ELSEVIER

Theoretical Computer Science 168 (1996) 39–51

Theoretical Computer Science

A comparison of two lower-bound methods for communication complexity

Martin Dietzfelbinger [a,*,1], Juraj Hromkovič [b,1,2], Georg Schnitger [c,3]

[a] *Fachbereich Informatik, Lehrstuhl II, Universität Dortmund, 44221 Dortmund, Germany*
[b] *Institut für Informatik und Praktische Mathematik, Universität zu Kiel, 24098 Kiel, Germany*
[c] *Fachbereich Informatik, Johann Wolfgang Goethe-Universität, 60325 Frankfurt a.M., Germany*

Abstract

The methods "Rank" and "Fooling Set" for proving lower bounds on the deterministic communication complexity of Boolean functions are compared. The main results are as follows.

(i) For almost all Boolean functions of $2n$ variables the Rank method provides the lower bound n on communication complexity, whereas the Fooling Set method provides only the lower bound $d(n) \leqslant \log_2 n + \log_2 10$. A specific sequence $\{f_{2n}\}_{n=1}^{\infty}$ of Boolean functions, where f_{2n} has $2n$ variables, is constructed such that the Rank method provides exponentially higher lower bounds for f_{2n} than the Fooling Set method.

(ii) A specific sequence $\{h_{2n}\}_{n=1}^{\infty}$ of Boolean functions is constructed such that the Fooling Set method provides a lower bound of n for h_{2n}, whereas the Rank method provides only $(\log_2 3)/2 \cdot n \approx 0.79 \cdot n$ as a lower bound.

(iii) It is proved that lower bounds obtained by the Fooling Set method are better by at most a factor of two compared with lower bounds obtained by the Rank method.

These three results together solve the last problem about the comparison of lower bound methods on communication complexity left open in Aho et al. (1983).

Finally, it is shown that an extension of the Fooling Set method provides lower bounds that are tight (up to a polynomial) for all Boolean functions.

1. Introduction and definitions

Communication complexity of two-party protocols, as introduced in [1, 15], is one of the most investigated complexity measures (see, for instance, surveys by Lovász [11] or Lengauer [9]), because it is closely related to fundamental complexity measures of several basic parallel and sequential computational models (e.g., Boolean circuits,

* Corresponding author.

[1] Supported in part by DFG Grant Di 412/2-1.

[2] Supported in part by SAV Grant No. 2/1138/94 of the Computer Science Institute of the Slovac Academy of Sciences.

[3] Supported in part by NSF Grant CCR-9114545.

PII S0304-3975(96)00062-X

VLSI circuits, branching programs, Turing machines, etc.). Here, we consider the standard model of deterministic two-party protocols computing a Boolean function f of $2n$ variables $x_1, x_2, \ldots, x_{2n}$ as follows. The computing model consists of two computers. At the beginning the "first" computer obtains the actual values $\alpha_1, \alpha_2, \ldots, \alpha_n$ of the variables $x_1, x_2, \ldots, x_n$, and the "second" computer obtains the values $\alpha_{n+1}, \ldots, \alpha_{2n}$ of the variables $x_{n+1}, \ldots, x_{2n}$. To compute the value $f(\alpha_1, \alpha_2, \ldots, \alpha_{2n})$ the computers may exchange several binary messages. The number of bits exchanged is the communication complexity of the two-party protocol on the input $\alpha_1, \alpha_2, \ldots, \alpha_{2n}$. The communication complexity of the two-party protocol is the maximum over all $\alpha \in \{0,1\}^{2n}$. The communication complexity $\mathrm{cc}(f)$ of f is the minimum over the communication complexities of all protocols computing f (for a formal definition see [1, 15]).

The communication complexity $\mathrm{cc}(f)$ of a Boolean function f is mainly used as a method for proving lower bounds on complexity measures concerning the computational models mentioned above. Thus, the main effort in the study of communication complexity is devoted to the development of methods for proving lower bounds on $\mathrm{cc}(f)$ for concrete functions f. The three basic lower-bound proof methods used are "Tiling" [16], "Rank" [12], and "Fooling Set" [2]. Let $\mathrm{t}(f), \mathrm{r}(f)$, and $\mathrm{fs}(f)$ denote the lower bounds provided by the Tiling, Rank, and Fooling Set method, respectively. Aho et al. [2] first dealt with a comparison of $\mathrm{cc}(f)$ and the lower bounds provided by the methods "Tiling", "Rank", and "Fooling Set". They showed the following.

(i) The tiling method always provides the highest lower bounds because
- for every $f, \mathrm{cc}(f)$ and the lower bound on $\mathrm{cc}(f)$ provided by the tiling method are polynomially related; namely $\mathrm{t}(f) - 1 \leqslant \mathrm{cc}(f) \leqslant (\mathrm{t}(f) + 1)^2$, and
- $\mathrm{r}(f) \leqslant \mathrm{t}(f)$ and $\mathrm{fs}(f) \leqslant \mathrm{t}(f)$ for every Boolean function f.

(ii) For any sufficiently large n, there exists a Boolean function f_{2n} of $2n$ variables such that $\mathrm{cc}(f_{2n}) = n$ and $\mathrm{fs}(f_{2n}) = \mathrm{O}(\log_2 n)$, i.e., in some cases the Fooling Set method can be very weak.

