precisely, if $\mathcal{M}^f_{s_1,s_2} := f(s_1, s_2)$ is the matrix associated with the problem, the *communication matrix*, then

$$\mathcal{M}^f_{s_1,s_2} = \mu\left(\mathcal{M}^{P,acc}_{s_1,s_2}, \mathcal{M}^{P,rej}_{s_1,s_2}\right).$$

A protocol P equipped with a counting acception mode μ is called a μ-protocol. The function $f : S_1 \times S_2 \to \{0,1\}$ computed by a μ-protocol P is called to be *μ–computed*. If a protocol μ–computing a function f is chosen in such a way that its length is minimal, we define the number $\mu\text{-CC}(f) := L$ to be the *μ–communication complexity of the function f*.

In this paper, we discuss in particular *the modular acception modes* MOD_m in which the protocol accepts an input if the number of accepting computations is not equal to 0 modulo m:

$$\mathrm{MOD}_m(n_1, n_2) = 1 \quad \Longleftrightarrow \quad n_1 \not\equiv 0 \pmod{m}.$$

We study the asymptotic modular communication $\mathrm{MOD}_m\text{-CC}(f_{2n})$ complexity of the sequence $(f_{2n} : \Sigma^n \times \Sigma^n \to \{0,1\})_{n \in \mathbb{N}}$.

We shall see later on that it is possible to settle down in a first step to modes MOD_m, where m is square free. In order to do so, we need some constructions which are in some sense polynomials of protocols.

Definition 1 *Let $S_1 \times S_2$ be as before, and let $P_1, \ldots, P_m$, and P be protocols of length $L_1, \ldots, L_m$, and L.*

1. *We define the product P_1P_2 of length $L_1 + L_2$ as follows. Given an input $(s_1, s_2) \in S_1 \times S_2$, the protocol P_1P_2 proceeds in the same way as P_1 does while the first L_1 bits are being exchanged, and according to P_2 then. But we make the following restriction in the last round: For $i = 1, 2$, if $\gamma^{(i)} \in \{0,1\}^{L_i}$ are r_i–round computations of the protocols P_i on (s_1, s_2), if $\gamma^{(i)}_{last} \in \{0,1\}$ are the last bits of $\gamma^{(i)}$, and if $\gamma^{(1)}_{last} \neq \gamma^{(2)}_{last}$, then the computation* is stopped without result *before the round $r_1 + r_2$. (This means, each computation of length $L_1 + L_2$ is either a concatenation of two accepting or of two rejecting computations.)*

2. *For $a_i \in \mathbb{N}$, $P_1^{a_1} \ldots P_m^{a_m}$ is defined on the basis of item 1 in the canonical way. (We assume P_i^0 to be the unique protocol of length 0.)*

3. *The protocol $\binom{P}{k}$, for $k \in \mathbb{N}$, of length kL on an input $(s_1, s_2) \in S_1 \times S_2$ works as follows. It proceeds as P^k does but with the following modification in the last round: Let $\{0,1\}^L$ be arbitrarily but fixed totally ordered. If $\gamma^{(i)} \in \{0,1\}^L$ are r_i–round computations of the protocol P_i on (s_1, s_2) such that $\gamma^{(1)} \ldots \gamma^{(k)}$ is a r-round computation of P^k, where $r = r_1 + \ldots + r_k$, then the computation is stopped without result unless $\gamma^{(1)} < \ldots < \gamma^{(k)}$.*

4. *Let $a_1, \ldots, a_m$ be positive natural numbers. The sum $(a_1P_1 + \ldots + a_mP_m)$ is the protocol of length $\lceil \log_2 (\sum_{i=1}^m a_i) \rceil + \max\{L_i \mid i = 1, \ldots, m\}$ which is defined on $(s_1, s_2) \in S_1 \times S_2$ by the following rules: First, the protocols are assumed to be prolonged to the length $\max\{L_i \mid i = 1, \ldots, m\}$. Second, processor $\mathcal{P}_1$ sheds nondeterministically a number $j \in \{1, 2, \ldots, \sum_{i=1}^m a_i\}$ as a message of length $\lceil \log_2 (\sum_{i=1}^m a_i) \rceil$. Third, $\mathcal{P}_2$ and $\mathcal{P}_1$ proceed in the same way as $\mathcal{P}_1$ and $\mathcal{P}_2$ do according to protocol P_i, provided that $\sum_{\nu=1}^{i-1} a_\nu + 1 \leq j \leq \sum_{\nu=1}^{i} a_\nu$.*

A straightforward calculation reveals the next lemma.

Lemma 2 *The situation is assumed to be as in Definition 1. Then, for all $(s_1, s_2) \in S_1 \times S_2$,*

$$\begin{aligned} \mathcal{M}_{s_1,s_2}^{P_1^{a_1} \ldots P_m^{a_m}, acc} &= \prod_{i=1}^m \left(\mathcal{M}_{s_1,s_2}^{P_i,acc}\right)^{a_i}, \\ \mathcal{M}_{s_1,s_2}^{\binom{P}{k},acc} &= \binom{\mathcal{M}_{s_1,s_2}^{P,acc}}{k}, \\ \mathcal{M}_{s_1,s_2}^{(a_1P_1 + \ldots + a_mP_m),acc} &= \sum_{i=1}^m a_i \mathcal{M}_{s_1,s_2}^{P_i,acc}. \end{aligned}$$

We observe that analogous relations for the numbers $\mathcal{M}_{s_1,s_2}^{\cdot,rej}$ are valid.

Now we have to define what we mean by reductions. Fortunately, this is much easier here than in machine–based complexity theory.

