MULTI-PARTY PROTOCOLS

by

Ashok K. Chandra[1] *Merrick L. Furst*[2] *Richard J. Lipton*[3]

1. Introduction. Many different types of interprocess communication have been examined from a complexity point of view [SP, Y]. We study a new model, in which a collection of processes

$$P_0, \ldots, P_{k-1}$$

that share information about a set of integers

$$\{a_0, \ldots, a_{k-1}\},$$

communicate to determine a 0-1 predicate of the numbers.

For example, suppose k poker players are sitting around a table, and each one is holding a number to his forehead for the others to see. This situation fits our model and we can determine how many bits the players need to communicate in order for all of them to know if the sum of the numbers is greater than n.

In this new model, tremendous sharing of information is allowed, while no single party is given enough information to determine the predicate on its own. Formally, each P_i has access to every a_j except for a_i. For simplicity, we only allow the parties to communicate as follows. At time $t = 0$, process P_0, examining the numbers it knows, broadcasts one bit. At time $t = 1$, process P_1, examining the numbers it knows, and the bit sent by P_0, broadcasts one bit. This continues in a cyclic fashion until, at a prearranged time , the processes are required to halt, with either all accepting or all rejecting.

The novelty of this shared-information model is apparent once three or more processes are involved. With three processes and three integers, each input can be known by two parties, without any one knowing all three. In this paper we prove both upper and lower bounds on the complexity of multi-party, shared-information protocols (MPP's) for determining certain basic predicates of sets of integers. Our proofs appeal to geometry and Ramsey-like counting arguments. The bounds we prove are unusual in the sense that they are stronger than might naively be expected, and tight—even though we don't know exactly what they are.

Multi-party protocols are a basic and important model of communication in any system, practical or theoretical, where there is shared information. We show that MPP's arise in the study of lower bounds for branching programs. For this model, the standard

[1] Mathematical Sciences Department, IBM Watson Research Center, Yorktown Heights, NY.
[2] Department of Computer Science, Carnegie Mellon University, Pittsburgh, Pa.
[3] Department of EECS, Princeton University, Princeton, NJ.

© 1983 ACM 0-89791-099-0/83/004/0094 $00.75

approach to proving lower bounds—demonstrating that a certain amount of progress must be made, and that no step makes more than δ progress, for some small δ—doesn't seem to work. The branching program model is too complicated and there doesn't seem to be a way to define progress appropriately. To prove non-trivial lower bounds a new insight is needed. Our definition of MPP's is motivated by the search for such a new technique. In the last section we make the connection between MPP's and branching programs, and show how the MPP lower bounds can be applied to obtain new lower bounds on time-space trade-offs.

2. Preliminaries.

From now on we assume that there are k-processes $P_0, \ldots, P_{k-1}$, and k variables $a_0, \ldots, a_k$ that range over the integers from 1 to n. Each process has access to the a_j, except that P_i is denied access to a_i. We study what happens when k is fixed and n grows.

DEFINITION. A *broadcast history* $b \in (0+1)^*$ is a record of all the bits transmitted by the processes up to a certain point in time.

DEFINITION. A *k-party protocol*, or just a *protocol* when k is known, is a deterministic algorithm that, for each process P_i, determines from the numbers P_i knows and the broadcast history, what bit P_i should transmit at times i, $i+k$, ...

DEFINITION. Let $H_n = \{1, \ldots, n\}^k$ be a k-dimensional hypercube. Each point $\langle v_0, v_1, \ldots, v_{k-1}\rangle \in H_n$ describes a *situation*, or assignment to the variables $a_0, \ldots, a_{k-1}$.

DEFINITION. Let Φ be a protocol for k processes working with integers in the range 1 to n. For each $\overline{v} \in H_n$, Φ determines a string $\Phi(\overline{v}) \in (0+1)^*$ that is the complete broadcast history for the processes in situation $\overline{v}$.

The following technical definition of validity for an assignment of broadcast histories is used to define the more natural notion of validity of protocols, and is motivated by its use in Lemma 2.1.