Two main problems left open in [2, 11] are the following:

(1) Does there exist a sequence of Boolean functions $\{h_{2n}\}_{n=1}^{\infty}$ such that the gap between $\mathrm{cc}(h_{2n})$ and $\mathrm{r}(h_{2n})$ is exponential? (The existence of such a sequence of functions was shown in [2] for a much weaker version of the Rank method than the general version considered in this paper.)

(2) What is the relation between the methods "Rank" and "Fooling Set"?

The aim of this paper is to deal with the second open problem and to consider extensions of the Fooling Set method. This is an important task since one usually applies the Rank method and the Fooling Set method to obtain lower bounds on communication complexity, whereas the Tiling method, which is the best one theoretically, is used very rarely. The reason is that lower bounds for $\mathrm{r}(f)$ and $\mathrm{fs}(f)$ usually are easier to obtain than for $\mathrm{t}(f)$. More precisely, a lower bound for $\mathrm{r}(f)$ for a function f is obtained by computing the rank (or a lower bound on the rank) of a given matrix, and a lower bound for $\mathrm{fs}(f)$ is obtained by constructing a set of inputs with some special properties. On the other hand, the tiling method requires solving a nontrivial optimization problem (a minimal cover of the 1's of a large matrix by disjoint

1-monochromatic submatrices). The extension of the Fooling Set method considered here is also based on constructing (or searching for the existence of) a set of inputs with some special properties. We prove that the extended Fooling Set method provides lower bounds polynomially close to $cc(f)$. This is the first "constructive" lower bound method (searching for an object with some given properties) that guarantees such close lower bounds for deterministic communication complexity. It is conjectured that the Rank method shares this property. Recently, it has been shown in [14] that the rank lower bound $r(f)$ may differ from $cc(f)$ by a nonconstant factor. Still, the conjecture remains open.

The paper is organized as follows. In the next two subsections we present our results. Section 2 provides the proofs required for a comparison of the Fooling Set and Rank methods, and in Section 3 we prove that the extended Fooling Set method provides lower bounds polynomially close to $cc(f)$. In the conclusion section we discuss some remaining open problems.

1.1. Fooling Set versus Rank

Let $B^m = \{f \mid f: \{0,1\}^m \to \{0,1\}\}$, the set of m-ary Boolean functions. For $f \in B^{2n}$, let $M(f) = [a_{i,j}]_{i,j=1,\ldots,2^n}$ denote the *communication matrix* of f, where $a_{i,j} = f(\alpha_i, \alpha_j) \in \{0,1\}$, and α_k is the kth word in $\{0,1\}^n$ in lexicographic order, for $k \in \{1,\ldots,2^n\}$.

Definition 1.1. Let f be a Boolean function. For an arbitrary field F with identity elements 0 and 1, let $\mathrm{Rank}_F(f)$ denote the rank of the matrix $M(f)$ over F. We define

$$\underline{\mathrm{Rank}(f)} = \max\{\mathrm{Rank}_F(f) \mid F \text{ is a field with identity elements 0 and 1}\}$$

and

$$\underline{r(f)} = \lceil \log_2(\mathrm{Rank}(f)) \rceil.$$

Note that $r(f) \leqslant cc(f)$ for every f and $\mathrm{Rank}(f) = \mathrm{Rank}_{\mathbb{Q}}(f)$ [12].

Definition 1.2. Let f be a Boolean function of $2n$ variables. For $\delta \in \{0,1\}$, a set

$$\underline{\mathscr{A}(f)} = \{(\alpha_1,\beta_1),(\alpha_2,\beta_2),\ldots,(\alpha_k,\beta_k)\}\ ,\ \alpha_i,\beta_i \in \{0,1\}^n \text{ for } i = 1,\ldots,k\,,$$

is called a δ-fooling set for f if

(i) $f(\alpha_i,\beta_i) = \delta$ for all $i \in \{1,\ldots,k\}$, and

(ii) $i \neq j, i,j \in \{1,\ldots,k\}$ implies that $f(\alpha_i,\beta_j) \neq \delta$ or $f(\alpha_j,\beta_i) \neq \delta$.

We define

$$\underline{\mathrm{Fool}_1(f)} = \max\{\mathrm{card}(\mathscr{A}(f)) \mid \mathscr{A} \text{ is a } \delta\text{-fooling set for } f \text{ and } \delta \in \{0,1\}\}$$

and

$$\underline{\mathrm{fs}(f)} = \lceil \log_2(\mathrm{Fool}_1(f)) \rceil.$$

Note that $\mathrm{fs}(f) \leqslant \mathrm{cc}(f)$ for every f [2]. Note also that the above definition of fooling sets [2] differs from the definition used in [9], where a weaker version is considered.

First, using counting arguments, we show that for random functions the Rank method is exponentially better than the Fooling Set method.

Theorem 1.3. (i) *If $n \in \mathbb{N}$ is sufficiently large then for at least a fraction of $\frac{1}{4}$ of the Boolean functions f of $2n$ variables the following holds:*

- $\mathrm{Fool}_1(f) \leqslant 10n$ (*i.e.*, $\mathrm{fs}(f) \leqslant \log_2 n + \log_2 10$), *and*
- $\mathrm{Rank}_{\mathbb{Z}_2}(f) = 2^n$ (*i.e.*, $\mathrm{r}(f) = \mathrm{cc}(f) = n$).

(ii) *Almost all Boolean functions f of $2n$ variables satisfy* $\mathrm{Rank}(f) = 2^n$ *and* $\mathrm{Fool}_1(f) \leqslant 10n$ (*i.e.*, $\mathrm{fs}(f) \leqslant \log_2 n + \log_2 10$ *and* $\mathrm{r}(f) = n$).

