MAX-PLANCK-INSTITUT FÜR INFORMATIK

Multi–Party Protocols and Spectral Norms

Technical Report No. MPII–1993–132

Vince Grolmusz
Max Planck Institute for Computer Science
and Eötvös University

August 13, 1993

INFORMATIK

Im Stadtwald
66123 Saarbrücken
Germany

Multi–Party Protocols and Spectral Norms

Technical Report No. MPII–1993–132

Vince Grolmusz
Max Planck Institute for Computer Science
and Eötvös University

August 13, 1993

Multi-Party Protocols and Spectral Norms

Technical Report No. MPII–1993–132

Vince Grolmusz

Max Planck Institute and Eötvös University

ABSTRACT:

Let f be a Boolean function of n variables with L_1 spectral norm $\mathrm{L}_1(f) > n^\varepsilon$ for some positive ε. Then f can be computed by a $O(\log \mathrm{L}_1(f))$ player multi–party protocol with $O(\log^3 \mathrm{L}_1(f))$ communication.

Address: Max Planck Institute for Computer Science, Im Stadtwald, D-66123 Saarbruecken, GERMANY; email: grolmusz@mpi-sb.mpg.de

1. INTRODUCTION

1.1 Multi–party games

The *multi–party communication game*, defined by *Chandra, Furst* and *Lipton* [CFL], is an interesting generalization of the 2–party communication game. In this game, k players: $P_1, P_2..., P_k$ intend to compute a Boolean function $g(x_1, x_2, ..., x_n) : \{0,1\}^n \to \{0,1\}$. On set $S = \{x_1, x_2, ..., x_n\}$ of variables there is a fixed partition of k classes $A_1, A_2, ..., A_k$, and player P_i knows every variable, *except* those in 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 $g(x_1, x_2, ..., x_n)$, such that at the end of the computation, every player knows this value. The cost of the computation is the number of bits written on the blackboard for the given $x = (x_1, x_2, ..., x_n)$ and $A = (A_1, A_2, ..., A_k)$. The cost of a multi–party protocol is the maximum number of bits communicated for any x from $\{0,1\}^n$ and the given A. The k-party communication complexity, $C_A^{(k)}(g)$, of a function g, with respect to partition A is the minimum of costs of those k-party protocols which compute g. The k-party symmetric communication complexity og g is defined as

$$C^{(k)}(g) = \max_A C_A^{(k)}(g),$$

where the maximum is taken over all k–partitions of set $\{x_1, x_2, ..., x_n\}$.

The theory of the 2–party communication games is well developed (see [L] for a survey), but much less is known about the multi–party communication complexity of functions. As a general upper bound both for two and more players, P_1 can compute any function of A with n bits of communication: 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 $g(A)$.

For two players, the communication complexity of a function is known to be between the rank and the logarithm of the rank of a $2^n \times 2^n$ matrix, containing the values of f for all possible input allocations. Better upper bounds were given for special classes of functions by *Lovász* and *Saks* [LS], using extensively lattice–theory and Moebius functions. For more than two players, no analogue results were known.

Chandra, Furst and *Lipton* [CFL] proved non–trivial upper and lower bounds for the k–communication complexity of a specific function, using intricate Ramsey-theoretic arguments.

An important progress was made by *Babai, Nisan* and *Szegedy*, [BNS], proving an $\Omega(\frac{n}{4^k})$ lower bound for the k–party communication complexity of the GIP function. It is proved in [G] that their lower bound is close to the optimal.

We proved in [G3] that any function, computed by a depth–2 MOD p circuit of size N can be computed with p players and $O(p)$ bits of communication, and the number of communicated bits do not depend on N.

In this paper we give a general non–trivial upper bound to the symmetric multi–party communication complexity of *arbitrary* Boolean functions. Our bound depend on the L_1 *spectral norm* of function f.