Definition 2 *Let $F = (f_{2n} : \Sigma^n \times \Sigma^n \to \{0,1\})_{n \in \mathbb{N}}$ and $G = (g_{2n} : \Gamma^n \times \Gamma^n \to \{0,1\})_{n \in \mathbb{N}}$ be two decision problems. We say that F is rectangular reducible to G with respect to q, where $q : \mathbb{N} \to \mathbb{N}$ is a nondecreasing function, iff there are two transformations l_n, $r_n : \Sigma^n \to \Gamma^{q(n)}$ such that for all n and for all $\vec{x}, \vec{y} \in \Sigma^n$ we have $f_{2n}(\vec{x}, \vec{y}) = g_{2q(n)}(l_n(\vec{x}), r_n(\vec{y}))$. We write $F \leq_{rec}^q G$.*

Clearly, we can utilize rectangular reductions for proving lower bounds on communication complexity in the usual way.

One efficient way to get rectangular reductions is to work with projection reductions. In accordance with Skyum and Valiant (see [14]) we define.

Definition 3 *Let* $F = (f_{2n}(x_1, \ldots, x_{2n}))_{n \in \mathbb{N}}$ *and* $G = (g_{2n}(y_1, \ldots, y_{2n}))_{n \in \mathbb{N}}$ *be two sequences of functions given in distributed form* $f_{2n}, g_{2n} : \{0,1\}^{2n} \to \{0,1\}$. *We write* $F \leq^p_\Pi G$ *if there is a* $2p(n)$*–projection reduction*

$$\Pi = (\pi_{2n} : \{y_1, \ldots, y_{2p(n)}\} \to \{x_1, \ldots, x_{2n}, \neg x_1, \ldots, \neg x_{2n}, 0, 1\})_{n \in \mathbb{N}}$$

(see [14]) which respects the distribution of the variables,

$$\begin{aligned} \pi_{2n}(\{y_1, \ldots, y_{p(n)}\}) &\subseteq \{x_1, \ldots, x_n, \neg x_1, \ldots, \neg x_n, 0, 1\}, \\ \pi_{2n}(\{y_{p(n)+1}, \ldots, y_{2p(n)}\}) &\subseteq \{x_{n+1}, \ldots, x_{2n}, \neg x_{n+1}, \ldots, \neg x_{2n}, 0, 1\} \ . \end{aligned}$$

On balance of this section, let us motivate the modular communication complexity. In fact, we do more. We motivate all counting modes.

In the eighties, AC_0-circuits with MOD_m–gates adjoined came into prominence. Let us denote by $\mathrm{AC}_0^{[m]}$ the class of all sequences of Boolean functions computable by polynomially size bounded circuits of this type. We know, due to Razborov [10] and Smolensky [16], that $\mathrm{AC}_0^{[p]}$ and $\mathrm{AC}_0^{[q]}$, for p, q different prime numbers, are incomparable to each other with respect to inclusion. In contrast to that case, the classes $\mathrm{AC}_0^{[m]}$, m a composite number of more than one prime divisor, and the class $\mathrm{ACC} := \bigcup_{m=2}^{\infty} \mathrm{AC}_0^{[m]}$ are not well-understood yet. The only nontrivial bound for the whole class, an upper one, is due to Yao [20] : $\mathrm{ACC} \subseteq TC^*_{0,3}$ where $TC^*_{0,3}$ is the class of all Boolean functions computable by quasipolynomially bounded (i.e. $2^{(\log n)^{O(1)}}$) threshold circuits of depth 3.

The aim is to separate ACC from classes like NC_1. In order to make one step in this direction, in [4] depth-two circuits with arbitrary symmetric gates in the bottom level and MOD_m–gates at the top, called $(\mathrm{SYMM}, \mathrm{MOD}_m)$–circuits, are considered. An exponential lower bound for the sequence equality function is shown.

The following theorem supplies a general lower bound for (SYMM, μ)–circuits in terms of counting communication complexity. Thereby, a (SYMM, μ)–circuit is a depth two circuit having arbitrary symmetric gates in the bottom level and a μ–gate at the top, where $\mu : \mathbb{N}^2 \to \{0,1\}$. Recall, a μ–gate for the gate function $g(z_1, \ldots, z_{\sigma''})$ is defined by $\mu(\#\{j \mid z_j = 1\}, \#\{j \mid z_j = 0\})$.

Theorem 1 *Let* $f(x_1, \ldots, x_n, y_1, \ldots, y_n)$ *be a boolean function. Let* C_{2n} *be a* (SYMM, μ)*–circuit computing* f_{2n}. *Then*

$$\log_2(\mathrm{SIZE}(C_{2n})) = \Omega(\mu\text{-CC}(f_{2n})).$$

Proof. Let $\sigma := \mathrm{SIZE}(C_{2n})$, and let $(e_1, \ldots, e_{\sigma''})$ be the sequence of wires of the circuit C_{2n} leading to the top gate. Then we define the sequence $(g_1(z_{11}, \ldots, z_{1\sigma'_1}), \ldots, g_1(z_{\sigma''1}, \ldots, z_{\sigma''\sigma'_{\sigma''}}))$ to be the sequence of gate functions of the source gates of $(e_1, \ldots, e_{\sigma''})$.

Let $\pi_{2n}^{(i)} : \{z_{ij} \mid j = 1, \ldots, \sigma'_i\} \to \{x_i^e, y_i^e, 0, 1 \mid i = 1, \ldots, n\}$, for $i = 1, \ldots, \sigma''$, be projections resulting from the wires of C_{2n} which lead from the input nodes to the bottom gates. Then the functions $\tilde{g}_i(x_1, \ldots, x_n, y_1, \ldots, y_n)$ are defined to be $g_i \circ (\pi_{2n}^{(i)})^t$, for $i = 1, \ldots, \sigma''$ (see Definition 3).

We describe a protocol P of length $2\lceil \log_2 \sigma \rceil + 1$, where processor $\mathcal{P}_1$ has $(x_1, \ldots, x_n)$ as inputs and $\mathcal{P}_2$ the vector $(y_1, \ldots, y_n)$.

Round 1. $\mathcal{P}_1$ chooses nondeterministically a number $i \in \{1, \ldots, \sigma''\}$ and sheds it, encoded as a message of length $\lceil \log_2 \sigma \rceil$.