Part (ii) of Theorem 1.3 shows that the Rank method is exponentially better than the Fooling Set method for almost all functions; part (i) shows that this is true for a substantial number of functions even if only rank over $\mathbb{Z}_2$ is used.

Our next result shows that the Fooling Set method cannot be much better than the Rank method.

Theorem 1.4. *For all Boolean functions f and all fields F,*

$$\mathrm{Fool}_1(f) \leqslant (\mathrm{Rank}_F(f) + 1)^2 \quad (\textit{i.e.}, \mathrm{fs}(f) \leqslant 2\mathrm{r}(f) + 2).$$

Furthermore, we consider the function $g_{2n}(x_1, \ldots, x_n, y_1, \ldots, y_n) = \sum_{i=1}^{n} x_i y_i \bmod 2$, the inner product over $\mathbb{Z}_2$. The family $\{g_{2n}\}$ provides a specific example for which the Rank method is exponentially better than the Fooling Set method.

Theorem 1.5. *For every $n \in \mathbb{N}$ we have*

(i) $\mathrm{Rank}(g_{2n}) = 2^n - 1$, *and*

(ii) $\mathrm{Fool}_1(g_{2n}) \leqslant (n+1)^2$.

Thus, $\mathrm{fs}(g_{2n}) \leqslant 2\log(n+1)$ *and* $\mathrm{r}(g_{2n}) = n$.

Finally, we show that there is a function for which the Fooling Set method is better than the Rank method.

Theorem 1.6. *There is an algorithm that, for any $n = 4^m$, $m \in \mathbb{N}$, constructs a Boolean function h_{2n} of $2n$ variables such that*

(i) $\mathrm{Fool}_1(h_{2n}) = 2^n$, *and*

(ii) $\mathrm{Rank}(h_{2n}) = 3^{n/2}$.

Thus, $0.79\ldots \cdot n = \frac{1}{2}\log_2 3 \cdot n = \mathrm{r}(h_{2n}) < \mathrm{fs}(h_{2n}) = n = \mathrm{cc}(h_{2n})$.

1.2. An extended Fooling Set method

The Fooling Set method is based on the fact that any two different elements of a fooling set require different communications, since in the communication matrix $M(f)$ they span a 2×2 submatrix which is not monochromatic. This can easily be generalized

(as noted already in [13]) to "fooling sets of order k" defined by the requirement that any $k+1$ elements of such a set define a non-monochromatic submatrix of $M(f)$. We now give the formal definition of such generalized fooling sets.

Definition 1.7. Let $f \in B^{2n}$, $k \in \mathbb{N}$, and $\delta \in \{0,1\}$. A set $\mathscr{A} \subseteq \{0,1\}^{2n}$ is a *δ-fooling set of order k* if

(i) $f(w) = \delta$ for all $w \in \mathscr{A}$,

(ii) for any selection of $k+1$ elements from $\mathscr{A}$ the submatrix of $M(f)$ that is spanned by $\mathscr{A}$ is not monochromatic.

Obviously, δ-fooling sets of order 1 coincide with the conventional δ-fooling sets. The extended Fooling Set method works as follows.

Definition 1.8. Let $\delta \in \{0,1\}$. We define

(i) $\underline{\text{Fool}_k^{\delta}(f)} = \max\{\text{card}(\mathscr{A}) | \mathscr{A}$ is a δ-fooling set of order $k\}$, for $k \in \mathbb{N}$;

(ii) $\underline{\text{Fool}^{\delta}(f)} = \max\{\text{Fool}_k^{\delta}(f)/k | k \in N\}$;

(iii) $\underline{\text{Fool}(f)} = \max\{\text{Fool}^0(f), \text{Fool}^1(f)\}$.

Let $\text{ncc}(f)$ denote the nondeterministic communication complexity of a Boolean function f. We will show that the extended Fooling Set method provides a tight lower bound for $\text{ncc}(f)$. As a consequence, the extended Fooling Set method also yields tight lower bounds (up to a square) for the deterministic communication complexity.

Theorem 1.9. *For any $n \in \mathbb{N}$ and $f \in B^{2n}$ we have:*

(i) $\log_2(\text{Fool}^1(f)) \leqslant \text{ncc}(f) \leqslant \log_2(\text{Fool}^1(f)) + \log_2(3.6n)$;

(ii) $\log_2(\text{Fool}(f)) \leqslant \text{cc}(f) \leqslant \left(\log_2(\text{Fool}(f)) + \log_2(3.6n) + 1\right)^2$.

2. A comparison of the Fooling Set method and the Rank method

We start with the proof of Theorem 1.3, which is a combination of counting arguments.

Fact 2.1. *If $n \in \mathbb{N}$ is sufficiently large, then $\text{Rank}_{\mathbb{Z}_2}(f_{2n}) = 2^n$ for at least $\frac{9}{32} \cdot \text{card}(B^{2n})$ functions in B^{2n}.*

Proof. This follows from the well-known fact that the probability for m randomly chosen vectors from $\{0,1\}^m$ to be linearly independent over $\mathbb{Z}_2$ is exactly $\prod_{1 \leqslant i \leqslant m}(1-2^{-i})$ (see, e.g., [3, p. 169]). Using the inequality $(1-\delta_1)(1-\delta_2) \geqslant 1-(\delta_1+\delta_2)$, which is valid for $0 \leqslant \delta_1, \delta_2 \leqslant 1$, one easily sees by induction that the term $\prod_{1 \leqslant i \leqslant m}(1-2^{-i})$ can be bounded below by $(1-2^{-1})(1-2^{-2})(1-\sum_{3 \leqslant i \leqslant m} 2^{-i})$, hence by

$$\frac{1}{2} \cdot \frac{3}{4} \cdot \left(1 - \sum_{i \geqslant 3} 2^{-i}\right) = \frac{3}{8} \cdot \frac{3}{4} = \frac{9}{32}. \qquad \square$$