DEFINITION. Let Q be a 0–1 predicate of k integers in the range 1 to n. An assignment Φ of broadcast histories to H_n is *valid* for Q if, for every $\overline{v}, \overline{w} \in H_n$ that differ in exactly one component,

$$\Phi(\overline{v}) \neq \Phi(\overline{w}) \text{ whenever } Q(\overline{v}) \neq Q(\overline{w}).$$

DEFINITION. A protocol Φ is *valid* for a predicate Q if Φ determines a valid broadcast history for Q.

LEMMA 2.1. *A k-party protocol Φ can be used by processes $P_0, \ldots, P_{k-1}$ to determine a predicate Q if and only if Φ is valid for Q.*

Proof. Suppose Φ is not valid for Q. Then there are two points $\overline{w}$ and $\overline{v}$ such that

(a) $Q(\overline{w}) \neq Q(\overline{v})$,

(b) $\overline{w}$ and $\overline{v}$ differ only in coordinate i, and

(c) $\Phi(\overline{w}) = \Phi(\overline{v})$.

Since process P_i cannot see coordinate i it must behave in exactly the same way in situations $\overline{w}$ and $\overline{v}$. Thus, P_i must accept or reject incorrectly at one of $\overline{w}$ or $\overline{v}$. Therefore, Φ cannot be used by the parties to correctly determine Q.

The other direction is straightforward. ▮

DEFINITION. The *cost* of a protocol Φ on H_n is the maximum length of a complete broadcast history on situations in H_n, *i.e.*,

$$\text{cost}_\Phi(n) = \max_{\overline{v} \in H_n} |\Phi(\overline{v})|.$$

DEFINITION. The *complexity* of a predicate Q is the minimum, over all valid protocols Φ for Q, of the cost of Φ. In this setting, complexity is a non-uniform function of n.

3. Lower Bound.

We now examine the inherent communication complexity of multi-party protocols. There is a relationship between Ramsey coloring numbers and the number of bits that processes must communicate in order to determine basic predicates. See [GRS] or [G] for background on Ramsey theory.

DEFINITION. The predicate *Exactly-n* is true of k integers $a_0, \ldots, a_{k-1}$ if and only if $a_0 + \cdots + a_{k-1} = n$.

We prove a lower bound on the number of bits k processes have to communicate in order to determine Exactly-n. This particular predicate is chosen for convenience, and our results hold for all predicates of this type. As has been noted before, lower bounds on exactly-equal-to predicates imply lower bounds on threshold predicates ("$\sum a_i \geq n$") [FSS].

The structure of the lower bound proof is as follows. Let $H_n = \{1, \ldots, n\}^k$ be the set of situations. Let $S_n = \{\overline{v} \in H_n \mid \sum \overline{v} = n\}$ be the hyperplane on which the processes must "accept", as shown in Figure 1. We prove that if a protocol determines too few distinct broadcast histories on the plane S_n, then it is not valid for Exactly-n. Thus, to be valid, a protocol must force the parties to transmit enough bits to ensure that there are enough distinct broadcast histories.

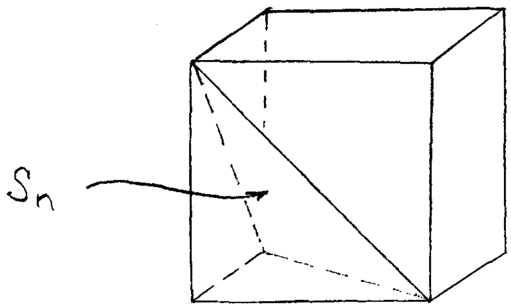

Figure 1. H_n and S_n.

We first show that protocols do not assign broadcast histories to situations in H_n in an arbitrary manner.

DEFINITION. A set of k distinct points $\overline{v}_0, \ldots, \overline{v}_{k-1} \in H_n$ is a *forbidden k-pattern* if there is a point $\overline{w} \in H_n$ such that, for each i, $\overline{w}$ differs from $\overline{v}_i$ only in coordinate i. (For the case $k = 3$, forbidden k-patterns on the plane S_n are just equilateral triangles.)

LEMMA 3.1. Let Φ be a protocol for processes $P_0, \ldots, P_{k-1}$. Let $H_n = \{1, \ldots, n\}^k$ be the hypercube of situations, and let $\overline{w} = \langle w_0, \ldots, w_{k-1} \rangle$ be any point in H_n. If $\overline{v}_0, \ldots, \overline{v}_{k-1}$ is a forbidden k-pattern for $\overline{w}$, and the complete broadcast histories at the $\overline{v}_j$ are identical, *i.e.*,

$$\alpha = \Phi(\overline{v}_0) = \Phi(\overline{v}_1) = \cdots = \Phi(\overline{v}_{k-1}),$$

then $\Phi(\overline{w}) = \alpha$.

