Towards a Theory of Multiparty Information Complexity

Sam Hopkins

Supervised by: Paul Beame

A senior thesis submitted in partial fulfillment of
the requirements for the degree of

Bachelor of Science
With Departmental Honors

Computer Science & Engineering

University of Washington

June 2013

Presentation of work given on _____

Thesis and presentation approved by _____

Date _____

Abstract

Communication complexity seeks to characterize the complexity of computation by measuring the amount of communication that must be used by players computing the output of some function f, each of whom have only partial information about the inputs to f. The past thirty years have seen the development of a number of lower-bound techniques in communication complexity, which fall roughly into two categories: rectangle-based techniques, which rely on combinatorial properties of the lookup table for f, and information-based techniques, which rely on bounds on the entropy in a communication protocol.

Recently, [JK, BW12, KLL$^+$12] have drawn a close-to-complete picture of the relationships among the various lower-bound techniques in the two-party communication model, concluding that information-based techniques are at least as good as almost all known rectangle-based techniques. Meanwhile, in the multiparty model, where, thus far, information bounds have not been used, it seems that new methods are required to prove nontrivial lower bounds when the number of players is large (logarithmic in the input size). Such bounds are important for one of the key applications of communication complexity—circuit lower bounds. Furthermore, it is conjectured that current techniques are insufficient to tightly characterize the complexity of some functions important both within communication complexity and for various applications, in particular the set disjointness function.

In this thesis, we will investigate the possibility of extending information-based techniques to the case of more than two players. We prove the first nontrivial lower bounds on the information in a k-player protocol for $k > 2$, introducing a new sensitivity-based method to lower-bound information. We discuss progress towards extending our technique to randomized multiparty communication models.

Acknowledgements

There are innumerable people who deserve hearty thanks here and I can only hope to remember a small fraction of them, but nonetheless I must make an attempt.

Thanks first and foremost to my family, without whose love, support, and friendship I could never have gotten to this point. Thanks to Jude, Dessy, and especially Jack—college (and life) wouldn't be the same without y'all.

Thanks to Paul Beame, who has of course provided invaluable help and guidance as I went from communication neophyte to proving my own results, but deserves particular thanking for his uncanny ability to give just the right amount of assistance—I could not have asked for a better advisor.

Thanks to Eric Allender, who introduced me to research in mathematics and therefore was was instrumental in setting me on my present course.

They say it takes a village to raise a child and surely the same is true of a mathematician. Of the many excellent teachers and mentors I've had, a few stand out (although I'm sure I have forgotten many more who are equally deserving). In no particular order, thanks to: Dan Grossman, Siggi Cherem, Vijay Menon, Bob Dumas, Jim Morrow, Larry BonJour, Nancy Sisko, Maren Halvorsen, Bennett Barr, David Evans, Brenda Ajbour, Quinn Thomsen, and David Pippin.

Finally, I have had the great pleasure to be surrounded by a number of excellent friends and colleagues without whom the intense but convivial atmosphere which has characterized my college experience would not have been possible. Thanks in particular to Reid Dale, Jerry Li, Nathan Weizenbaum, Jonathan Ettel, Matt Junge, and Steve Rutherford.

Contents

Chapter 1

Introduction and Background

1.1 Introduction

Communication complexity seeks to characterize the difficulty of computation by measuring the number of messages that must be passed between parties jointly computing the output of some function, each party holding only partial information about the input. First introduced by Andrew Yao in [Yao79], the communication model has been successful in two respects: (i) thirty years of work has produced a wealth of interesting upper and lower bounds on the complexity of a variety of communication problems and (ii) those bounds often transfer readily to other computation models. In addition to applications where communication is obviously relevant (distributed computing, for example), results have been transfered to space bounds for data stream algorithms, circuit size and depth lower bounds, time-space tradeoffs for Turing machines, area-time tradeoffs for VLSI, lower bounds for algebraic query complexity (used in [AW08] to prove a striking new barrier result), and communication lower bounds for combinatorial auction algorithms, among others (see [KN97] for details).

We will study a number of variations on the basic communication model. In particular, we distinguish between two-party and multiparty models. In two-party models, two players, Alice and Bob, hold inputs x and y respectively to a function f and must communicate to jointly compute $f(x, y)$. In multiparty models, k players $1, \ldots, k$ each hold some fraction of the input to a function whose output they will jointly compute. In particular, we will be concerned with *number-on-forehead* (NOF) multiparty models, which are characterized by information overlap between players: player i has an input x_i on her forehead, so that player i sees all inputs *except* x_i.

Past work has produced a variety of general-purpose techniques for proving communication lower bounds in both two-party and multiparty models. These techniques fall roughly into two categories: *rectangle-based* and *information-based*. A recent series of results relate rectangle-based lower-bound techniques to each other [JK] and the best rectangle-based techniques to information-based ones in the two-party setting [BW12, KLL$^+$12]. In this thesis, we will survey this recent work, with an eye towards adapting it to multiparty models, and prove the first nontrivial multiparty information lower bounds.

This work was completed while the author was an undergraduate student in mathematics and computer science at the University of Washington under the advisement of Paul Beame.

1.1.1 Intended Audience and Other Resources

Most of this thesis should be comprehensible to senior undergraduates with standard courses in computational complexity theory and discrete probability and (that magical catch-all) appropriate mathematical maturity. Basic background in complexity can be found in [Sip]; a more modern but somewhat more difficult resource is [AB09]. We will provide some motivation for the results we present here, but the breadth of applicability of communication complexity makes drawing a complete picture in this thesis impossible. The reader is encouraged to consult [AB09, KN97] and the many expository works on communication complexity available on the Internet for further discussion of applications and references to the research literature. In particular, see [LS] for a thorough discussion of many of the lower-bound techniques we are only able to mention here in passing, and [CP] for background on set disjointness, the most important explicit function in the communication model.

1.2 Background

1.2.1 Notation

The notation $f : S \to T$ denotes a function from a set S to a set T. Given a set S, we use S^n to denote n-tuples of elements in S. We let $S \times T = \{(s,t) : s \in S, t \in T\}$.