Fact 2.2.

$$\lim_{n\to\infty} \operatorname{card}(\{f_{2n} \in B^{2n} \mid \operatorname{Rank}_{\mathbb{Q}}(f_{2n}) = 2^n\})/\operatorname{card}(B^{2n}) = 1.$$

Proof. Komlós [7, 8] has proved that a random 0–1 $m \times m$-matrix has rank m over $\mathbb{Q}$ with probability tending to 1 for $m \to \infty$. □

This means that for almost all Boolean functions of $2n$ variables the Rank method provides the optimal lower bound n. Next, we show that most Boolean functions have small fooling sets. We do so by investigating the communication matrix $M(f)$ as a representation of f.

Definition 2.3. Let $\delta \in \{0,1\}$, $d \in \mathbb{N} - \{0\}$. A 0-1 $d \times d$-matrix $[m_{ij}]_{i,j=1,\ldots,d}$ is called a *δ-fooling matrix* if

(i) $m_{ii} = \delta$ for $i = 1,\ldots,d$, and

(ii) for all $r,s \in \{1,\ldots,d\}, r \neq s$, we have $m_{rs} \neq \delta$ or $m_{sr} \neq \delta$.

Any matrix M' obtained from a δ-fooling matrix M by any permutation of rows and columns of M is called a *δ-quasifooling matrix.*

Observation 2.4. *Let $M(f)$ be the communication matrix for $f \in B^{2n}$ and $\delta \in \{0,1\}$. Each δ-fooling set $\mathscr{A}$ for f unambiguously defines a* $\operatorname{card}(\mathscr{A}) \times \operatorname{card}(\mathscr{A})$ *δ-quasifooling submatrix of $M(f)$.*

Proof. Let $\mathscr{A} = \{(\alpha_i, \beta_i) \mid i = 1,\ldots,k\}$. Then the intersection of the rows corresponding to $\alpha_1,\ldots,\alpha_k$ and the columns corresponding to $\beta_1,\ldots,\beta_k$ is a $k \times k$ δ-quasifooling matrix. □

Now let us study how large quasifooling submatrices are for random 0–1 matrices.

Lemma 2.5. *Let* $\operatorname{Mf}(N,k)$ *be the number of all $N \times N$ Boolean matrices having a δ-quasifooling submatrix of size at least $k \times k$ for some $\delta \in \{0,1\}$. Then*

$$\operatorname{Mf}(N,k) \leqslant 2 \cdot \binom{N}{k}^2 \cdot k! \cdot 3^{\binom{k}{2}} \cdot 2^{N^2-k^2}.$$

Proof. There are 2 choices for $\delta \in \{0,1\}$, and $\binom{N}{k}^2$ ways to choose a placement of the $k \times k$ δ-quasifooling submatrix M' ($\binom{N}{k}$ ways to choose k rows (columns) from N rows (columns)). There are k elements of M' that have fixed value δ whose positions in M' can be chosen in $k!$ different ways. If we permute the rows of M' to get a δ-fooling matrix M, we see that there are only three possibilities for assigning values to any pair of symmetric elements of M (namely $(\bar{\delta},\bar{\delta})$, $(\bar{\delta},\delta)$, or $(\delta,\bar{\delta})$). Thus, there are $3^{\binom{k}{2}}$ possibilities for choosing the values for the elements in M'. All other elements lying outside M' may be chosen arbitrarily, providing $2^{N^2-k^2}$ possibilities. □

Lemma 2.6. *Let* $k \geqslant \lceil 10 \log_2 N \rceil$. *Then*

$$\mathrm{Mf}(N,k)/2^{N^2} = N^{-\Omega(\log N)}.$$

Proof. It is sufficient to show that $2 \cdot \binom{N}{k}^2 \cdot (k!) \cdot 3^{\binom{k}{2}} \cdot 2^{-k^2} = N^{-\Omega(\log N)}$ for $k \geqslant 10 \log_2 N$ (see Lemma 2.5). Let us bound this expression in the following way:

$$2 \cdot \binom{N}{k}^2 \cdot (k!) \cdot 3^{\binom{k}{2}} \cdot 2^{-k^2} \leqslant 2N^{2k} \cdot 3^{k^2/2} \cdot 2^{-k^2} = 2^{1+2k \log_2 N + (k^2/2) \cdot \log_2 3 - k^2}$$
$$= 2^{k^2(1/k^2 + (2 \log_2 N)/k + (\log_2 3)/2 - 1)}.$$

Since $(\log_2 3)/2 < \frac{4}{5}$ and $(2 \log_2 N)/k \leqslant \frac{1}{5}$, the claim follows. □

Now we are ready to complete the proof of Theorem 1.3.