Round 2. $\mathcal{P}_2$ sheds the number $\sum_{e=0}^{1} \sum_{k=1}^{n} y_k^e \cdot |(\pi_{2n}^{(i)})^{-1}(y_k^e)|$, again encoded as a message of length $\lceil \log_2 \sigma \rceil$.

Round 3. Since the functions $\tilde{g}_i$ are symmetric and $\tilde{g}_i \leq_{\pi_{2n}^{(i)}} g_i$, processor $\mathcal{P}_1$ is able to compute $\tilde{g}_i(x_1, \ldots, x_n, y_1, \ldots, y_n)$ now. It sheds the result.

It is easy to see that the protocol P μ–computes the function f if and only if the top gate is a μ–gate. □

2 A Partial Solution of the "Log Rank" Conjecture for Modular Communication Complexity

Throughout this section, let f_{2n} denote a function $f_{2n} : \Sigma^n \times \Sigma^n \to \{0, 1\}$, where Σ is a finite alphabet. If $m = p_1^{l_1} \cdot \ldots \cdot p_r^{l_r}$ is the unique decomposition of the natural number m into powers of pairwise different prime numbers, the so–called primary decomposition, then $\rho(m)$, the so called *radical*, is defined to be the number $p_1 \cdot \ldots \cdot p_r$.

This section is aimed at proving Theorem 2 and Corollary 1. This theorem characterizes the modular communication complexity of a function sequence f_{2n} in terms of a rigidity function depending on the communication matrix $\mathcal{M}^{f_{2n}}$. To do so, we adopt and weaken in Definition 4 the concept of variation ranks of communication matrices developed in [4]. (In Section 4 and Section 3, it turns out, that this extends and simplifies the possibilities to calculate the modular communication complexity. This will justify the headline of this section.)

Our explanation is going on in two global steps. The first one is devoted to prove Proposition 1 saying that MOD_m–protocols and $\mathrm{MOD}_{\rho(m)}$–protocols are of equal power. In the second one we introduce our rigidity function "rigidity rank", justify it in Lemma 6, and formulate and prove Theorem 2.

Let us turn to the first global step. We prove first that $\mathrm{MOD}_{\rho(m)}$–protocols can be efficiently simulated by MOD_m–protocols.

Lemma 3 *If $m_1|m_2$, then* MOD_{m_2}-$\mathrm{CC}(f) \leq \mathrm{MOD}_{m_1}$-$\mathrm{CC}(f) + \lceil \log_2 \frac{m_2}{m_1} \rceil$ *for each function* f.

Proof. Let P be the MOD_{m_1}–protocol of length L for f. Define the protocol P' of length $L + \lceil \log_2 \frac{m_2}{m_1} \rceil$ to be $\left(\frac{m_2}{m_1} P\right)$. Then $\mathcal{M}^{P',acc}_{s_1,s_2} = \frac{m_2}{m_1} \cdot \mathcal{M}^{P,acc}_{s_1,s_2}$. Consequently,

$$\mathcal{M}^{P',acc}_{s_1,s_2} \equiv 0 \pmod{m_2} \iff \mathcal{M}^{P,acc}_{s_1,s_2} \equiv 0 \pmod{m_1}.$$

□

Now, we show that prime powers are not more powerful than primes.

Before formulating the next lemma, let us introduce the following notations for $a, r \in \mathbb{N}$.

$$\begin{aligned} \binom{a}{r}^{(0)} &:= a \\ \binom{a}{r}^{(i+1)} &:= \binom{\binom{a}{r}^{(i)}}{r} \end{aligned}$$

If P is a protocol of length L, then, of course, we can recursively define protocol $\binom{P}{r}^{(i)}$ of length $p^i L$ such that

$$\mathcal{M}^{\binom{P}{r}^{(i)},acc}_{s_1,s_2} = \binom{\mathcal{M}^{P,acc}_{s_1,s_2}}{r}^{(i)},$$

by Definition 1 and Lemma 2.

Lemma 4 *Let p be a prime number and $l \geq 2$ a natural number. Then*

$$\mathrm{MOD}_p\text{-}\mathrm{CC}(f_{2n}) = \Theta\left(\mathrm{MOD}_{p^l}\text{-}\mathrm{CC}(f_{2n})\right)$$

for all f_{2n}.

Proof. Again we write f instead of f_{2n}.

The lower bound for MOD_p-$\mathrm{CC}(f_{2n})$ follows directly from Lemma 3.

Let us turn to prove the upper bound. If $a \in \mathbb{N}$, then it is well–known that

$$a \equiv 0 \pmod{p^l} \iff \binom{a}{p} \equiv 0 \pmod{p^{l-1}} \text{ and } a \equiv 0 \pmod{p}.$$

Consequently, using the above notations,

$$\begin{aligned} & a \equiv 0 \pmod{p^l} \\ & \iff \binom{a}{p}^{(i)} \equiv 0 \pmod{p}, \text{ for all } i = 0, \ldots, l-1 \\ & \iff 1 + (p-1) \prod_{i=0}^{l-1} \left(1 + (p-1)\left(\binom{a}{p}^{(i)}\right)^{p-1}\right) \equiv 0 \pmod{p}. \end{aligned}$$

Let P be a MOD_{p^l}–protocol of length L for f. Let $\tilde{P}$ be the protocol of length 1, where processor $\mathcal{P}_1$ sheds 1. Define P' to be

$$\left(\tilde{P} + (p-1)\left(\tilde{P} + (p-1)\left(\binom{P}{p}^{(0)}\right)^{p-1}\right)\ldots\left(\tilde{P} + (p-1)\left(\binom{P}{p}^{(l-1)}\right)^{p-1}\right)\right).$$

Then

$$\mathcal{M}_{s_1,s_2}^{P',acc} \;=\; 1 + (p-1)\prod_{i=0}^{l-1}\left(1 + (p-1)\left(\binom{\mathcal{M}_{s_1,s_2}^{P,acc}}{p}^{(i)}\right)^{p-1}\right)$$

Thus

$$\mathcal{M}_{s_1,s_2}^{P,acc} \equiv 0 \pmod{p^l} \iff \mathcal{M}_{s_1,s_2}^{P',acc} \equiv 0 \pmod{p}.$$

The length of the protocol P' is as desired. □

The following lemma ensures that Lemma 4 can be applied to prove Proposition 1 in the following way. First, dissect the modulus m into its primary components by Lemma 5, Claim 1. Second, apply Lemma 4 to each of these components. Third, combine the protocols resulting from Lemma 4 by Lemma 5, Claim 2.