We will generally employ capital letters $X, Y, Z, \ldots$ for random variables and lower-case letters $x, y, z, \ldots$ for values of those variables. We use standard notation for conditioning, expectation, etc. All random variables considered in this thesis are discrete.

Very often we will concern ourselves with vectors in $\mathbb{F}_2^n$ (where $\mathbb{F}_2$ denotes the field with two elements, as usual). Usually we will write this interchangeably with $\{0,1\}^n$, except when we occasionally find it convenient to use $\{-1,1\}^n$ instead. Which we intend is usually clear from context. For a vector $x \in \{0,1\}^n$, we write x_i for the i-th component of x and x_{-i} for every component of x but the i-th (so the i-th slot is "empty"). We write x^j for $x + e_j$, where e_j is the j-th standard basis vector. (So x^j is x with the j-th bit flipped.) $\mathbf{0}$ and $\mathbf{1}$ denote the all-0's vector and the all-1's vector, respectively.

For a function p, we write $\operatorname{supp} p$ for the set of inputs on which p is nonzero.

We will have occasion to employ the discrete *Hellinger distance*, a metric on the space of distributions on a fixed probability space. For random variables X, Y distributed according to ν_X and ν_Y, respectively, the Hellinger distance between X and Y, denoted interchangeably by $h(X,Y)$ or $h(\nu_X, \nu_Y)$, is given by

$$\frac{1}{\sqrt{2}} \sqrt{\sum_{x \in \operatorname{supp} X \cup \operatorname{supp} Y} \left(\sqrt{\nu_X(x)} - \sqrt{\nu_Y(x)}\right)^2}.$$

The Hellinger distance is a metric, so the triangle inequality applies: for distributions ν_0, ν_1, ν_2,

$$h(\nu_0, \nu_1) + h(\nu_1, \nu_2) \geq h(\nu_0, \nu_2).$$

1.2.2 The Communication Model

In this section we survey the basic two-party and multiparty communication models with various forms of randomness. Readers familiar with e.g. [KN97] can skip this section without consequence. For more examples, exercises, and fuller proofs, see [KN97].

Deterministic Two-Party Communication

The most basic model is as follows. Two players, Alice and Bob, wish to compute the output of some function $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$. Each has unlimited computational power (for example, either may compute the answer to the Halting problem if they so desire). Alice receives an input $x \in \mathcal{X}$ and Bob receives $y \in \mathcal{Y}$. They will compute according to a *protocol* Π which tells them at each step of the computation what message to send as a function of their input and previous messages sent. At the end of the protocol, each player must know the value $f(x, y)$.

Definition 1.2.1. A two-player deterministic communication protocol is a binary tree where every internal node v is labeled either by a function $a_v : \mathcal{X} \to \{0, 1\}$ or a function $b_v : \mathcal{Y} \to \{0, 1\}$, every internal node has precisely two children, with one outgoing edge labeled 1 and the other 0, and every leaf node is labeled with an output in $\mathcal{Z}$.

To execute a protocol, players walk down the tree, starting at the root and ending at a leaf. At each step, the owner of the current node v (Alice if the label is a_v, Bob if b_v) evaluates the function at v on their input and sends the resulting value as a message. The players then walk down the edge incident to the current node labeled with that message. At the end of the execution the players output the label of the terminating leaf node.

Definition 1.2.2. The transcript of the execution of a protocol Π on an input (x, y), is the sequence of edge-labels in the players' tree traversal. We write $\Pi(x, y)$. We assume that the output of the protocol is the last bit of the transcript. When we mean the output of Π run on (x, y), we write $\operatorname{out}\Pi(x, y)$.

Definition 1.2.3. The communication cost $CC(\Pi)$ of a protocol Π is the length of the longest of any $\Pi(x, y)$ over all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Equivalently, $CC(\Pi)$ is the height of the protocol tree for Π.

Note that we are interested here only in the worst-case complexity (i.e. transcript length) of the protocol.

If for some function $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ and for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ it is the case that $\operatorname{out}\Pi(x, y) = f(x, y)$, we say that Π computes f with zero error (later we will allow protocols to err on some inputs).

Definition 1.2.4. The deterministic zero-error communication complexity of f, denoted $D(f)$, is the minimum of $CC(\Pi)$ over all Π computing f with zero error.

Usually (in complexity generally) we are not interested in a function $f : \{0, 1\}^n \to \{0, 1\}$ for fixed n; rather n is a parameter in both the definition of f and the complexity of f. So we think of $D(f)$ as a function of n.

Note that every function f admits a trivial protocol in which Alice sends Bob her input x, Bob computes $f(x, y)$, and sends it back to Alice. Thus we have

Theorem 1.2.5. *For all $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$, $D(f) \leq \log|\mathcal{X}| + \log|\mathcal{Z}|$.*

If $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, this becomes $D(f) \leq n+1$. Because of this, we think of efficient computation in the communication model as requiring $O(\log n)$ or perhaps $\log^{O(1)}(n)$ communication. In order for the model to be of interest, there must actually be some efficiently-computable functions. The following examples give two such functions

Example 1.2.6 (taken from [KN97])**.** Suppose Alice and Bob have as input $x \subseteq [n]$ and $y \subseteq [n]$, respectively, and they wish to compute $\max(x \cup y)$. Alice computes the maximum m over x and sends it to Bob (we model this as all nodes of depth at most $\log n$ in the protocol tree belonging to Alice). Bob then computes the maximum over $\{m\} \cup y$ and sends it back to Alice, again requiring $\log n$ bits of communication. Thus the depth of the protocol tree Π corresponding to this protocol is $2\log n$, and so $D(\max) \leq 2\log n$.

Example 1.2.7 (taken from [KN97])**.** Alice and Bob again have inputs $x, y \subseteq [n]$, and they wish to compute median$(x \cup y)$, where we now think of $x \cup y$ as a multiset. We will give a communication protocol using binary search. In order to run a binary search, Alice and Bob must be able to decide whether the median is above or below some number m. To do so, they can use the following procedure ABOVE-OR-BELOW(m):

1. Alice sends the number of elements in her input below m.
2. Bob sends 0 if $m <$ median$(x \cup y)$ and 1 otherwise.

Clearly ABOVE-OR-BELOW(m) has communication cost $O(\log n)$, and it must be run $\log n$ times, so $D(\text{median}) \in O(\log^2(n))$.