Proof of Theorem 1.3. Following Fact 2.1, at least $\frac{9}{32} \cdot 2^{2^{2n}} \geqslant 0.26 \cdot 2^{2^{2n}}$ Boolean functions f from B^{2n} have $\mathrm{Rank}_{\mathbb{Z}_2}(f) = 2^n$. Following Lemma 2.6 with $N = 2^n$, for all sufficiently large n the number of functions $h \in B^{2n}$ with a fooling set of cardinality at least $10n$ (a δ-quasifooling submatrix of size $10n \times 10n$) is bounded by $\frac{1}{100} \cdot 2^{2^{2n}} = \mathrm{card}(B^{2n})/100$. For all such n there are $\mathrm{card}(B^{2n})/4$ functions f from B^{2n} with $\mathrm{Rank}(f) = 2^n$ and $\mathrm{Fool}_1(f) \leqslant 10n$. This proves assertion (i) of Theorem 1.3. Assertion (ii) follows from Fact 2.2 and Lemma 2.6. □

For the following, we need the notion of the Kronecker product of two matrices.

Definition 2.7. For arbitrary finite index sets $I, J, K, L \neq \emptyset$ and matrices $A = (\alpha_{i,j})_{i \in I, j \in J} \in F^{I \times J}$, $B = (\beta_{k,l})_{k \in K, l \in L} \in F^{K \times L}$ over some field F the *Kronecker product* $A \otimes B$ is defined as the matrix $C = (\gamma_{(i,k),(j,l)})_{(i,k,j,l) \in I \times K \times J \times L}$, where $\gamma_{(i,k),(j,l)} = \alpha_{i,j} \beta_{k,l}$.

Informally speaking, C is obtained by replacing the entry α_{ij} in A by the submatrix $\alpha_{ij} \cdot B$. The following property of the Kronecker product is well known.

Fact 2.8 (Kronecker fact). *For arbitrary matrices A and B over some field F as in Definition 2.7 we have*

$$\mathrm{Rank}_F(A \otimes B) = \mathrm{Rank}_F(A) \cdot \mathrm{Rank}_F(B).$$

To prove Theorem 1.4, we construct a function $f^* : \{0,1\}^{4n} \to \{0,1\}$ for every Boolean function $f : \{0,1\}^{2n} \to \{0,1\}$ as follows: $f^*(x_1x_2, y_1y_2) = f(x_1, y_1) \cdot f(y_2, x_2)$, for $x_1, y_1, x_2, y_2 \in \{0,1\}^n$. Define the function $f^R : \{0,1\}^{2n} \to \{0,1\}$ by setting $f^R(u,v) = f(v,u)$, where $u, v \in \{0,1\}^n$. Then, obviously, $M(f^*) = M(f) \otimes M(f^R)$.

Lemma 2.9. *Let* $\mathscr{A}$ *be a* 1*-fooling set for* f. *Then, over any field* F,

$$\mathrm{Rank}_F(f^*) \geqslant \mathrm{card}(\mathscr{A}).$$

Proof. Assume that $\mathscr{A} = \{(x_i, y_i) \mid 1 \leqslant i \leqslant r\}$. Set $X = \{x_i y_i \mid 1 \leqslant i \leqslant r\}$ and $Y = \{y_i x_i \mid 1 \leqslant i \leqslant r\}$. We claim that the submatrix of $M(f^*)$ obtained by the intersection of the row set X and the column set Y is a diagonal matrix. For this, observe that

$$a_{ij} = M(f^*)[x_i y_i, y_j x_j] = f^*(x_i y_i, y_j x_j) = f(x_i, y_j) \cdot f(x_j, y_i) \quad \text{for } 1 \leqslant i, j \leqslant r.$$

If $i = j$ then $a_{ij} = 1$ because $\mathscr{A}$ is a 1-fooling set. If $i \neq j$, then the fact that $\mathscr{A}$ is a 1-fooling set implies $f(x_i, y_j) = 0$ or $f(x_j, y_i) = 0$. Thus $a_{ij} = 0$. □

Lemma 2.10. *For every Boolean function f and any field F,*

$$\mathrm{Rank}_F(f)^2 = \mathrm{Rank}_F(f^*).$$

Proof. This follows from the Kronecker Fact 2.8, since $M(f^*) = M(f) \otimes M(f^R)$. □

Now, we are prepared to prove Theorem 1.4.

Proof of Theorem 1.4. Let $\mathscr{A}$ be a fooling set for f such that $\mathrm{card}(\mathscr{A}) = \mathrm{Fool}_1(f)$. We distinguish two cases:

(i) $\mathscr{A}$ is a 1-fooling set. Then, by Lemmas 2.9 and 2.10,

$$\mathrm{card}(\mathscr{A}) \leqslant \mathrm{Rank}_F(f^*) \leqslant (\mathrm{Rank}_F(f))^2 \quad \text{for any field } F.$$

(ii) $\mathscr{A}$ is a 0-fooling set. Then $\mathscr{A}$ is a 1-fooling set for $g = f \oplus 1$. Thus, as in (i), $\mathrm{card}(\mathscr{A}) \leqslant \mathrm{Rank}_F(g^*) \leqslant (\mathrm{Rank}_F(g))^2 \leqslant (\mathrm{Rank}_F(f) + 1)^2$, for any field F. □

Proof of Theorem 1.5. Recall the definition of the inner product function

$$g_{2n}(x_1, \ldots, x_n, y_1, \ldots, y_n) = \sum_{i=1}^{n} x_i y_i \bmod 2.$$

According to Theorem 1.4 we have $\mathrm{Fool}_1(g_{2n}) \leqslant (\mathrm{Rank}_{\mathbb{Z}_2}(g_{2n}) + 1)^2$. Thus, it suffices to show that $\mathrm{Rank}_{\mathbb{Q}}(g_{2n}) = 2^n - 1$ and $\mathrm{Rank}_{\mathbb{Z}_2}(g_{2n}) = n$.