Lemma 5 *1. If $m = m_1 \cdot \ldots \cdot m_r$, where the numbers $m_i \in \mathbb{N}$ are pairwise coprime, then each f_{2n} has a representation $f_{2n} = f_{2n,1} \vee \ldots \vee f_{2n,r}$, such that*

$$\max\{\mathrm{MOD}_{m_i}\text{-CC}(f_{2n,i}) \mid i = 1,\ldots,r\} \leq \mathrm{MOD}_m\text{-CC}(f_{2n}).$$

2. Assume that f_{2n} is represented by $f_{2n} = f_{2n,1} \vee \ldots \vee f_{2n,r}$. Assume, moreover, that $m = m_1 \cdot \ldots \cdot m_r$, where the numbers $m_i \in \mathbb{N}$ are pairwise coprime. Then

$$\mathrm{MOD}_m\text{-CC}(f_{2n}) = O\left(\max\{\mathrm{MOD}_{m_i}\text{-CC}(f_{2n,i}) \mid i = 1,\ldots,r\}\right)$$

Proof. Throughout this proof, we write f instead of f_{2n} and f_i instead of $f_{2n,i}$.

Claim 1. Let P be the protocol that MOD_m–computes the function f. Define f_i to be the functions MOD_{m_i}–computed by the protocol P. Clearly, $f = f_1 \vee \ldots \vee f_r$.

Claim 2. Let P_i, for $i = 1,\ldots,r$, be MOD_{m_i}–protocols for f_i of length L_i. Let $e_1,\ldots,e_r \in \{1,\ldots,m-1\}$ be representatives of the orthogonal idempotents in $\mathbb{Z}/m_1\mathbb{Z} \times \ldots \times \mathbb{Z}/m_r\mathbb{Z}$, which are, of course, pairwise different. Remember, that for all $x, x_1,\ldots,x_r \in \mathbb{Z}$,

$$x_i \equiv x \pmod{m_i}, \text{ for } i = 1,\ldots,r \iff x \equiv \sum_{i=1}^{r} x_i e_i \pmod{m}$$

We consider the protocol $P := (e_1P_1 + \ldots + e_rP_r)$ of length

$$\left\lceil \log_2 \sum_{i=1}^{r} e_i \right\rceil + \max\{L_i \mid i = 1,\ldots,r\},$$

which is, of course, less than

$$\max\{L_i \mid i = 1, \ldots, r\} + \left\lceil \log_2(rm - \binom{r+2}{2}) \right\rceil .$$

It follows from Lemma 2 that, for all $(s_1, s_2) \in S_1 \times S_2$,

$$\begin{aligned} \mathcal{M}_{s_1,s_2}^{P_i,acc} &\equiv 0 \pmod{m_i} \text{ for all } i = 1, \ldots, k \\ \iff f_i(s_1, s_2) &= 0 \text{ for all } i = 1, \ldots, k \\ \iff \mathcal{M}_{s_1,s_2}^{P,acc} &\equiv 0 \pmod{m}. \end{aligned}$$

Consequently, the protocol P MOD_m–computes f. □

Proposition 1 follows now.

Proposition 1 *Let $m > 1$ be a natural number. Then*

$$\mathrm{MOD}_m\text{-CC}(f_{2n}) = \Theta\left(\mathrm{MOD}_{\rho(m)}\text{-CC}(f_{2n})\right)$$

Before defining our rigidity function, let us justify it.

Lemma 6 *Let $m = p_1 \cdot \ldots \cdot p_r$, where the p_i are pairwise different prime numbers. If P is a MOD_m–protocol of length L computing the function f, then there is a protocol P' of length $L \cdot \prod_{i=1}^{r}(p_i - 1)$ computing the same function which fulfills the following property: For all inputs $(s_1, s_2) \in f^{-1}(1)$, there is a nonempty index set $\emptyset \neq I \subseteq \{1, \ldots, r\}$ such that $\forall \nu \in I \left(\mathcal{M}_{s_1,s_2}^{P',acc} \equiv 1 \pmod{p_\nu}\right)$ and $\forall \nu \notin I \left(\mathcal{M}_{s_1,s_2}^{P',acc} \equiv 0 \pmod{p_\nu}\right)$.*

Proof. We consider the protocol $P' := P^{\prod_{i=1}^{r}(p_i-1)}$ of length $L \cdot \prod_{i=1}^{r}(p_i - 1)$. Then the claim immediately follows from Lemma 2 and Fermat's Little Theorem. □

Now we are prepared for our crucial definition.

Definition 4 *1. Let m be a product $p_1 \cdot \ldots \cdot p_r$, where the p_i are pairwise different prime numbers. Two $N \times N$–matrices A and B over the ring of integers are defined to be mod_m–equivalent, if and only if, for all entry indices i, j, and all prime number indices $k = 1, \ldots, r$,*

$$a_{ij} \equiv b_{ij} \pmod{p_k} \quad \text{or} \quad a_{ij} b_{ij} \equiv 0 \pmod{p_k}$$

and

$$a_{ij} \equiv 0 \pmod{m} \iff b_{ij} \equiv 0 \pmod{m}.$$

2. *Let A be an integer matrix, and let $m > 0$ be an arbitrary natural number. We define the rigidity rank* $\mathrm{rrk}_m(A)$ *to be the minimum of all numbers* $\mathrm{rk}_{\mathbb{Z}/\rho(m)\mathbb{Z}}(B \bmod \rho(m))$ *where B is an integer matrix which is* $\mathrm{mod}_{\rho(m)}$*–equivalent to A.*

Theorem 2 *Let $m > 1$ be a natural number. Then*

$$\mathrm{MOD}_m\text{-CC}(f_{2n}) = \Theta\left(\log_2\left(\mathrm{rrk}_m(M^{f_{2n}})\right)\right).$$

Proof. Referring to Proposition 1, we assume without loss of generality that $m = \rho(m)$.