Functions of Note

In this section we give a brief tour of some noteworthy and natural functions whose communication complexity is of interest. See [KN97] for details.

EQ The equality function, $\text{EQ} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is given by $\text{EQ}(x,y) = 1$ if $x = y$ and 0 otherwise.

IP The inner product $\text{IP} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is given by

$$\text{IP}(x,y) = \sum x_i \wedge y_i \mod 2.$$

GIP The generalized inner product $\text{GIP} : \{0,1\}^k \to \{0,1\}$ is given by

$$\text{GIP}(x_1, \ldots, x_n) = \sum_{i \leq n} \bigwedge_{j \leq k} x_{ij} \mod 2.$$

GIP counts the parity of the number of all-1 rows in the input matrix (where the columns are $x_i \in \{0,1\}^k$). We also write $\text{GIP} = XOR \circ AND$.

DISJ We define the disjointness problem DISJ as follows in both the two-party and multiparty context. Consider the input vectors $x_i \in \{0,1\}^n$ to be characteristic vectors of subsets of $[n]$. Then $\mathrm{DISJ}(x) = 0$ if $\bigcap x_i = \varnothing$. and 1 otherwise. We often find it useful to express DISJ as $\mathrm{DISJ} = OR \circ AND$.

Applications of Communication Complexity

As a natural model of computation, two-player communication is of some inherent interest. Motivation to study it is also provided by a wide range of applications to other computation models. It turns out to be fairly natural to identify within a computation a need for information to flow between distinct parts of a model, which induces a reduction to communication complexity. Two-player communication has applications in streaming algorithms, combinatorial auctions, data structures, and time-space tradeoffs, among other places. As an example of such a reduction, we describe a connection between two-player communication and area-time tradeoffs for very-large-scale integrated circuits (VLSI). (Another such connection, which we describe in brief below, gives a key motivation to study lower-bound techniques in the multiparty model.)

The VLSI model, which is closely-related to the real-world problem of laying out transisters and wires in silicon, is as roughly as follows. We wish to lay out on a chip a circuit computing some function f of n inputs and n outputs. We are concerned with the area occupied by the resulting circuit and the amount of time the circuit takes to compute f. Available to us are processing elemements which each take some number of inputs d and output d bits, which compute some d-ary function and are assumed to be of area proportional to d^2, and wires, which connect inputs and outputs of processing elements. Wires are laid out in a two-dimensional grid with unit spacing and transmit one bit per unit length per unit time. Wires may bend and cross each other at grid points but may only connect at via processing elements. Each unit of wire cost a unit of area.

Input and output occurs at processing elements: n of the processing elements are designated as inputs and n (not necessarily distinct) processing elements are designated as outputs.

To a VLSI layout there is naturally an associated graph where nodes are processing elements and edges are wires. That graph has some minimum bisection width (that is, the minimum size cut so that exactly half the nodes lie on each side of the cut). The following geometric theorem first appears in [Tho79].

Theorem 1.2.8. *The area A of a VLSI circuit is at least $w^2/4$, where w is the minimum bisection width of the associated graph.*

Observe that a bisection of the circuit induces a communication problem: Alice gets the inputs on side $\mathcal{A}$ of the cut and must produce the outputs on side $\mathcal{A}$, and, respectively, Bob gets inputs on side $\mathcal{B}$ of the cut and must produce the outputs on side $\mathcal{B}$. For some fixed function f, suppose we can prove a communication lower bound of C across all the communication problems induced by all possible bisections of the inputs and outputs. Then the time T required by the circuit is at least C/w. Together with

Theorem 1.2.9. *Consider a VLSI circuit with area cost A and time cost T, which computes a function f. Suppose that C is a lower bound on the communication complexity of*

all possible communication problems induced by partitioning the inputs and outputs of f. Then

$$AT^2 \geq \frac{C^2}{4}.$$

Example uses of theorem 1.2.9 are an area-time tradeoff of $AT^2 \geq N^2/16$ for the discrete Fourier transform of a length-N vector in [Tho79] and a tradeoff of $AT^2 \geq n^2/64$ for the product of two n-bit integers in [AA80].

Multiparty Communication

In the multiparty setting, we have k players $p_1, \ldots, p_k$. Writing $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_k$, the players wish to compute the output of some $f : \mathcal{X} \to \mathcal{Z}$. Again, players have unlimited computational power. Instead of passing messages to each other, they write the messages on a shared blackboard, so that all players can see all messages.

There are two models of interest. One, the *number-in-hand* model, is the most straightforward generalization from two parties—player p_i receives the input $x_i \in \mathcal{X}_i$. We will be concerned almost entirely with the *number-on-forehead* model, where player p_i places input $x_i \in \mathcal{X}_i$ on his forehead, so that p_i can see all inputs $x_{-i} \in \mathcal{X}_{-i}$ and not x_i.

Definition 1.2.10. A k-party (or k-way) deterministic multiparty number-on-forehead protocol Π is a binary tree where each node is labeled by a function $f_i : \mathcal{X}_{-i} \to \{0, 1\}$ for some $1 \leq i \leq k$. Protocols are executed exactly as in the two-party case.

Applications and Barriers for Multiparty Communication

In addition to being a natural model of communication with overlapping information, NOF multiparty communication has wide-ranging applications. Probably the most important of these is that proving that some function requires super-polylogarithmic communication for super-polylogarithmically-many players would separate NP and ACC^0, the class of poly-size constant-depth boolean circuits with unbounded fan-in and MOD_m gates. It would suffice to prove such a bound for a very restricted class of protocols: *simultaneous* protocols, where players do not interact at all, instead each sending one message to an external referee who computes the output.

However, as we will see, current techniques seem unable to prove a nontrivial lower bound for even $k = \omega(\log n)$ players (the so-called $\log(n)$ barrier), even for simultaneous protocols. This brings us to the purpose of this thesis: to seek new lower bound techniques for multiparty communication.