To see that $\mathrm{Rank}_{\mathbb{Z}_2}(g_{2n}) = n$, consider the n rows of $M(g_{2n})$ corresponding to the $(x_1, \ldots, x_n)$-parts $10^{n-1}, 010^{n-2}, \ldots, 0^i 10^{n-i-1}, \ldots, 0^{n-1}1$ of the input. It can easily be observed that all other rows are linear combinations of these n rows (more precisely, if a row corresponds to an input part with 1's in the positions $i_1, i_2, \ldots, i_r$, then this row is the sum of the rows corresponding to the input assignments $0^{i_1-1}10^{n-i_1}$, $0^{i_2-1}10^{n-i_2}, \ldots, 0^{i_r-1}10^{n-i_r}$).

Let J_n denote the $2^n \times 2^n$ matrix with $J_n[i, j] = 1$ for all $1 \leqslant i, j \leqslant 2^n$. It is well known that $2M(g_{2n}) - J_n$ is a Hadamard matrix [4, p. 74–75] and hence $\mathrm{Rank}_{\mathbb{Q}}(2M(g_{2n}) - J_n) = 2^n$. On the other hand, the transformation $M \to 2 \cdot M - J_n$ can increase the rank by at most 1. This actually occurs, since $M(g_{2n})$ possesses a null row, namely the row that corresponds to input 0. Hence $\mathrm{Rank}_{\mathbb{Q}}(g_{2n}) = 2^n - 1$. □

To prove Theorem 1.6, we have to find a function f such that there is a large fooling set $\mathscr{A}(f)$, while the rank of $M(f)$ is significantly smaller than $\mathrm{card}(\mathscr{A}(f))$.

$$M_1 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

Fig. 1. A 1-fooling matrix of rank 3.

For this, it is sufficient to build a δ-fooling matrix M with $\mathrm{Rank}(M)$ significantly smaller than the size of M. (Note that each Boolean matrix of size $2^d \times 2^d$ together with an arbitrary partition of $2d$ variables unambiguously defines a Boolean function of $2d$ variables. Moreover, if this matrix is a δ-fooling matrix, then the set of the 2^d inputs corresponding to the diagonal is a fooling set for f).

We start by presenting (in Fig. 1) a 1-fooling matrix M_1 of size 4×4 with $\mathrm{Rank}_F(M_1) = 3$ for every field F.

That M_1 is singular over every field F is obvious, since the sum of rows 1 and 3 equals the sum of rows 2 and 4. Starting from M_1 we construct a sequence of fooling matrices by defining $M_{d+1} = M_d \otimes M_1$, for $d \geqslant 1$. It is clear that Theorem 1.6 is an immediate consequence of the following lemma.

Lemma 2.11. *M_d is a 1-fooling matrix of size $4^d \times 4^d$ that satisfies $\mathrm{Rank}(M_d) = 3^d$.*

Proof. We proceed by induction on d. For $d = 1$ the matrix $M_1 = [a_{r,s}]_{r,s=1,\ldots,4}$ obviously has the required properties.

Now consider the 1-fooling matrix $M_d = [b_{i,j}]_{i,j=1,\ldots,2^d}$ with $\mathrm{Rank}(M_d) = 3^d$. Since $M_{d+1} = M_d \otimes M_1$, we have $\mathrm{Rank}(M_{d+1}) = 3^{d+1}$, by the Kronecker Fact 2.8. Obviously, M_{d+1} has size $4^{d+1} \times 4^{d+1}$ and we only have to verify that M_{d+1} is a 1-fooling matrix.

We can assume that $M_{d+1} = [c_{(i,r),(j,s)}]_{i,j=1,\ldots,4^d;r,s=1,\ldots,4}$, where $c_{(i,r),(j,s)} = b_{i,j} \cdot a_{r,s}$. Since the diagonal entries of M_d and M_1 are all identical to 1, the diagonal of M_{d+1} consists only of 1's.

Now consider two different diagonal elements of M_{d+1}, i.e., $c_{(i,r),(i,r)}$ and $c_{(j,s),(j,s)}$. If $i = j$, then $r \neq s$ and, since M_1 is a 1-fooling matrix, $c_{(i,r),(i,s)} = 0$ or $c_{(i,s),(i,r)} = 0$. If $i \neq j$, then $b_{i,j} = 0$ or $b_{j,i} = 0$ and again we have the fooling set property $c_{(i,r),(j,s)} = 0$ or $c_{(j,s),(i,r)} = 0$. □

3. The extended Fooling Set method and nondeterministic communication

The aim of this section is to show that the extended Fooling Set method provides tight lower bounds for the nondeterministic communication complexity $\mathrm{ncc}(f)$. We will obtain an "almost" tight bound for deterministic communication as a direct consequence. First we verify that the extended Fooling Set method provides lower bounds.

Observation 3.1. *Let f be a Boolean function. Then*

(i) $\mathrm{ncc}(f) \geqslant \lceil \log_2(\mathrm{Fool}^1(f)) \rceil$.

(ii) $\mathrm{cc}(f) \geqslant \max\{\lceil \log_2(\mathrm{Fool}^1(f)) \rceil, \lceil \log_2(\mathrm{Fool}^0(f)) \rceil\} = \lceil \log_2(\mathrm{Fool}(f)) \rceil$.