Proof. By induction on the length of the broadcast history we prove that, for all i, the i^{th} bit broadcast at $\overline{w}$ is the same as the i^{th} bit broadcast at $\overline{v}_{i(\text{mod } k)}$.

Base: If $|\alpha| = 0$, then every process immediately halts on $\overline{w}$ since it immediately halts on $\overline{v}_i$.

Induction: Suppose, for all histories of length $< t$, the inductive hypothesis is true. Consider the t^{th} bit to be broadcast. By induction, $P_{t(\text{mod } k)}$ sees the same broadcast history at $\overline{w}$ and at $\overline{v}_{t(\text{mod } k)}$, and hence, $P_{t(\text{mod } k)}$ broadcasts the same bit at time t at $\overline{w}$ as it broadcasts at time t at $\overline{v}_{t(\text{mod } k)}$. ∎.

Note that this Lemma is true for all protocols, not just valid ones.

LEMMA 3.2. A k-party protocol Φ is not valid for the predicate Exactly-n if it assigns the same broadcast history to all the points of a forbidden k-pattern of S_n.

Proof. Suppose $\overline{v}_0, \ldots, \overline{v}_{k-1}$, is a forbidden k-pattern on S_n for the point $\overline{w}$, such that

$$\alpha = \Phi(\overline{v}_0) = \Phi(\overline{v}_1) = \cdots = \Phi(\overline{v}_{k-1}).$$

By geometry, $\overline{w}$ is not on S_n, and by Lemma 3.1, $\Phi(\overline{w}) = \alpha$. Therefore, Φ is not valid for Exactly-n. ∎

DEFINITION. The integer $\chi_k(n)$ is the smallest number of colors required to color the points of S_n so that no forbidden k-pattern on S_n is colored monochromatically.

THEOREM 3.3. *The complexity of any k-party protocol for Exactly-n is bounded below by the logarithm of the number of colors required to color S_n so that no forbidden k-patterns are colored monochromatically.*

Proof. Consider a protocol Φ for Exactly-n. Let each distinct broadcast history $\Phi(\overline{v})$, for $\overline{v} \in H_n$, define a color. If some forbidden k-pattern on S_n is colored monochromatically, then, by Lemma 3.2, Φ is not valid for Exactly-n. This is a contradiction. Therefore, there must be more than $\chi_k(n)$ distinct broadcast histories, and hence, in some situation, at least $\log(\chi_k(n))$ bits must be communicated. ∎

DEFINITION. Let

$$B_k(n) = \{(v_0, \ldots, v_{k-1}) \mid 1 \leq v_i \leq n\}.$$

For $0 \leq i \leq k-1$, let $\overline{e}_i$ be the vector whose j^{th} coordinate is δ_{ij}.

We now state a powerful Theorem from Ramsey theory. It is actually a corollary of a stronger result and can be thought of as a generalized pigeon-hole principle.

THEOREM 3.4. *(Gallai [G,GRS]) For every c and k, there exists a λ and an n, such that every c-coloring of $B_k(n)$ contains k distinct vectors*

$$\overline{v},\ \overline{v} + \lambda\overline{e}_0,\ \ldots,\ \overline{v} + \lambda\overline{e}_{k-1}$$

that are identically colored.

We are now in a position to prove the main lower bound.

THEOREM 3.5. *Threshold has multi-party protocol complexity greater than any constant.*

Proof. By Theorem 3.3 it suffices to prove that for a fixed k, $\chi_k(n)$ is unbounded. To get a contradiction, assume that $\chi_k(n)$ is bounded by some constant c.