The seeming insurmountability of the $\log n$ barrier is not the only reason to believe that new lower-bound methods are needed in multiparty communication complexity. The set disjointness function, DISJ, has received a great deal of attention in both the two-party and multiparty models: it is a complete problem for the two-player communication complexity analogue of NP, and many of the applications of communication complexity to other models go via disjointness.

In the two-player case we know $CC(\mathrm{DISJ}) = \Theta(n)$. Multiparty set disjointness is less well understood. The best known upper bound is $\tilde{O}(n/2^k)$. A recent series of papers

by Sherstov and others, culminating in [She13], improves the best known lower bound to $\Omega(\sqrt{n}/2^k k)$. However, the techniques used for this lower bound also apply to quantum communication complexity, where the best known upper bound is $\tilde{O}(\sqrt{n/2^k})$. Thus, existing techniques cannot improve the classical bound to $\Omega(n)$. Indeed, even in the simultaneous model it is not known whether DISJ $= \Omega(n^\alpha)$ for any $\alpha > \frac{1}{2}$. Any new lower bound method in the multiparty case would be of interest, even if it only applied to a weakened computation model.

Randomized Protocols

An understanding of the communication model (or indeed any model of computation) is not complete without understanding its behavior when randomness is introduced.

There are several ways we might introduce randomness to the communication model. Intuitively, we think of players being able to flip evenly-weighted coins whenever they like and allow their computations to branch on the outcomes of those flips. The question is, who sees the coins?

In the two-party case we only have two options, fully-public and fully-private randomness. A protocol in which Alice and Bob can both see all coins is *public-coin*; one in which both have their own coins not visible to the other we call *private-coin*. In the multiparty case we can have fully-public coins and fully-private coins; we might also have coins seen by some other number of players—here we will have occasion to discuss protocols in which for each player i there is a source of random coins visible to p_{-i}, all players except i. We call these *coins on the forehead*.

For convenience, our formalism will model public-coin and private-coin protocols somewhat differently.

Definition 1.2.11. A public-coin (two-party, multiparty) protocol Π is a distribution over some set of deterministic protocols Π_r.

Definition 1.2.12. A private-coin (two-party, multiparty) protocol Π is a (two-party, multiparty) protocol in which each tree node v is labeled with functions that have as input the inputs visible to the player owning v and a random string r_i sampled from a distribution ν_i specific to player i.

Definition 1.2.13. A multiparty protocol with coins on the forehead is a protocol in which nodes belonging to player i are labeled with functions that have, in addition to the usual inputs, a (partial) tuple r_{-i} of random strings drawn from a distribution ν_{-i}.

We will often have occasion to work with protocols that have coins at multiple levels of privacy; we leave it as an exercise to the reader to modify the definitions accordingly.

Protocols with Nonzero Error

Once we introduce randomness, we need not require that protocols compute functions exactly. We say that a randomized protocol Π computes f with ϵ error if for every input (x, y), or input vector x, in the multiparty case,

$$\Pr[\Pi(x, y) = f(x, y)] \geq 1 - \epsilon.$$

We write $R_\epsilon^{pub}(f)$ for the communication cost of the least-cost protocol computing f with error ϵ and public-coin randomness. For private-coin randomness we simply write $R_\epsilon(f)$, and for NOF randomness we write $R_\epsilon^{NOF}(f)$.

1.2.3 Combinatorial Lower Bounds on Communication

Rectangles and Cylinder Intersections

We begin with the two-player case. Now that we have formulated the model quite generally, we will drop the $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ notation and assume we are dealing with functions $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$. In general this reduces our technical burden but does not fundamentally change any of the results, so we leave the most general formulations to the reader.

To $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ we associate a matrix M_f with dimensions $2^n \times 2^n$, whose rows and columns are indexed by elements of $\{0,1\}^n$, so that $M_f[x,y] = f(x,y)$. (Thus M_f is a sort of function table for f.)

Definition 1.2.14. A rectangle in M_f is a subset S of the entries of M_f so that $S = A \times B$, where $A, B \subseteq \{0,1\}^n$. A rectangle $R \subseteq M_f$ is f-monochromatic (or just monochromatic if f is clear from context) if there is $b \in \{0,1\}$ so that for all $(x,y) \in R$ we have $f(x,y) = b$.

Theorem 1.2.15. *A deterministic protocol Π with complexity c computing f with zero error partitions M_f into at most 2^c f-monochromatic rectangles.*

See [KN97] for a proof. Theorem 1.2.15 provides us with a way to lower-bound the complexity of a protocol computing a function f with zero error—lower bound size of any partition of M_f into f-monochromatic rectangles.

Before we see how to produce such a bound, however, let us see the appropriate multiparty generalization. Let $f : \{0,1\}^{nk} \to \{0,1\}$. We associate to f a k-dimensional tensor T_f of size n so that $T_f[x_1] \dots [x_k] = f(x_1, \dots, x_k)$.

Definition 1.2.16. An i-cylinder in T_f is a subset $S \subseteq T_f$ so that membership in S does not depend on the i-th coordinate; that is, for any $x, x' \in \{0,1\}^n$, if $x_1, \dots, x_{i-1}, x, x_{i+1}, \dots, x_k \in S$ then $x_1, \dots, x_{i-1}, x', \dots, x_k \in S$.

Definition 1.2.17. A cylinder intersection in T_f is a subset $S \subset T_f$ so that $S = \bigcap S_i$ for some family $\{S_i\}$ of cylinder intersections.

The following theorem is the direct multiparty analogue of theorem 1.2.15:

Theorem 1.2.18. *A deterministic k-player protocol Π with complexity c computing f with zero error partitions T_f into at most 2^c f-monochromatic cylinder intersections.*

As before, theorem 1.2.18 provides us with a way to lower-bound the complexity of a protocol computing f with zero error.

Distributional Complexity

In order to use this rectangle/cylinder-intersection paradigm to bound randomized complexity, we need a way to bound randomized complexity by working strictly with deterministic protocols. The following result provides just that.