Proof. We verify only part (i). Part (ii) follows from part (i), since $\mathrm{ncc}(f), \mathrm{ncc}(\overline{f}) \leqslant \mathrm{cc}(f)$. According to [16], $\mathrm{ncc}(f) = \lceil \log_2(\mathrm{cov}^1(f)) \rceil$, where $\mathrm{cov}^1(f)$ is the minimal number of 1-chromatic submatrices needed to cover the 1's of the communication matrix $M(f)$ of f. Let $\mathscr{A}$ be a 1-fooling set of order k. Then any 1-chromatic submatrix of $M(f)$ can intersect $\mathscr{A}$ in at most k elements. Thus, $\mathrm{cov}^1(f) \geqslant |\mathscr{A}|/k$. By Definition 1.7, this means $\mathrm{cov}^1(f) \geqslant \mathrm{Fool}^1(f)$, which implies (i) by Yao's formula. □

The proof of Theorem 1.6 is based on [10], where it is shown that for covering problems the greedy method provides results close to an optimal solution. We obtain our result by regarding the problem of covering the 1's in $M(f)$ by 1-chromatic submatrices as an optimization problem. In [6], a similar view was taken as a start, but in that paper relaxations of the covering problem ("fractional covers", as suggested by Lovász) were studied, a method quite different from that one used here.

Definition 3.2. Let f be a Boolean function in B^{2n}. The *greedy cover algorithm for* $M(f)$ is described by the following recursive construction:

$$UC_0 := \{(i,j) | M[i,j] = 1\}.$$

Initially, all 1's are "uncovered". For $i \geqslant 1$ we proceed inductively:

Let S_i = a 1-chromatic submatrix of $M(f)$ that covers a maximal number of 1-entries from UC_{i-1}. (In case of a tie, choose the lexicographically smallest such submatrix.) Let $h_i = |UC_{i-1} \cap S_i|$. Then define

$$UC_i = UC_{i-1} - S_i.$$

$$\text{g-cov}^1(f) = \min\{i \geqslant 0 \mid UC_i = \emptyset\}.$$

(This is the number of steps made by the greedy method for constructing a covering of the 1's of f by monochromatic submatrices.)

The following lemma summarizes some further simple observations.

Lemma 3.3. *For all Boolean functions f the following holds*:

(i) $\mathrm{cov}^1(f) \leqslant \text{g-cov}^1(f)$.
(ii) UC_{i-1} *is a 1-fooling set of order* h_i *for* $1 \leqslant i \leqslant \text{g-cov}^1(f)$.
(iii) $|UC_i| = |UC_{i-1}| - h_i$, *for* $1 \leqslant i \leqslant \text{g-cov}^1(f)$.
(iv) $|UC_{\text{g-cov}^1(f)-1}| = h_{\text{g-cov}^1(f)} \geqslant 1$.

Proof. (i) is obvious, since the greedy algorithm constructs a covering with $\text{g-cov}^1(f)$ submatrices. (ii) By construction, h_i is the maximal number of 1's in UC_{i-1} that can be covered by a 1-chromatic submatrix of $M(f)$. (iii), (iv) are obvious from the greedy algorithm and the definition of $\text{g-cov}^1(f)$. □

Lemma 3.4.

$$\text{g-cov}^1(f) - 1 \leqslant \text{Fool}^1(f) \cdot 2 \ln 2 \cdot n.$$

Proof. We let, with the notation from above,

$$B = \max\left\{ \frac{|UC_{i-1}|}{h_i} \mid 1 \leqslant i \leqslant \text{g-cov}^1(f) \right\},$$

and note that by Lemma 3.3(ii) we have $B \leqslant \text{Fool}^1(f)$. Thus, it suffices to show the following.

Claim.

$$B \geqslant (\text{g-cov}^1(f) - 1)/(2 \ln 2 \cdot n).$$

In the following, we prove the claim. By the definition of B, we have $h_i \geqslant (1/B) \cdot |UC_{i-1}|$ for all i, $1 \leqslant i \leqslant \text{g-cov}^1(f)$. Thus, by Lemma 3.3(iii)

$$|UC_i| = |UC_{i-1}| - h_i \leqslant |UC_{i-1}| \cdot (1 - 1/B) \quad \text{for } 1 \leqslant i \leqslant \text{g-cov}^1(f).$$

This implies

$$|UC_i| \leqslant |UC_0|(1 - 1/B)^i < |UC_0| e^{-i/B} \quad \text{for } 1 \leqslant i \leqslant \text{g-cov}^1(f).$$

Using Lemma 3.3(iv), we get

$$1 \leqslant |UC_{\text{g-cov}^1(f)-1}| < |UC_0| e^{-(\text{g-cov}^1(f)-1)/B}.$$

Taking logarithms yields

$$B > (\text{g-cov}^1(f) - 1)/\ln(|UC_0|).$$

The simple observation that $|UC_0| =$ the number of 1's in $M(f) \leqslant 2^{2n} = e^{2 \ln 2 \cdot n}$ yields the claim. □

Proof of Theorem 1.9. We must verify that

$$\log_2(\text{Fool}^1(f)) \leqslant \text{ncc}(f) \leqslant \log_2(\text{Fool}^1(f)) + \log_2(3.6n).$$

The first inequality was established in Observation 3.1. For the second inequality we use