Define the projection p from S_n to $B_{k-1}(n)$ by

$$p(x_0, \ldots, x_{k-1}) = (x_0, \ldots, x_{k-2}).$$

Since $\chi_k(n) \leq c$, there is a c-coloring C of S_n in which no forbidden k-pattern is colored monochromatically. The mapping p is one-to-one, so C and p induce, in a natural way, a c-coloring of the points in $p(S_n)$. Consider a smaller set, $B_{k-1}(n/4k)$. It is a subset of $p(S_n)$ and is thus also c-colored via p. For large enough n, Gallai's Theorem implies the existence of $k-1$ distinct vectors

$$\overline{v},\ \overline{v} + \lambda\overline{e}_0,\ \ldots,\ \overline{v} + \lambda\overline{e}_{k-2},$$

in $B_{k-1}(n/4k)$ that are colored identically.

Let $\overline{v} = (v_0, \ldots, v_{k-2})$, and let

$$\overline{w} = (v_0, \ldots, v_{k-2}, n - s - \lambda),$$

where $s = v_0 + v_1 + \cdots + v_{k-2}$. Since $s > 0$ and $\lambda > 0$, the last coordinate of $\overline{w}$ is $\leq n$. Furthermore, $1 \leq n - s - \lambda$ since s is at most $(k-1)n/4k$ and λ is at most $n/4k$. Thus, $\overline{w}$ is a situation in H_n.

We now have a contradiction since the points

$$p^{-1}(\overline{v}),\ p^{-1}(\overline{v} + \lambda\overline{e}_0),\ \ldots,\ p^{-1}(\overline{v} + \lambda\overline{e}_{k-2})$$

form a monochromatic forbidden k-pattern for $\overline{w}$. ∎

4. Upper bound.

In this section we prove that, within an additive constant, the lower bound of $\log(\chi_k(n))$ bits proved in the previous section is also an upper bound. That is, we exhibit a cost $[\log(\chi_k(n)) + k]$-bit k-party protocol that is valid for the predicate Exactly-n.

Color the hyperplane $S_n = \{\overline{v} \in H_n \mid \sum \overline{v} = n\}$ of H_n with $\chi_k(n)$ colors such that no forbidden k-pattern is monochromatic. Let $\langle v_0, \ldots, v_{k-1}\rangle$ be a situation in H_n. Let process P_i know every number but v_i. Each process P_i computes the point

$$\overline{w}_i = \left\langle v_0, \ldots, v_{i-1}, n - (\textstyle\sum_{j \neq i} v_j), v_{i+1}, \ldots, v_{k-1} \right\rangle \in S_n.$$

At time $t = 0$, P_0 broadcasts bit 0 of the color at $\overline{w}_0$. At $t = 1$, P_1 broadcasts bit 1 of the color at $\overline{w}_1$. This continues until $\log(\chi_k(n))$ bits are transmitted. Then, each P_i in turn transmits a 1 if and only if the color it sees matches the color transmitted in the first $\log(\chi_k(n))$ bits. Finally, the processes halt and accept if and only if every process braodcast a 1 in the last phase.

THEOREM 4.1. *The above protocol is valid for Exactly-n.*

Proof. The points $\overline{w}_i$ are a forbidden k-pattern in S_n. Thus, if the processes agree that all are the same color, the points must coincide. Hence $\langle v_0, \ldots, v_{k-1}\rangle \in S_n$ and the processes correctly accept.

If $\langle v_0, \ldots, v_{k-1}\rangle \not\in S_n$, then the processes cannot all see the same color since the forbidden points are not monochromatic. Therefore, the processes correctly reject. ∎

Note that even though χ_k is not known, the optimal protocol can be implemented after an exhaustve search. Although precise bounds are not known, we can relate χ_k to classic Ramsey numbers.

DEFINITION. Let $C_k(N)$ be the minimum number of colors needed to color $1, \ldots, N$ such that no length-k arithmetic progression is colored monochromatically.

THEOREM 4.2. *For $N = kn$,*

$$\chi_k(n) \leq C_k(N).$$

Proof. Define the map q from S_n to $\{1, \ldots, N\}$ by

$$q(v_0, v_1, \ldots, v_{k-1}) = v_0 + 2v_1 + \cdots + k \cdot v_{k-1}.$$

Color $1, \ldots, N$ with $c = C_k(N)$ colors, avoiding monochromatic length-k arithmetic progressions. Color each $\overline{v} \in S_n$ with the color of $q(\overline{v})$. We prove by contradiction that this coloring of S_n contains no monochromatic forbidden k-patterns.