The lower bound is obvious, if we choose the optimal MOD_m–protocol P' in such a way that the conditions of Lemma 6 are fulfilled.

The upper bound. We choose an integer matrix B which is mod_m–equivalent to $M^{f_{2n}}$, such that $r = \mathrm{rk}_{\mathbb{Z}/m\mathbb{Z}}(B \bmod m) = \mathrm{rrk}_m(M^{f_{2n}})$. Then $B = B^{(1)} + \ldots + B^{(r)}$, where the $B^{(k)}$ have $\mathbb{Z}/m\mathbb{Z}$–rank 1. This is equivalent to $B_{ij}^{(k)} \equiv U_i^{(k)} \cdot V_j^{(k)} \pmod m$, for $U_i^{(k)}, V_j^{(k)} \in \{1, \ldots, m\}$, and for $i, j = 1, \ldots N$.

Now we can describe the following protocol P. Assume that the input is $(i, j) \in S_1 \times S_2$.

First, processor $\mathcal{P}_1$ chooses nondeterministically some indices k, $1 \leq k \leq r$, and l_1, $1 \leq l_1 \leq U_i^{(k)}$, and sheds (k, l_1).

Second, processor $\mathcal{P}_2$ chooses nondeterministically some index l_2, $1 \leq l_2 \leq V_j^{(k)}$, and sheds $(l_2, 1)$.

Clearly, there are $\sum_{k=1}^{r} U_i^{(k)} \cdot V_j^{(k)} \equiv B_{ij} \pmod m$ many accepting computations assigned to the input (i, j). It follows that the μ-protocol P computes the function f. Obviously, the length of the protocol is bounded above by $\log_2 r + 2\log_2 m + 1$. □

In the case of m being a prime number, we can even do better.

Corollary 1 *If $m = p$ is a prime number, we have*

$$\mathrm{MOD}_p\text{-CC}(f_{2n}) = \Theta\left(\log_2\left(\mathrm{rk}_{\mathbb{Z}/p\mathbb{Z}}(M^{f_{2n}})\right)\right).$$

3 Application 1: Invertible Incidence Functions

A function $f : S_1 \times S_2 \to \{0, 1\}$, where $|S_1| = |S_2|$, is called an *invertible incidence function* if and only if its communication matrix can be transformed by interchanging rows and columns into a triangular form with the main diagonal elements all being 1. Clearly, the next lemma is crucial.

Lemma 7 *Let D_N denote an upper or lower triangular $N \times N$–matrix over $\{0, 1\}$ with 1's in the main diagonal, and let $m = p_1 \cdot \ldots \cdot p_r$, where the p_i are pairwise different prime numbers. Then* $\mathrm{rrk}_m(D_N) \geq \lceil N/r \rceil$.

Proof. Let $A = (a_{ij})$ be an integer matrix such that A is mod_m–equivalent to D_N and $\mathrm{rrk}_m(D_N) = \mathrm{rk}_{\mathbb{Z}/m\mathbb{Z}}(A \bmod m)$.

By definition we have, for all i, $a_{ii} \not\equiv 0 \pmod m$, and $a_{ij} \equiv 0 \pmod m$ for all $j < i$. Moreover, for all $k = 1, \ldots, r$, and all $i = 1, \ldots, N$, $a_{ii} \equiv 1 \pmod{p_k}$ or $a_{ii} \equiv 0 \pmod{p_k}$.

We conclude that there is an index k_0 and a set of indices $\mathcal{I} \subseteq \{1, 2, \ldots, N\}$, $\#\mathcal{I} \stackrel{def}{=} N' \geq \lceil N/r \rceil$, such that $a_{ii} \equiv 1 \pmod{p_{k_0}}$ for all $i \in \mathcal{I}$. After deleting all rows and columns of A whose indices do not belong to $\mathcal{I}$, we get an integer $N' \times N'$–matrix D' which is upper triangular and $\mathrm{mod} p_{k_0}$ with 1's on the main diagonal.

Obviously, we have
$\mathrm{rk}_{\mathbb{Z}/m\mathbb{Z}}(A \bmod m) \geq \mathrm{rk}_{\mathbb{Z}/p_{k_0}\mathbb{Z}}(A \bmod p_{k_0}) \geq \mathrm{rk}_{\mathbb{Z}/p_{k_0}\mathbb{Z}}(D' \bmod p_{k_0}) = N'$ □

The main result of this section, the next theorem, immediately follows from Lemma 7 and Theorem 2. The following corollary makes use of Theorem 1.

Theorem 3 *Let $m > 1$ be a natural number, and let f_{2n} be an invertible incidence function. Then*

$$\mathrm{MOD}_m\text{-CC}(f_{2n}) = \Theta(n).$$

Corollary 2 *Let C_{2n} be a $(\mathrm{SYMM}, \mathrm{MOD}_m)$–circuit computing an invertible incidence function f_{2n}. Then $\mathrm{SIZE}(C_{2n}) = 2^{\Omega(n)}$.*

The following functions are, for example, invertible incidence functions. (See Section 4, Remark 1 for more.)