1.2 Spectral Norms

There is a vast literature on representing the Boolean functions by polynomials above some field (see, e.g. [ABFR], [Be], [BRS], [BS], [LMN], [NS], [Sm]). One reason for this may be that the polynomials offer a more developed machinery than the "pure" Boolean functions. One tool in this machinery is the Fourier–transform of Boolean functions [LMN], [BS], [KKL], [NS]:

Let us represent Boolean function f as a function $f:\{-1,1\}^n \to \{-1,1\}$ where -1 stays for "true". The set of all real valued functions over $\{-1,1\}^n$ forms a 2^n dimensional vector–space over the reals with an inner product:

$$< f,g >= 2^{-n} \sum_{x\in\{-1,1\}^n} f(x)g(x).$$

Let us define for $\alpha = (\alpha_1, \alpha_2, ..., \alpha_n) \in \{0,1\}^n$

$$X^\alpha = \prod_{i=1}^n x_i^{\alpha_i}.$$

The monomials X^α for $\alpha \in \{0,1\}^n$ form an *orthonormal basis*:

$$< X^\alpha, X^\alpha >= 1$$

$$< X^\alpha, X^\beta >= 0, \text{ for } \alpha \neq \beta.$$

Consequently, any function $h:\{-1,1\}^n \to \mathbf{R}$ can be uniquely expressed as

(1) $$h(x_1, x_2, ..., x_n) = \sum_{\alpha\in\{0,1\}^n} a_\alpha X^\alpha$$

The right–hand–side of equation (1) is called the *Fourier–transform* of function h. Set $\{a_\alpha : \alpha \in \{0,1\}^n\}$ is called *spectral coefficients* or the *spectrum* of h.

Note. *If h is a Boolean function, then*

$$a_\alpha = Pr(h(x) = X^\alpha) - Pr(h(x) \neq X^\alpha)$$

where x is chosen uniformly and randomly in $\{-1,1\}^n$.

Several *spectral norms* of h can be defined:
The L_1 norm of h is:

$$L_1(h) = \sum_{\alpha\in\{0,1\}^n} |a_\alpha|$$

The L_2 norm:

$$L_2(h) = \left(\sum_{\alpha\in\{0,1\}^n} a_\alpha^2\right)^{\frac{1}{2}} =< h,h >^{\frac{1}{2}}$$

and the L_∞ norm:

$$L_\infty(h) = \max_{\alpha\in\{0,1\}^n} |a_\alpha|.$$

We mention here some nice results concerning the spectral and computability properties of Boolean functions.

Linial, Mansour and *Nisan* [LMN] proved that if f is a Boolean function computed by a depth–d size–M Boolean circuit, then

$$\sum_{\substack{\alpha\in\{0,1\}^n \\ \sum_{i=1}^n \alpha_i \geq t}} a_\alpha^2 \leq M 2^{-\frac{1}{4}t^{\frac{1}{d+3}}},$$

i.e. the L_2 norm of the end–segments of the Fourrier–transform of f are exponentially small.

Bruck and *Smolensky* [BS] proved that if f is a Boolean function with small L_1 norm, then it can be represented as the sign of a *sparse* polynomial. More exactly:

Theorem 1. ([BS] Theorem 1, Lemma 1)
Given an $f : \{-1,1\}^n \to \{-1,1\}$ with its Fourrier–transform

$$f(x) = \sum_{\alpha\in\{0,1\}^n} a_\alpha X^\alpha.$$

Let

$$p_\alpha = \frac{|a_\alpha|}{L_1(f)}$$

for $\alpha \in \{0,1\}^n$, and let

$$Z_i = \text{sgn } (a_\alpha) X_\alpha \textit{ with probability } p_\alpha.$$

Define

$$G(x) = \sum_{i=1}^{N} Z_i, \textit{ where } N = 2nL_1^2(f).$$

Then

$$Pr(\forall x \in \{-1,1\}^n : f(x) = \text{sgn } (G(x))) > 0. \tag{2}$$

1.3 Our Results

Our main result is the following

Theorem 2. *Let f be an arbitrary Boolean function of n variables. Then the k–party symmetric communication complexity of f,*

$$C^{(k)}(f) = O\left(k^2 \log\left(n\mathrm{L}_1(f)\right) \left\lceil \frac{n\mathrm{L}_1^2(f)}{2^k} \right\rceil\right).$$