Assume $\overline{v}_0, \ldots, \overline{v}_{k-1}$ is a monochromatic forbidden k-pattern. By definition there is a $\overline{w} \in H_n$ such that

$$\overline{v}_i = \overline{w} + \lambda_i\overline{e}_i,$$

for some $\lambda_0, \ldots, \lambda_{k-1}$. Since each $\overline{v}_i$ is on S_n, it follows that $\lambda_0 = \lambda_1 = \cdots = \lambda_{k-1}$. Consider the points

$$q(\overline{v}_0),\ q(\overline{v}_1),\ \ldots,\ q(\overline{v}_{k-1}).$$

The map q is linear, so $q(\overline{v}_i) = q(\overline{w}) + \lambda_0 q(\overline{e}_i)$. By definition of q,

$$q(\overline{e}_i) = i + 1;$$

hence

$$q(\overline{v}_i) = a + (i+1)b,$$

where $a = q(\overline{w})$ and $b = \lambda_0$. But then there is a monochromatic arithmetic progression of length k, which is a contradiction. ∎

Determining the true rate of growth of $C_k(N)$ is a difficult open problem. It is known that C_k is unbounded for fixed k, however, for $k \geq 4$ it can only be shown to grow extremely slowly. In the remainder of this section we look at the specific case of $k = 3$ by relating $C_k(N)$ to another Ramsey-like function.

DEFINITION. Let $R_k(N)$ be the size of a largest subset $A \subset \{1, \ldots, N\}$ that contains no length-k arithmetic progression.

THEOREM 4.3. *For some constant $a > 0$,*

$$\tfrac{N}{R_k(N)} \leq C_k(N) \leq \tfrac{aN \lg N}{R_k(N)}.$$

Proof. Suppose $\{1, \ldots, N\}$ can be c-colored in such a way that there are no length-k arithmetic progressions. Then some color class must be used at least N/c times. Therefore, $R_k(N) \geq N/c$. The first inequality holds since $c = C_k(N)$.

Now, assume that A is a subset of $\{1, \ldots, N\}$ that contains no arithmetic progressions of length k. We demonstrate that the second inequality holds by proving that it is possible to c-color $\{1, \ldots, N\}$ so that it has no length-k arithmetic progressions provided $c \leq aN \lg N/|A|$. This follows directly from:

LEMMA 4.4. Let $A \subset \{1, \ldots, N\}$. No more than $O(N \lg N/|A|)$ translates of A are needed to completely cover $\{1, \ldots, N\}$.

Proof of Lemma: We use a probabilistic existence argument. Choose random translations $t_1, \ldots, t_\ell$, with each t_i in the interval $-N$ to N. The probability that a value $x \in \{1, \ldots, N\}$ is not covered by a particular translation is at most $1 - |A|/2N$. Therefore, the expected number of missed values is bounded above by

$$N \cdot \left(1 - \tfrac{|A|}{2N}\right)^{\ell}.$$

For large enough a, choosing $\ell = aN \lg N/|A|$ makes this less than 1 and the Lemma is proved. ∎

THEOREM 4.5. *(Roth [R]) $R_3(n) > ne^{-c\sqrt{\log n}}$.*

COROLLARY 4.6. Three players can determine Exactly-n in $O(\sqrt{\log n})$ bits.

Proof. From the above,

$$\chi_3(n) \leq C_3(3n) \leq aN \lg N/R_3(3n) \leq a \lg n e^{c\sqrt{\log n}}.$$

Therefore, $\log \chi_3(n)$, the number of bits that have to be communicated, is $O(\sqrt{\log n})$. ∎

5. Application to Branching Programs.

The lower bounds for multi-party protocols give lower bounds on the space and time of branching programs. Branching programs play an important role in the study of time and space tradeoffs and were initially described by Tompa [T] and have been studied by Pippenger [P] and Borodin, Fischer, Kirkpatrick, Lynch, and Tompa [B]. A lower bound for branching programs of width-2 appears elsewhere in these proceedings [BD].

DEFINITION. A *branching program* on the bits $b_0, \ldots, b_{n-1}$ is a rooted, directed, acyclic graph whose internal nodes are labeled with queries of the form "bit b_i?", whose edges are labeled with possible query answers "$b_i = 0$", or "$b_i = 1$", and whose leaves are labeled either "accept", or "reject". For a given assignment of zeroes and ones to the b_i, a branching program either accepts or rejects as the path defined by the bit values leads to an accept or reject leaf. *Time* for a branching program is the maximum path length from the root to a leaf. *Space* for a branching program is the maximum, over all d, of the number of vertices at distance d from the root.