It says roughly that the complexity of computing a function with an ϵ-error randomized protocol is the same as the complexity of computing a function with a deterministic protocol that can be wrong on an ϵ-fraction of the inputs.

For a distribution ν on the inputs $\mathcal{X} \times \mathcal{Y}$ to a function f, let $D^{\nu}_{\epsilon}(f)$ be the communication cost of the least-cost deterministic protocol Π so that $\Pr_{\nu}[\Pi(x,y) = f(x,y)] \geq 1 - \epsilon$. The measure $D^{\nu}_{\epsilon}(f)$ is the *distributional complexity* of f. We write $D_{\epsilon}(f)$ for the maximum of $D^{\nu}_{\epsilon}(f)$ over all ν.

Lemma 1.2.19 (Yao's Lemma)**.**

$$R^{pub}_{\epsilon}(f) = \max_{\nu} D^{\nu}_{\epsilon}(f).$$

The proof of lemma 1.2.19 is easy in the direction we care about ($R^{pub}_{\epsilon}(f) \geq \max_{\nu} D^{\nu}_{\epsilon}(f)$) by a counting argument. The other direction employs the Von Neumann minimax theorem. See [KN97] for a full proof.

Discrepancy

With a little additional cleverness, theorems 1.2.15 and 1.2.18 give us a technique (actually, several) to lower-bound $D^{\nu}_{\epsilon}(f)$. We present the two-party version; the multiparty version is entirely analogous.

Suppose that some (deterministic) protocol Π computes f with low error under ν. Since Π partitions M_f into 2^c rectangles so that on each rectangle its output is constant, there must be such a partitioning in which all rectangles are either small (i.e. low-weight under ν) or almost-monochromatic, so that they do not introduce too much error. This leads to the method of *discrepancy*:

Definition 1.2.20. Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, let ν be a distribution on M_f. Let R be a rectangle in M_f. Then

$$\text{Disc}_{\nu}(f, R) = |\nu(R \cap f^{-1}(1)) - \nu(R \cap f^{-1}(0))|$$

and

$$\text{Disc}_{\nu}(f) = \max\{\text{Disc}_{\nu}(f, R) : R \text{ a rectangle in } M_f\}$$

and finally

$$\text{Disc}(f) = \min\{\text{Disc}_{\nu}(f) : \nu \text{ a distribution on } M_f\}.$$

Replacing rectangles with cylinder intersections and M_f with T_f gives the multiparty version of discrepancy.

The following theorem captures our intuitive discussion above. For a proof see [KN97].

Theorem 1.2.21.

$$D_{\epsilon}(f) \geq \log \frac{1 - 2\epsilon}{\text{Disc}(f)}.$$

Theorem 1.2.21 allows us to prove linear lower bounds on communication for functions with exponentially-small discrepancy. In the two-player case, discrepancy can often be bounded correctly. (However, not all functions for which we want to prove linear lower bounds have exponentially-small discrepancy. DISJ is one such. A better rectangle-based technique, *corruption*, gets around the too-large discrepancy of DISJ. See e.g. [CP] for a thorough discussion.)

The BNS-Chung Criterion for Multiparty Discrepancy

Discrepancy is more difficult to bound in the multiparty model. All known uses of discrepancy go via the following theorem, which bounds discrepancy in terms of a much more easily computed quantity. We give a heavily condensed version of the presentation in [Raz00], to which the reader is referred for details and proofs.

Fix a k-way function f with inputs in $\mathcal{X}_1 \times \ldots \times \mathcal{X}_k$ with $\mathcal{X}_i = \{0,1\}^n$ and outputs in $\{-1,1\}$. A cube D is a multi-set $\{a_1, b_1\} \times \ldots \{a_k, b_k\}$ where $a_i, b_i \in X_i$. Define the sign of the cube D with respect to f as

$$S_f(D) = \prod_{d_1 \in \{a_1, b_1\}} \cdots \prod_{d_k \in \{a_k, b_k\}} f(d_1, \ldots, d_k).$$

Let

$$\mathcal{E}(f) = \mathbb{E}_D[S_f(D)]$$

where D is sampled uniformly at random from all possible cubes. The following is the key bound.

Theorem 1.2.22.

$$\mathcal{E}(f) \geq \mathrm{Disc}(f)^{2^k}.$$

The 2^k in the exponent arises from k uses of Cauchy-Schwarz (one for each layer of the product in the definition of $S_f(D)$). It is also the reason that current techniques are unable to break the $\log n$ barrier: setting $k = \log n$ we see that the resulting communication bound degrades badly unless $\mathcal{E}(f)$ is very small indeed.

1.2.4 Information Complexity

A Crash Course in Information Theory

This section is a light-speed tour of information theory. See e.g. [CT06] for intuition and full proofs.

In what follows, let X, Y, Z be random variables and p_X, p_Y, p_Z be the corresponding probability mass functions (and $p_{X,Y}, p_{X,Z}$, etc., be the probability mass functions for the joint distributions $(X,Y), (X,Z)$, etc.) We will occasionally drop the subscripts when they are clear from context.

Definition 1.2.23. The entropy of X, denoted $H(X)$ is

$$H(X) = \sum_{x \in \operatorname{supp} p_X} p_X(x) \log p_X(x).$$

Definition 1.2.24. The mutual information between X and Y, denoted $I(X;Y)$, is

$$I(X;Y) = \sum_{(x,y) \in \operatorname{supp} p_{X,Y}} p_{X,Y}(x,y) \log \frac{p_{X,Y}(x,y)}{p_X(x)p_Y(y)}.$$

These quantities have conditional versions. Conditioning on specific events works as follows:

Definition 1.2.25. For $z \in \operatorname{supp} p_Z$,

$$H(X|Z=z) = \sum_{x \in \operatorname{supp} p_X} p_X(x|Z=z) \log p_X(x|Z=z).$$

$$I(X;Y|Z=z) = \sum_{(x,y) \in \operatorname{supp} p_{X,Y}} p_{X,Y}(x,y|Z=z) \log \frac{p_{X,Y}(x,y)}{p_X(x|Z=z)p_Y(y|Z=z)}.$$

Often, we want to condition not on the specific event $Z = z$ but on the value of the random variable Z, whatever it may be.