Corollary 3. *Let f be an arbitrary Boolean function of n variables. Let $k = \Omega(\log(n\mathrm{L}_1(f)))$. Then*

$$C^{(k)}(f) = O(\log^3(n\mathrm{L}_1(f))).$$

∎

In other words, if f is a Boolean function with its L_1 spectral norm bounded by a polynomial in n, then its *symmetric* k–party communication complexity is at most $O(\log^3 n)$, with $k = \Omega(\log n)$. Or, in another setting:

Corollary 4. *Suppose that $\mathrm{L}_1(f) > n^\varepsilon$ for some $\varepsilon > 0$. Then there exists a multi–party protocol with $\Omega(\log \mathrm{L}_1(f))$ players and of $O(\log^3 \mathrm{L}_1(f))$ communication which computes f.* ∎

2. PROOF OF THEOREM 2

Let f be an arbitrary Boolean function. By Theorem 1, there exists a (fixed, "deterministic") function $G(x)$ such that

$$\forall x \in \{-1,1\}^n \quad f(x) = \mathrm{sgn}\ G(x)$$

holds. $G(x)$ is of the form

$$G(x) = \sum_{i=1}^{N} Z_i,$$

where $N = 2n\mathrm{L}_1^2(f)$. Let $G_1(x)$ be the sum of Z_i's with positive sign, and let $G_2(x)$ be the sum of $(-Z_i)$'s, where Z_i has a negative sign. So:

$$G(x) = G_1(x) - G_2(x),$$

and G_1 has N_1 terms, G_2 has N_2 terms, $N_1 + N_2 = N$.

Let us observe that $G_j(x)$ is the sum of N_j terms of form

$$X^\alpha = \prod_{i=1}^{n} x_i^{\alpha_i} = \prod_{i:\alpha_i=1} x_i$$

for $j = 1, 2$.

Clearly,

$$X^\alpha = \begin{cases} -1, & \text{if } |\{i : x_i = -1, \alpha_i = 1\}| \text{ is odd} \\ 1 & \text{otherwise} \end{cases}$$

For $j = 1,2$ let b_j the number (counting the possible multiplicity) of those terms X^α in $G_j(x)$ for which $|\{i : x_i = -1, \alpha_i = 1\}|$ is odd . Then $G_j(x) = (N_j - b_j) - b_j = N_j - 2b_j$, so:

$$G(x) = G_1(x) - G_2(x) = N_1 - N_2 + 2b_2 - 2b_1. \tag{3}$$

Let us denote

$$y_i = \begin{cases} 1, & \text{if } x_i = -1 \\ 0, & \text{if } x_i = 1 \end{cases}$$

then

$$X^\alpha = -1 \Longleftrightarrow \sum_{i=1}^{n} y_i \alpha_i = 1 \bmod 2.$$

Let us form a matrix $M^{(j)}$ with N_j rows and n columns, for $j = 1,2$. Each row is corresponded to a term X^α in $G_j(x)$, and the i^{th} entry of that row is $y_i \alpha_i$.

Obviously, the number of those rows of $M^{(j)}$ which have odd sum is equal to b_j.

Suppose now that we are given Boolean function $f(x_1, x_2, ..., x_n)$, players $P_1, P_2, ..., P_k$ and a k-partition $A = (A_1, A_2, ..., A_k)$ of the set $\{x_1, x_2, ..., x_n\}$. We assume that player P_ℓ knows function f, partition A, functions $G_1(x)$, $G_2(x)$, and the values of all variables, except those in A_ℓ, for $\ell = 1, 2, ..., k$. Then the players, without any communication can compute privately matrices $M^{(1)}$ and $M^{(2)}$, and exactly those entries of these matrices will be not known for player P_ℓ which were corresponded to variables in class A_ℓ. The set of these entries will be called B_ℓ, for $\ell = 1, 2, ..., k$. The following lemma shows a protocol by which the players can first compute b_1 and then b_2, and consequently, $G(x)$ and $f(x) = \operatorname{sgn} G(x)$, by equation (3).

Lemma 5. *Let $M \in \{0,1\}^{m \times n}$, $M = \{m_{ij}\}$, and let $B = \{B_1, B_2, ..., B_k\}$ a partition of the set $\{m_{ij} : 1 \le i \le m, 1 \le j \le n\}$, such that player P_ℓ knows every m_{ij} except those in B_ℓ, for $\ell = 1, 2, ..., k$. Then there exists a k–party protocol which computes the number of the rows with odd sum in M with communicating*

$$O\left(k^2 \log m \left\lceil \frac{m}{2^k} \right\rceil\right)$$

bits.