The MPP model described in the first few sections allows us to deal with processes that share information. In the most general situation we can think of processes as having access to a proper subset of a set of bits $b_0, \ldots,$ b_{n-1}. The next lemma shows that the integer-sharing model is also appropriate here.

LEMMA 5.1. Let $P_0, \ldots, P_{k-1}$ be processes. Let P_i have access to subset s_i of the bits $\{b_0, \ldots, b_{n-1}\}$. If $|s_i| \leq \alpha \cdot n$, for some $\alpha < 1$, *i.e.*, if each process is missing at least a constant fraction of the bits, then there exists a constant β, and disjoint subsets $r_0, \ldots,$ r_{k-1} of the b_i such that,

(i) $s_i \cap r_i = \emptyset$, *i.e.*, process P_i doesn't have access to bits in subset r_i, and

(ii) $|r_i| \geq \beta \cdot n$.

Proof. An iterated application of Hall's matching Theorem gives this result directly. ∎

This means that if the bits outside the union of the r_i are held constant, the P_i share integers whose values are the number of bits "on" in the r_j.

THEOREM 5.2. *Any constant-space branching program requires super-linear length to compute Exactly-n.*

Proof. Suppose not. Then, for some integers c, d there are branching programs of width d and length $c \cdot n$ that accept Exactly-n. Think of a branching program of length $c \cdot n$ as being composed of $2cd$ equally-long segments. Each segment can be thought of as a process, P_i, that has access to no more than $n/2$ bits. The whole branching program can thus be viewed as $2cd$ processes, each missing $1/2$ the bits, communicating a constant amount of information ($2cd \log d$ bits) and determining Exactly-n. Using Lemma 5.1 we can show that this contradicts Theorem 3.3. ∎

COROLLARY 5.3. Any constant-space branching program for Threshold requires greater than linear time.

6. Acknowledgments. The authors wish to thank Lyle McGeoch and Larry Stockmeyer for many helpful suggestions. Special thanks to Bill Scherlis for last minute help with the typesetting. The IBM Thomas Watson Research Center at Yorktown heights provided the creative atmosphere in which this work was begun. This work was partially funded by DARPA contracts N00014082-K-0549 and F33615-81-K-1539.

References

[B] A. Borodin, M. Fischer, D. Kirkpatrick, N. Lynch, M. Tompa, "A time-space tradeoff for sorting and related non-oblivious computations." Toronto, Dept. of Computer Science, Technical Report 79-01-01, 1979.

[BD] A. Borodin, D. Dolev, F. Fich, W. Paul, "Bounds for width-2 branching programs." These proceedings.

[EG] P. Erdös and R. Graham, **Old and New Problems and Results in Combinatorial Number Theory.** L'Enseignement Mathématique, Université de Genève, 1980.

[FSS] M. Furst, J. Saxe, M. Sipser, "Parity, circuits and the polynomial-time hierarchy." 22nd *Symposium on the Foundations of Computer Science,* 1981, pp. 260-270; To appear in *Mathematical Systems Theory.*

[G] R. Graham, **Rudiments of Ramsey theory.** Regional Conference Series in Mathematics, number 45, 1981.

[GRS] R. Graham, B. Rothschild, J. Spencer, **Ramsey Theory.** Wiley-Interscience, 1980, p. 38.

[P] N. Pippenger, "A time-space trade-off." *Journal of the ACM,* 25, 1978, pp. 509-515.

[R] K. Roth, "On certain sets of integers." *J. London Math. Soc.,* 29, 1954, pp. 20-26.

[SP] M. Sipser, C. Papadimitriou, "Communication complexity." 14th *Annual ACM Symposium on Theory of Computing,* 1982, pp. 196-200.

[T] M. Tompa, "Time-space tradeoffs for computing functions, using connectivity properties of their circuits." 10th *Annual ACM Symposium on Theory of Computing,* 1978, pp. 196-204.

[Y] A. Yao, "Some complexity questions related to distributed computing." 11th *Annual ACM Symposium on Theory of Computing,* 1979, pp. 209-213.