Definition 1.2.26.

$$H(X|Z) = \mathbb{E}_{z \sim Z}[H(X|Z=z)]$$
$$I(X;Y|Z) = \mathbb{E}_{z \sim Z}[I(X;Y|Z=z)]$$

The following theorem is central to our use of entropy to bound communication complexity.

Theorem 1.2.27. *$H(X) \leq \log |\operatorname{supp} p_X|$. Thus, if X takes values in $\{0,1\}^n$, it has entropy at most n. Conversely, if $H(X) > n$ and we know X takes values in $\{0,1\}^{n'}$ for some n', then $n' > n$.*

Our route to a bound on $H(X)$ goes via a bound on $I(X;Y)$, enabled by the following theorem

Theorem 1.2.28. *$H(X) \geq I(X;Y)$, with equality if and only if X and Y are independent.*

Information cannot increase by computing a function of a random variable. For random variables X, Y and a function g, the *data processing inequality* says

$$I(X;Y) \geq I(X;g(Y)).$$

We state without proof the following result which relates Hellinger distance to mutual information. For a proof, see [Lin91].

Lemma 1.2.29 (Lin's Lemma)**.** *Let X be a random variable which is uniform on $\{0,1\}$, and let Z be any random variable. Then*

$$I(X;Z) \geq h^2(Z|X=0, Z|X=1).$$

Two-Party Information Complexity

In what follows, let Π be a two-party communication protocol whose input (X, Y) is distributed according to ν.

Definition 1.2.30. The information cost of Π is

$$IC_\nu(\Pi) = I(X; \Pi(X,Y)|Y) + I(Y; \Pi(X,Y)|X).$$

Definition 1.2.31. For a function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, a distribution ν on $\{0,1\}^n \times \{0,1\}^n$, and $\epsilon > 0$ we define the ϵ-error information cost of f to be

$$IC^\epsilon_\nu(f) = \min IC_\nu(\Pi)$$

where the minimum is taken over protocols Π computing f with ϵ-error under ν.

This notion is first implicit in [BYJKS04]. The following theorem relates information cost to communication complexity.

Theorem 1.2.32. *Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$. For any ϵ, ν,*

$$D^\epsilon_\nu(f) \geq IC^\epsilon_\nu(f).$$

Proof. Special case of theorem 2.3.3, which we prove independently. □

1.3 Our Contributions

We provide a natural generalization of the definition of information cost to the multiparty case and prove that it is a lower-bound on the communication cost.

Definition 1.3.1. Let Π be a k-party NOF (randomized) communication protocol with inputs $X = X_1 \dots X_k$ sampled from a distribution ν. Then the information cost of Π is

$$IC_\nu(\Pi) = \sum_i I(X_i; \Pi(X)|X_{-i}).$$

Our main new theorem computes the NOF information cost of AND_k, the AND of k single bits. Where $DIC(f)$ and $R^{pub}IC(f)$ are the deterministic and public-coin information cost of f (see definitions in the next chapter), we prove

Theorem 1.3.2.

$$DIC_\mu(AND_k) \geq \frac{k+1}{2^{k-1}}$$
$$R^{pub}IC^0_\mu(AND_k) \geq \frac{k+1}{2^{k-1}}.$$

Note that in both cases there is a trivial bound of $k/2^{k-1}$ simply from considering the mutual information between the output and the inputs. Our main contribution is therefore the additional $1/2^{k-1}$, which is information that much be revealed in the course of computing the AND.

We also prove partial results towards extending theorem 1.3.2 to the case of private randomness. We are unable to prove a general result; instead, we prove a result when the number of messages sent is bounded.

Theorem 1.3.3. *Let Π be a protocol with r rounds exactly computing AND_k with private coins. Then*

$$IC^0_\mu(\Pi) \geq \frac{1}{8(kr)^2 2^{k-1}} + \frac{k}{2^{k-1}}.$$

Chapter 2

Multiparty Information Complexity

2.1 Introduction and Two-Party Background

In the two-party setting, the relationships among the various lower-bound techniques are relatively well-understood. We will not present the whole picture here (in particular we will omit discussions of various known gaps between techniques), but what follows is what is necessary to provide context for our coming discussion of the multiparty setting.

The authors of [JK] exploit linear-programming formulations of all known two-party rectangle-based techniques to characterize relationships among them. In particular, they formulate a new technique, the *partition bound*, expressed as the optimum of a linear program (where the LP instance is dependent on the function f for which we want a lower-bound, as well as an error parameter ϵ). They prove that the partition bound beats all other known combinatorial bounds (discrepancy, as presented in the previous chapter, and several stronger rectangle-based bounds).

Since [BW12], it has been known in the two-party case that that information complexity is at least as good a lower bound as discrepancy (in particular, the discrepancy of a function f provides a lower bound on f's information complexity). In [KLL$^+$12], a modification of the partition bound (the *relaxed partition bound*), which still beats all known combinatorial bounds with the exception of the partition bound itself, is shown to be a lower bound on information complexity. Thus, in the two-party setting, all known lower bound techniques (except for the partition bound, which has never been used to bound an explicit function) are beaten by information complexity.

These results inspire the idea that a multiparty generalization of information complexity could aid in the search for the new lower bound techniques that seem to be required to break the $\log(n)$ barrier and tighten DISJ lower bounds.

In this section, we show that, despite a major pitfall arising if number-on-forehead randomness is not avoided, there is indeed hope for a useful theory of multiparty information complexity. We do this by proving the first nontrivial lower bounds on the information complexity of a multiparty function in several variants of the multiparty communication model.

Before we can understand the importance of the pitfall, however, we need some further two-party background.

2.2 The Direct-Sum Paradigm

In general, a *direct-sum* theorem says if computing f has cost C, then computing n independent copies of f has cost $\Omega(nC)$. Information-theoretic quantities have nice direct sum properties (the entropy of n i.i.d. variables is n times the entropy of one of them).