(a) Yao's formula,
(b) Lemma 3.3,
(c) Lemma 3.4,

(d) the (harmless) assumption $\text{Fool}^1(f)\cdot n \geqslant 3$ and the fact that $(1.8-2\ln 2)\cdot 3 \geqslant 1$ to estimate

$$\begin{aligned}
\text{ncc}(f) &\overset{(a)}{=} \lceil \log_2(\text{cov}^1(f)) \rceil \\
&\leqslant \log_2(2\,\text{cov}^1(f)) \\
&\overset{(b)}{\leqslant} \log_2(2\ \text{g-cov}^1(f)) \\
&\overset{(c)}{\leqslant} \log_2(2(\text{Fool}^1(f)\cdot 2\ln 2\cdot n+1)) \\
&\overset{(d)}{\leqslant} \log_2(\text{Fool}^1(f)) + \log_2(3.6n).
\end{aligned}$$

Finally, we prove part (ii) of Theorem 1.9. We have to verify that

$$\log_2(\text{Fool}(f)) \leqslant \text{cc}(f) \leqslant (\log_2(\text{Fool}(f)) + \log_2(3.6n) + 1)^2.$$

The first inequality has already been proved in Observation 3.1. From [11, 2] we know that $\text{cc}(f) \leqslant (\text{ncc}(f)+1)(\text{ncc}(\overline{f})+1)$. Therefore, the second inequality follows directly from part (i). □

4. Conclusion

We have compared two lower bound proof methods for communication complexity. We have shown that the Rank method can be much better than the Fooling Set method, and that the Fooling Set method can be better, but only by a factor of 2, than the Rank method. To complete this comparison into the smallest details the following problems have to be solved.

Open Problem 1. In Theorem 1.3 we prove the existence of a Boolean function f_{2n} with $\text{Fool}_1(f_{2n}) \leqslant 10n$ and $\text{Rank}(f_{2n}) = 2^n$. Find a concrete function f_{2n} with this property. (Note that Theorem 1.5 provides an example of a concrete function g_{2n} with $\text{Fool}_1(g_{2n}) \leqslant (n+1)^2$ and $\text{Rank}(g_{2n}) = 2^n - 1$.)

Open Problem 2. Theorem 1.4 shows that $\text{Fool}_1(f) \leqslant (\text{Rank}(f)+1)^2$ and Theorem 1.6 shows $\text{Fool}_1(h_{2n}) = 2^n$ and $\text{Rank}(h_{2n}) = 3^{n/2}$ for a specific function h_{2n}. Which of these two theorems can be improved? What is the largest constant d such that $\text{fs}(f) \geqslant d \cdot \text{r}(f)$ for a Boolean function f? Note that Theorems 1.4 and 1.6 show that $1.261\ldots \approx (\frac{1}{2}\log_2 3)^{-1} \leqslant d \leqslant 2$. M. Hühne [5] has constructed an example which yields $d \geqslant \frac{1}{2}\log 6 \approx 1.292\ldots$.

Furthermore, we have shown that the extended Fooling Set method provides tight lower bounds for deterministic as well as for nondeterministic communication.

References

[1] H. Abelson, Lower bounds on information transfer in distributed computations, in: *Proc. 19th IEEE Symp. on Foundations of Computer Science* (1978) 151–158.

[2] A.V. Aho, J.D. Ullman and M. Yannakakis, On notions of information transfer in VLSI circuits, in: *Proc. 15th Ann. ACM Symp. on Theory of Computing* (1983) 133–139.

[3] E. Artin, *Geometric Algebra* (Interscience, New York, 1957).

[4] B. Bollobas, *Combinatorics: Set Systems, Hypergraphs, Families of Vectors and Combinatorial Probability* (Cambridge Univ. Press, Cambridge, 1986).

[5] M. Hühne, personal communication.

[6] M. Karchmer, E. Kushilevitz and N. Nisan, Fractional covers and communication complexity, to appear in *SIAM J. Discrete Math.* (Early version in: *Proc. 7th IEEE Conference on Structure in Complexity Theory* (1992) 262–274.)

[7] J. Komlós, On the determinant of (0,1)-matrices, *Studia Sci. Math. Hungar.* **2** (1965) 7–21.

[8] J. Komlós, On the determinant of random matrices, *Studia Sci. Math. Hungar.* **3** (1968) 387–399.

[9] Th. Lengauer, VLSI theory, in: J. van Leeuwen, ed., *Handbook of Theoretical Computer Science, Vol. A, Algorithms and Complexity* (Elsevier, Amsterdam, 1990) 835–868.

[10] L. Lovász, On the ratio of optimal integral and fractional covers, *Discrete Math.* **13** (1975) 384–390.

[11] L. Lovász, Communication complexity: a survey, in: B. Korte et al., eds., *Paths, Flows, and VLSI-Layout* (Springer, Berlin, 1990) 235–265.

[12] K. Mehlhorn and E. Schmidt, Las Vegas is better than determinism in VLSI and distributed computing, in: *Proc. 14th Ann. ACM Symp. on Theory of Computing* (1982) 330–337.

[13] A. Orlitsky and A. El Gamal, Communication complexity, in: Y.S. Abu-Mustafa, ed., *Complexity in Information Theory* (Springer, Berlin, 1988) 16–61.

[14] R. Raz and B. Spieker, On the "log rank"-conjecture in communication complexity, in: *Proc. 34th IEEE Symp. on Foundations of Computer Science* (1993) 168–176.

[15] A.C. Yao, Some complexity questions related to distributive computing, in: *Proc. 11th Ann. ACM Symp. on Theory of Computing* (1979) 209–213.

[16] A.C. Yao, The entropic limitations on VLSI computations, in: *Proc. 13th Ann. ACM Symp. on Theory of Computing* (1981) 308–311.