Proof. First, the players compute a matrix $Q \in \{0,1\}^{m \times k}$ from M, with no communication: for each row of M a row of Q is corresponded; the first element of row j of Q is the mod 2 sum of those entries of the j^{th} row of M which are the elements of B_1 at the same time. Analogously, the i^{th} element of row j of Q is the mod 2 sum of those entries of the j^{th} row of M which are the elements of B_i at the same time.
Clearly, the number of rows with odd sum in M and in Q is the same. Moreover, player P_ℓ knows every column of matrix Q, except column ℓ, for $\ell = 1, 2, ..., k$.

With an additional assumption Lemma 6 gives a protocol with $O(k^2 \log m)$ communication:

Lemma 6. *Let $\beta \in \{0,1\}^k$. Suppose it is known to each player that β does not occur as a row of Q. Then there exists a k–party protocol which computes the number of the odd rows with a communication of $O(k^2 \log m)$ bits.*

Proof. Without restricting the generality we may suppose that β is the all–1 vector of length k.

Let $ODD(\gamma_1\gamma_2...\gamma_\ell)$ and $EVEN(\gamma_1\gamma_2...\gamma_\ell)$ denote the number of those rows of Q which have odd (respectively, even) sums, and they begin with $\gamma_1\gamma_2...\gamma_\ell$, $\ell \leq k$, $\gamma_i \in \{0,1\}$. For example, P_1 do not know the first column of Q, but he can communicate $ODD(0) + EVEN(1)$ if P_1 counts those rows which has odd sum in its second through kth position. Similarly P_2 can communicate $ODD(10)+EVEN(11)$ if he counts those rows which begins with 1, and the sum of their first, 3rd, 4th,...,kth elements is odd.

This observation motivates the following protocol:

PROTOCOL ODDCOUNT

The goal: to compute b, the number of rows with odd sum in Q. Number b will be the sum of values u_i announced by player P_i, $i = 1, 2, ..., k$.

P_1 announces $u_1 = ODD(0) + EVEN(1)$.

remark: $b = u_1 + ODD(1) - EVEN(1)$.

P_2 announces $u_2 = ODD(10) + EVEN(11) - EVEN(10) - ODD(11)$.

remark: $b = u_1 + u_2 - 2EVEN(11) + 2ODD(11)$

P_3 announces $u_3 = 2ODD(110) + 2EVEN(111) - 2EVEN(110) - 2ODD(111)$.

remark: $b = u_1 + u_2 + u_3 - 4EVEN(111) + 3ODD(111)$

.

.

.

P_i announces $u_i = 2^{i-2}ODD(11...10) + 2^{i-2}EVEN(11...11) - 2^{i-2}EVEN(11...10) - 2^{i-2}ODD(11...11)$

remark: $b = \sum_{j=1}^{i} u_j - 2^{i-1}EVEN(11...1) + (2^{i-1} - 1)ODD(11...1)$, where the bit–sequences in each bracket have length i.

After P_k announces u_k the players privately add up the u_i's from $i = 1$ through k. Let us remark that

$$b = \sum_{j=1}^{k} u_j - 2^{k-1}EVEN(11...1) + (2^{k-1} - 1)ODD(11...1).$$

However, as we assumed at the beginning, there are no all–1 rows in Q, so

$$b = \sum_{j=1}^{k} u_j$$

and we are done. Each u_i can be communicated using $O(k \log m)$ bits, so the total communication is $O(k^2 \log m)$. ∎

Now we return to the proof of Lemma 5. Let us divide the rows of matrix Q into blocks of $2^{k-1} - 1$ contiguous rows plus a leftover of at most $2^{k-1} - 1$ rows. The players cooperatively determine the number of the odd rows in each block, and then privately add up the results.