- The *set disjointness test* $\mathrm{SDT} = (\mathrm{SDT}_{2n} : \{0,1\}^{2n} \to \{0,1\})_{n \in \mathbb{N}}$, defined by

$$\mathrm{SDT}_{2n}(x_1, \ldots x_n, y_1, \ldots, y_n) \stackrel{def}{=} 1 - \bigvee_{i=1}^{n} (x_i \wedge y_i)$$

  (see e.g. [6]),

- The *sequence equality function* $\mathrm{SEQ} = (\mathrm{SEQ}2n : \{0,1\}^{2n} \to \{0,1\})_{n \in \mathbb{N}}$, defined by

$$\mathrm{SEQ}_{2n}(x_1, \ldots, x_n, y_1, \ldots, y_n) = \bigwedge_{i=1}^{n} (1 - ((x_i + y_i) \bmod 2)),$$

- The *order function* $\mathrm{ORDER} = (\mathrm{ORDER}_{2n} : \{0,1\}^{2n} \to \{0,1\})_{n \in \mathbb{N}}$, defined by

$$\mathrm{ORDER}_{2n}(x_1, \ldots x_n, y_1, \ldots, y_n) = 1 \iff \sum_{i=1}^{n} (x_i - y_i) 2^{i-1} \geq 0.$$

Corollary 3 *For arbitrary m, we have that*

$$\mathrm{MOD}_m\text{-CC}(\mathrm{SEQ}_{2n}) = \mathrm{MOD}_m\text{-CC}(\mathrm{SDT}_{2n}) = \mathrm{MOD}_m\text{-CC}(\mathrm{ORDER}_{2n}) = \Theta(n).$$

4 Application 2: Undirected Graph Connectivity

The *graph connectivity problem for undirected graphs* $\mathrm{UCON} = (\mathrm{UCON}_{n(n-1)})_{n\in\mathbb{N}}$ in distributed form can be formulated as follows. Assume that we are given two not necessarily edge-disjoint undirected graphs $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ on a common n–set of vertices V, where both graphs are represented as Boolean vectors of length $\binom{n}{2}$. The question is whether or not the graph $G \stackrel{def}{=} G_1 \cup G_2 = (V, E_1 \cup E_2)$ is connected, i.e. each pair of vertices in G is connected. In [17], the major developments in understanding the complexity of the graph connectivity problem in several computational models are surveyed.

We pursue the aim to prove the following theorem in this section:

Theorem 4 *Let m be arbitrary. Then* $\mathrm{MOD}_m\text{-CC}(\mathrm{UCON}_{n(n-1)}) = \Theta(n)$.

But first we review some results and methods which are strongly related to ours. Hajnal, Maass, and Turan proved in [3] the following theorem:

Theorem A $\mathrm{CC}(\mathrm{UCON}_{n(n-1)}) = \Theta(n \log n)$.

Their method involves the use of the Möbius function μ for the lattice of partitions of an n–set. Lovasz and Saks extended in [6] and [7] this idea to a large class of problems, the so-called *meet problems for finite lattices*, which can be formulated as follows. Let S be a finite lattice, and let both processors $\mathcal{P}_1$ and $\mathcal{P}_2$ be given an element x and y, respectively. Then they have to decide whether $x \wedge y = 0$. More formally, $\mathrm{MEET}_S : S \times S \to \{0,1\}$ is defined by $\mathrm{MEET}_S(x, y) = \delta(0, x \wedge y)$.

Theorem B *Let* MEET_S *be the meet problem of a finite lattice. Let S have a atoms and b Möbius elements (i.e., elements x such that $\mu(0, x) \neq 0$). Then*

$$\log b \leq \mathrm{CC}(\mathrm{MEET}_S) \leq (\log a)(\log b).$$

Recently, Raz and Spieker [12] proved

Theorem C *If processor $\mathcal{P}_1$ as well as processor $\mathcal{P}_2$ have a bipartite perfect matching on $2n$ vertices with two colors of size n as an input, and if their goal is to determine whether the union of the two matchings forms a Hamiltonian cycle, the nondeterministic communication complexity of the problem is $\Omega(n \log \log n)$.*

Since the problem of Theorem C is a subproblem of UCON , it follows

Corollary D $\text{N-CC}(\mathrm{UCON}_{n(n-1)}) = \Omega(n \log \log n)$.

Thus the modular acception modes are better than ordinary nondeterminism for detecting undirected graph connectivity.

Let us turn to the proof of Theorem 4 now. We proceed as follows. We transform the Möbius function method for proving lower bounds on the length of deterministic protocols to the case of MOD_m–protocols. Thus we prove the upper bound of Proposition 3. We get the lower bound of Proposition 4 when we reduce the sequence equality function (see Section 3, Corollary 3) to undirected graph connectivity via a polynomial projection reduction.

We can only give a very brief treatment on Möbius functions. For more see, e.g., [13]. Let S be a finite partially ordered set. The *incidence algebra* $\mathcal{A}(S)$ is defined as follows: Consider the set of functions of two variables $f(x,y)$, for x and y ranging in S, having values in $\mathbb{R}$, the field of real numbers, and with the property that $f(x,y)=0$ whenever $x \not\leq y$. The sum and the multiplication by scalars are defined pointwise. The product of f and g is defined as follows.

$$(fg)(x,y) \stackrel{def}{=} \sum_{z \in S} f(x,z)g(z,y)$$

Clearly, Kronecker's δ–function is the 1 of $\mathcal{A}(S)$. The *zeta function* $\zeta(x,y) \in \mathcal{A}(S)$ is defined by $\zeta(x,y)=1$ if $x \leq y$ and $\zeta(x,y)=0$ otherwise.

It is easy to see that the zeta function ζ has in $\mathcal{A}(S)$ a multiplicative inverse, the so–called *Möbius function*, which has in fact values in $\mathbb{Z}$.

The *Möbius Inversion Formula* is well–known: Let $f(x)$ be an $\mathbb{R}$–valued function, for x ranging in the finite poset S, and let $g(x)=\sum_y f(y)\zeta(y,x)$. Then $f(x)=\sum_y g(y)\mu(y,x)$.

It is easy to see that $\mu(x,y)$ only depends on the the structure of the interval $[x,y]$, and not on the whole poset P. Moreover, we know that if μ^* is the Möbius function of the dual poset S^*, then $\mu^*(x,y)=\mu(y,x)$.