The direct sum problem, of interest in its own right, is whether computing n independent copies of a function f requires n times the resources to compute one copy. The general direct sum problem for communication complexity is addressed in [BBCR10].

Direct-sum-based techniques can be used to prove bounds on communication problems that are not merely n copies of some simple function. Beginning with the work in [BYJKS04], in the two-player case information-based lower bounds for a function f are proved roughly as follows:

1. Express f as a composition of two simple functions, $f = g \circ h$. For example, $DISJ = OR \circ AND$.
2. Show that h has information cost at least J.
3. Show that a protocol for f can be used as a black box in a protocol to compute h in n independent ways (one for each copy of h being computed).
4. Conclude that f has information cost at least nJ.

Item (3) above is tricky. For example, consider the problem of designing a protocol for AND using a protocol for $DISJ$ as a black box. Alice and Bob receive single bits a, b respectively and need to generate an input (x, y) to $DISJ$ so that $DISJ(x, y) = AND(a, b)$. For a fixed position i, they will set $x_i = a$ and $y_i = b$, and they need to generate x_{-i} and y_{-i} so that there is no $j \neq i$ where $AND(x_j, y_j) = 1$. Naively, this could be done by setting x_j, y_j to 0 for $j \neq i$, but the resulting distribution is not "hard" enough (leaving aside the details) to give the direct sum. There is a good deal of subtlety to the technique employed in [BYJKS04] to generate inputs (x, y) from a sufficiently-hard distribution. We refer the reader there for details—the key point is the generation of the input (x, y).

How might this be extended to the multiparty setting? All players but the i-th must agree on an input place on player i's forehead. Furthermore, to keep everything sufficiently "hard," it would seem that player i must not know that input. The only way we know of to accomplish this is to use coins on player i's forehead to sample those input bits. But this, as we will see, is a big problem. We proceed to define multiparty information complexity, after which point we will be able to discuss why NOF randomness is disastrous for multiparty information.

2.3 Multiparty Information Complexity

Here is the definition we've been waiting for:

Definition 2.3.1 (restatement of 1.3.1)**.** Let Π be a k-party NOF (randomized) communication protocol with inputs $X = X_1 \dots X_k$ sampled from a distribution ν. Then the

information cost of Π is

$$IC_\nu(\Pi) = \sum_i I(X_i; \Pi(X)|X_{-i})$$

Definition 2.3.2. Let $f : \{0,1\}^{nk} \to \{0,1\}$ and let ν be a distribution on $\{0,1\}^{nk}$. We define a few different notions of ϵ-error information complexity for f, depending on where we allow randomness in the protocol. First of all, the deterministic information cost of f is

$$DIC_\nu(f) = \min_\Pi IC_\nu(\Pi)$$

where the minimum is taken over deterministic protocols Π computing f with zero error.

Now let $\epsilon > 0$. The public-randomness ϵ-error information cost of f is

$$R^{pub}IC^\epsilon_\nu(f) = \min_\Pi IC_\nu(\Pi)$$

where the minimum is across protocols Π with public randomness computing f with ϵ-error under ν. The private-randomness ϵ-error information cost of of f is

$$RIC^\epsilon_\nu(f) = \min_\Pi IC_\nu(\Pi)$$

where the minimum is now across protocols with public and private randomness computing f with ϵ-error under ν.

Finally, the NOF-randomness ϵ-error information cost of f is

$$R^{NOF}IC^\epsilon_\nu(f) = \min_\Pi IC_\nu(\Pi)$$

where the minimum is across protocols with public, private, and NOF randomness computing f with ϵ-error under ν.

In the two-party setting, information cost is an absolute lower bound on communication cost. In the multiparty setting we lose a factor of $k-1$ (which of course disappears at $k = 2$) because the same bit of information in the transcript might contribute to shared information with both X_i and $X_{i'}$ for some $i \neq i'$. At the cost of extra technical complication the resulting loss can generally be avoided; a discussion of how to do so follows the next theorem.

Theorem 2.3.3. *Let Π be a k-party protocol and let ν be a distribution on the input to Π. Then*

$$IC_\nu(\Pi) \leq (k-1)CC(\Pi).$$

Proof. We adapt the proof in [BR11]. Let Π_t be the t-th bit of Π. By definition and the chain rule for mutual information,

$$\begin{aligned} IC_\nu(\Pi) &= \sum_i I(X_i; \Pi(X)|X_{-i}) \\ &= \sum_{j=0}^{CC(\Pi)} \sum_{i=0}^{k} I(X_i; \Pi_j(X)|X_{-i}, \Pi_1(X) \dots \Pi_{j-1}(X)) \end{aligned}$$

For $\gamma \in \{0,1\}^{i-1}$, let E_γ be the event that the first $i-1$ bits of $\Pi(X)$ are γ. We have

$$IC_\nu(\Pi) = \sum_{j=0}^{CC(\Pi)} \sum_{i=0}^{k} \mathbb{E}_\gamma[I(X_i; \Pi_j(X)|E_\gamma, X_{-i})].$$

where Π_j is the jth message of Π. Observe that $\sum_{i=0}^{k} I(X_i; \Pi_j(X)|E_\gamma, X_{-i}) \leq k-1$, since each term is at most 1 (since $\Pi_j(X)$ is just one bit), and if γ requires player i' to speak message Π_j then $I(X_{i'}; \Pi_j(X)|E_\gamma, X_{-i'})$ must be 0. Thus, $IC_\nu(\Pi) \leq (k-1)CC(\Pi)$ as desired. □

The proof of this theorem actually shows something slightly more: if for each stage j of a protocol Π we can identify a small set S of players (where we think of $|S|$ as small relative to k) for whom we can lower-bound

$$\sum_{i \in S} I(X_i; \Pi_j(X)|X_{-i}, \Pi_1(X) \dots \Pi_{j-1}(X))$$

then we do not lose the factor $(k-1)$. Most of the bounds that we prove will actually proceed by such a technique, but we will trade the factor of $(k-1)$ for simplicity and not bother to make this explicit.