Next we show how to obtain the number of the odd rows for a single block at the cost of $O(k^2 \log m)$ bits of communication. P_1 knows all the columns, except the first, so he knows at most $2^{k-1} - 1$ rows of length $k - 1$ in a block, so he can find an $\beta' \in \{0,1\}^{k-1}$, $\beta' = (\beta_2, \beta_3, \ldots, \beta_k)$ which is not a row of the $k - 1$ column wide part of the block seen by P_1. Let $\beta = (1, \beta_2, \beta_3, \ldots, \beta_k)$. Then β does not occur as a row in this block. So if P_0 communicates β, and they play protocol ODDCOUNT of Lemma 6 for a given block. They use $k^2 \log m$ bits for a block, and, since there are at most $\left\lceil \frac{m}{2^{k-1}-1} \right\rceil$ blocks, the total communication is

$$O\left(k^2 \log m \left\lceil \frac{m}{2^k} \right\rceil\right).$$

∎

REFERENCES

[A] E. Allender: A note on the power of threshold circuits, Proc. 30th IEEE FOCS, 1989, pp. 580-584

[AB] M. Ajtai, M. Ben–Or: A theorem on probabilistic constant depth computations, Proc. 16th ACM STOC, 1984, pp. 471-474

[ABFR] J. Aspnes, R. Beigel, M. Furst, S. Rudich: The expressive power of voting polynomials, Proc. 23rd ACM STOC, 1991, pp. 402-409

[Ba] D. A. Barrington: Bounded-width polynomial size branching programs recognize exactly those languages in NC^1, Proc. 18th ACM STOC, 1986, 1-5

[BBR] D. A. Barrington, R. Beigel, S. Rudich: Representing Boolean functions as polynomials modulo composite numbers, Proc. 24th ACM STOC, 1992, pp. 455-461

[Be] When do extra MAJORITY gates help?, Proc. 24th ACM STOC, 1992, pp. 450-454

[BG] C.G. Bennet, J. Gill: Realtive to a random oracle A, $P^A \neq NP^A \neq co - NP^A$ with probability 1. SIAM J. on Computing, 10, (1981) pp. 96–113.

[BRS] R. Beigel, N. Reingold, D. Spielman: The perceptron strikes back, Proc. 6th Annual Conference on Structure in Complexity Theory, IEEE Comp. Soc. Press, 1991

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

[BS] J. Bruck, R. Smolensky: Polynomial threshold functions, AC^0 functions and spectral norms, Proc. 32nd IEEE FOCS, 1991, pp. 632–641

[BT] R. Beigel, J. Tarui: On ACC, Proc. 32nd IEEE FOCS, 1991, pp. 783-792

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

[CG] B. Chor, O. Goldreich: Unbiased bits from sources of weak randomness and probabilistic communication complexity, Proc. 26th IEEE FOCS, 1985, pp. 429-442

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

[G2] V. Grolmusz: Circuits and Multi–Party Protocols, Technical Report No. MPII-1992-104, Max Planck Institute for Computer Science, Saarbruecken, Germany, 1992,

[G3] V. Grolmusz: Separating the communication complexities of MOD m and MOD p circuits, Proc. 33rd IEEE FOCS, 1992, pp. 278-287

[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.

[JKS] J. JaJa, V.K. Prasanna Kumar, J. Simon: Information transfer under different stes of protocols, SIAM J. on Computing, 13 (1984) pp. 840-849

[KM] J. Kahn, R. Meshulam: On mod p Transversals, Combinatorica, 1991, (11) No. 1. pp. 17–22.

[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. 539–550

[KKL] J. Kahn, G. Kalai, N. Linial: The influence of variables on Boolean functions, Proc. 29th IEEE FOCS, 1988, pp. 68–80

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

[LMN] N. Linial, Y. Mansour, N. Nisan: Constant depth circuits, Fourier transform and learnability, Proc. Proc. 30th IEEE FOCS, 1989, pp. 574–579

[LS] L. Lovász, M. Saks: Lattices, Moebius functions and communication complexity, Proc. 29th IEEE FOCS, pp. 81–90

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

[NS] N. Nisan, M. Szegedy: On the degree of Boolean functions as real polynomials, Proc. 24th ACM STOC, 1992, pp. 462–467

[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. 30th IEEE FOCS, 1989, pp. 574-579

[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 STOC, pp. 77-82, (1987).

[Ya1] 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.