Let us motivate the notation "invertible incidence function" of Section 3 at this point.

Remark 1 *Let $f : S_1 \times S_2 \to \{0,1\}$ be an invertible incidence function in the sense of Section 3. Then, of course, there is a poset S, and there are bijections $\kappa_j : S_j \to S$, for $j=1,2$, such that $f \circ (\kappa_1 \times \kappa_2)$ is a unit in $\mathcal{A}(S)$.*

The other way round, let $f \in \mathcal{A}(S)$ be a $\mathbb{Z}$–valued unit, and let, moreover, f be naturally represented as a function from $S \times S$ to $\{0,1\}^k$, for some $k \in \mathbb{N}$, by encoding $f(S)$ as a set of strings in $\{0,1\}^k$. If we define μ-CC(f) to be $\max_{1 \leq \nu \leq k}(\mu\text{-CC}(f_\nu))$, where f_ν is the νth coordinate function of f, then it results from Theorem 3 that $\mathrm{MOD}_m\text{-CC}(f)=\Theta(|S|)$.

Let us assume from now on that the poset S is a lattice. Let M be a 0–1 matrix. Check whether there are two equal rows or colomns in M and if this is the case, then delete one of them. Do that as long as possible. The resulting matrix $\tilde{M}$ is called the *core* of M. Clearly, any communication complexity of two problems whose communication matrices have the same core is the same. Now it is not difficult to see that the core of $\mathcal{M}^{\mathrm{UCON}_{n(n-1)}}$ equals the core of $\mathcal{M}^{\mathrm{MEET}_{\mathcal{P}(n)^*}}$, where $\mathcal{P}(n)^*$ is the lattice dual to the lattice of partitions of an n–set. Thus, in order to prove the upper bound of Proposition 3, we can proceed as follows.

First, we show that the MOD_p–communication complexity of the meet problem for a finite lattice S is equal to the logarithm of a number which might be called the number of Möbius elements $\bmod p$ of the lattice S (see Proposition 2). Second, we compute this number in the case of $S = \mathcal{P}(n)^*$ (see Lemma 8). We are done with Lemma 3 then.

Proposition 2 *Let p be a prime number. Let MEET_S be the meet problem of a finite lattice S. Let S have c elements x such that $\mu(0,x) \not\equiv 0 \pmod p$. Then*

$$\mathrm{MOD}_p\text{-CC}(\mathrm{MEET}_S) = \Theta(\log c).$$

Proof. Let $\mathcal{M}$ be the communication matrix of the meet problem assigned to the finite lattice S. Let $\tilde{\mathcal{M}}$ be the diagonal matrix $\mathrm{diag}(\mu(0,x))_{x\in S}$, which has, as we have already stressed, coefficients in the ring $\mathbb{Z}$ of integers, and let $\zeta = (\zeta(x,y))_{x,y\in S}$ be the matrix associated with the zeta function. Wilf observed in [18], that $\zeta^T \cdot \tilde{\mathcal{M}} \cdot \zeta = \mathcal{M}$. The claim follows from the Möbius Inversion Formula and from Corollary 1. □

Lemma 8 *Let $\mathcal{P}(n)^*$ be the lattice dual to the lattice $\mathcal{P}(n)$ of partitions, let $p < n$ be a prime number, and let μ^* be the Möbius function of $\mathcal{P}(n)^*$. Then*

$$\#\{x \in \mathcal{P}(n)^* \mid \mu^*(0,x) \not\equiv 0 \pmod p\} \leq p^n.$$

Proof. The following three facts can be found in any standard book of combinatorics.

Fact 1. If $x \in \mathcal{P}(n)$, and if $b(x)$ is the number of blocks of the partition x, then $[x,1] \cong \mathcal{P}(b(x))$.

Fact 2. If μ is the Möbius function of $\mathcal{P}(n)$, then $\mu^*(0,1) = \mu(0,1) = (-1)^{n-1}(n-1)!$.

Fact 3. Let $S(n,k)$ denote the number of partitions of an n–set into exactly k blocks (Stirling numbers of the second kind), then $\sum_{k=0}^{n} S(n,k)[X]_k = X^n$, where X is an indeterminant and $[X]_k = X \cdot (X-1) \cdot \ldots \cdot (X-k+1)$ is the falling factorial.

We get the following sequence of equations.

$$\begin{aligned}
\#\{x \in \mathcal{P}(n)^* \mid \mu^*(0,x) \not\equiv 0 \pmod p\} &= \#\{x \in \mathcal{P}(n) \mid \mu(x,1) \not\equiv 0 \pmod p\} \\
\#\{x \in \mathcal{P}(n) \mid \mu(x,1) \not\equiv 0 \pmod p\} &= \sum_{k=0}^{p} S(n,k) \\
\sum_{k=0}^{p} S(n,k) &\leq \sum_{k=0}^{p} S(n,k)[p]_k = p^n,
\end{aligned}$$

where the first one is clear, the second one follows from Fact 1 and Fact 2, and the third one from Fact 3. □

Proposition 3 *Let m be arbitrary. Then $\mathrm{MOD}_m\text{-CC}(\mathrm{UCON}_{n(n-1)}) = O(n)$.*

Lemma 9 SEQ $= (\mathrm{SEQ}_{2n})_{n\in\mathbb{N}}$ *is reducible to* UCON $= (\mathrm{UCON}_{n(n-1)})_{n\in\mathbb{N}}$ *given in distributed form via an* $O(n^2)$*–projection reduction with respect to the partition of the variables.*

Proof. Consider an input $(t_1,\ldots,t_n,u_1,\ldots,u_n)$ of SEQ_{2n}. The projection reduction

$$\pi_{n(n-1)} : \{x_{ij}, y_{ij} \mid i,j = 1,\ldots,n,\ i<j\} \to \{0,1,t_\nu,u_\nu,\neg t_\nu,\neg u_\nu \mid \nu = 1,\ldots,n\},$$

where the values of the Boolean variables x_{ij} and y_{ij} define the graphs G_1 and G_2 accessible to the processors $\mathcal{P}_1$ and $\mathcal{P}_2$, is defined as shown in Figure 1 and Figure 2. We visualize this graph in such a way that the edges which are not constant are labelled with the corresponding literals. The meaning is that such an edge belongs to the graph if and only if the labelling literal is true. Clearly, this graph is connected if and only if

$$\mathrm{SEQ}_{2n}(t_1,\ldots,t_n,u_1,\ldots,u_n) = 1.$$

□

The lower bound easily results from Lemma 9, and Corollary 3:

Proposition 4 *Let* m *be arbitrary. Then* $\mathrm{MOD}_m\text{-CC}(\mathrm{UCON}_{n(n-1)}) = \Omega(n)$.

Corollary 4 *Let* C_{2n} *be a* $(\mathrm{SYMM},\mathrm{MOD}_m)$*–circuit computing* $(\mathrm{UCON}_{n(n-1)})_{n\in\mathbb{N}}$. *Then* $\mathrm{SIZE}(C_{2n}) = 2^{\Omega(n)}$.

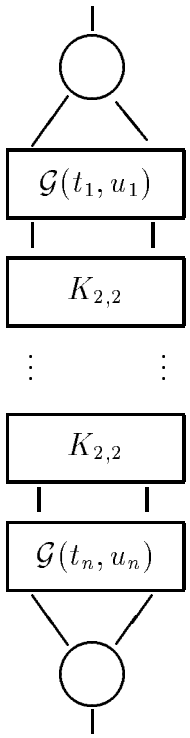

Figure 1. *The graph shows the construction of the projection of* SEQ_{2n}. *(*$K_{2,2}$ *denotes the full bipartite graph having* 2×2 *nodes,* $\mathcal{G}(t_\mu,u_\mu)$ *is defined in Figure 2.)*

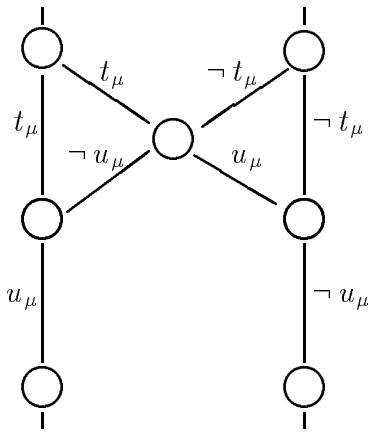

Figure 2. *The graphs* $\mathcal{G}(t_\mu,u_\mu)$ *of Figure 1.*

References

[1] A. Borodin, A. Razborov, R. Smolensky, *On Lower Bounds for Read–k–times Branching Programs*, Computational Complexity **3**(1993), pp. 1–18.

[2] C. Damm, M. Krause, Ch. Meinel, St. Waack, *Separating Counting Communication Complexity Classes*, in: Proc. 9th STACS, Lecture Notes in Computer Science **577**, Springer Verlag 1992, pp. 281–293.

[3] A. Hajnal, W. Maass, G. Turan, *On the Communication Complexity of Graph Problems*, in: Proc. 20th ACM STOC 1988, pp. 186–191.

[4] M. Krause, St. Waack, *Variation ranks of communication matrices and lower bounds for depth two circuits having symmetric gates with unbounded fan–in*, in: Proc. 32nd IEEE FOCS 1991, pp. 777–782.

[5] S. Lokam, *Spectral methods for matrix rigidity with applications to size–depth tradeoffs and communication complexity*, in: Proc. 36th IEEE FOCS 1995.

[6] L. Lovasz, *Communication complexity: A survey*, in: *Paths, flows and VLSI–layouts*, Springer–Verlag 1990, pp. 235–266.

[7] L. Lovasz, M. Saks, *Communication complexity and combinatorial lattice theory*, Journal of Computer and System Sciences **47**(1993), pp. 322-349.

[8] K. Mehlhorn, E. M. Schmidt, *Las Vegas is better than determinism in VLSI and distributed computing*, in: Proc. 14th ACM STOC 1982, pp. 330–337.

[9] P. Pudlak, *Large Communication in Constant Depth Circuits*, Combinatorica **14**(2)(1994), pp. 203–216.

[10] A.A. Razborov, *Lower bounds for the size of circuits of bounded depth with basis* $\{\wedge, \oplus\}$, Journ. Math. Zametki **41**(1987), pp. 598–607.

[11] A.A. Razborov, *On rigid matrices*, Manuscript.

[12] R. Raz, B. Spieker, *On the "log rank"– conjecture in communication complexity*, in: Proc. 34th IEEE FOCS 1993, pp. 168–176.

[13] G.-C. Rota, *On the foundation of combinatorial theory: I. Theory of Möbius functions*, Z. Wahrscheinlichkeitstheorie **2**(1964), pp. 340–368.

[14] L. Skyum, L. V. Valiant, *A complexity theory based on Boolean algebra*, in: Proc. 22th IEEE FOCS, pp. 244-253.

[15] V. Shoup, R. Smolensky, *Lower Bounds for Polynomial Evaluation and Interpolation*, in: Proc. 32nd IEEE FOCS (1991), pp. 378–383.

[16] R. Smolensky, *Algebraic methods in the theory of lower bounds for Boolean circuit complexity*, in: Proc. 19th ACM STOC (1987), pp. 77–82.

[17] A. Wigderson, *The complexity of graph connectivity*, TR 92-19, Leibnitz Center for Research in Computer Science, Institute of Computer Science, Hebrew University, Jerusalem.

[18] S. Wilf, *Hadamard determinants, Möbius functions and the cromatic number of graphs*, Bull./Amer. Math. Soc. **74**(1968), pp. 960–964.

[19] A. C.-C. Yao, *Some Complexity Questions Related to Distributed Computing*, in: Proc. 11st ACM STOC 1979, pp. 209–213.

[20] A. C.-C. Yao, *On* ACC *and threshold circuits*, in: Proc. 31st IEEE FOCS 1990, pp. 619–627.