2.4 Secure Multiparty Computation

We can now discuss why NOF randomness is disastrous for multiparty information: in the NOF-randomness setting, all functions can be computed with zero information.

In [BOGW88], a surprising protocol is given to compute any k-way function in such a fashion that after the protocol is run, no player can compute anything about any other player's input except what is already available by virtue of the output's dependence on the inputs. The computation model in question is, however, crucially different from ours: each pair of players may communicate on an *untappable private channel*. We refer the reader to [BOGW88] for details on the protocol; the key question here is whether the result applies to our multiparty model, which has blackboard communication.

Unfortunately, if NOF randomness is present, the answer is yes.

Theorem 2.4.1. *For all f, ν, ϵ,*

$$R^{NOF}IC_\nu^\epsilon(f) = \sum I(X_i; f(X)|X_{-i}).$$

That is, the information complexity of f is no greater than the information available about each input in the output of f.

Corollary 2.4.2. *If f is a boolean function, then for all ν, ϵ*

$$R^{NOF}IC_\nu^\epsilon(f) \leq k-1.$$

Proof. We assume the existence of a zero-information protocol in the private-channel case and show how to simulate private channels with NOF randomness. The idea is simple: each pair of players i, j share a secret with which they can securely encrypt their communication. That secret is the XOR of the random bits on the foreheads of all other players. By standard arguments about one-time padding, no player $l \neq i, j$ can compute anything about any message thus encrypted, so i, j have an untappable private channel. Thus, players can simulate the zero-information protocol using private channels to achieve the desired protocol for f. □

As the following two sections show, NOF randomness is crucial to this result. Without the shared secrets that NOF randomness gives players, it is not possible in general to compute without nontrivial information leakage.

2.5 The Information Complexity of AND

In this section, we develop a family of related techniques to prove explicit multiparty information complexity lower bounds. Our model function will be AND_k, the AND of k bits. We will present the simplest technique first, which permits bounds on DIC and $R^{pub}IC$, to lay bare the core combinatorial idea, and then we will generalize to allow private randomness.

Our techniques rely on identifying very small families of inputs on which we can tightly characterize sensitivities of both AND and the messages in our protocol to small changes in the input vector. These small families form combinatorial squares, which we call *critical squares*. In the deterministic case, a single critical square is all that's needed to get a strong lower bound. When private randomness is added, the players become able to spread out the information they share over arbitrarily-many messages and therefore we to get a strong bound will need to work with many critical squares; however, our techniques are not yet capable of this.

Our techniques should extend readily to functions with similar input sensitivity to AND_k (for example, OR_k). It is not yet clear how and whether the technique generalizes to more complicated functions (such as GIP or DISJ).

2.5.1 Lower Bounds on $DIC(AND)$ and $R^{pub}IC(AND)$

In this section we prove the following theorem:

Theorem 2.5.1 (restatement of 1.3.2)**.**

$$DIC_\mu(AND_k) \geq \frac{k+1}{2^{k-1}}$$
$$R^{pub}IC^0_\mu(AND_k) \geq \frac{k+1}{2^{k-1}}$$

The following lemma captures the main new intuition behind the proof. We prove a deterministic version here, which gives strong lower bounds but does not apply to the randomized case. (Later, we will prove a randomized version which gives weaker bounds.) version.

Lemma 2.5.2. *Let Π be a protocol computing AND_k exactly. Let T be the protocol tree for Π. Given a family σ of inputs, let T_σ be the subtree of T induced by considering only inputs in σ. Suppose there exists a player i so that there is a node owned by a player j in $T_{\mathbf{1}} \cap T_{\mathbf{1}^i} \cap T_{\mathbf{1}^j} \cap T_{\mathbf{1}^{i,j}}$ which is labeled by a function f so that $f|_{\mathbf{1}^j_{-i}}$ is nonconstant. Then $I(X_i; \Pi(X)|X_{-i}, AND_k(X)) \geq 2^{-(k-1)}$, where the information is with respect to the uniform distribution on inputs.*

Proof. The proof is just a calculation:

$$\begin{aligned} I(X_i; \Pi(X)|X_{-i}, AND_k(X)) &= \mathbb{E}_{x_{-i} \sim X_{-i}}[I(X_i; \Pi(X)|X_{-i} = x_{-i}, AND_k(X))] \\ &\geq \frac{1}{2^{k-1}} I(X_i; \Pi(X)|X_{-i} = \mathbf{1}^j_{-i}, AND_k(X)) \end{aligned}$$

Since $AND_k|_{\mathbf{1}^j_{-i}}$ is constantly 0, this is exactly

$$\frac{1}{2^{k-1}} I(X_i; \Pi(X)|X_{-i} = \mathbf{1}^j_{-i}).$$

Since we know f appears in the transcript $\Pi(X)$ when X takes a value in $\mathbf{1}^j_{-i}$, we get that this is at least

$$\frac{1}{2^{k-1}} I(X_i; f(X)|X_{-i} = \mathbf{1}^j_{-i}) = \frac{1}{2^{k-1}}$$

as desired. □

Proof of theorem 2.5.1. We begin by proving $DIC_\mu(AND_k) \geq \frac{1}{2^{k-1}}$.

Let Π be a k-way deterministic protocol exactly computing AND_k. By definition and the chain rule for mutual information,

$$IC_\mu(\Pi) = \sum_i I(X_i; \Pi(X)|X_{-i}) = \sum_i I(X_i; \operatorname{out} \Pi(X)|X_{-i}) + I(X_i; \Pi(X)|X_{-i}, \operatorname{out} \Pi(X)).$$

We will deal with the boring part first. We know $\operatorname{out} \Pi(X) = AND_k(X)$, and it is not too hard to see that

$$\mathbb{E}_{x_{-i} \sim X_{-i}}[I(X_i; AND_k(X)|X_{-i} = x_{-i})] = \frac{1}{2^{k-1}} I(X_i; AND_k(X)|X_{-i} = \mathbf{1}_{-i}) = \frac{1}{2^{k-1}}$$

and therefore that

$$\sum_i I(X_i; \operatorname{out} \Pi(X)|X_{-i}) = \frac{k}{2^{k-1}}.$$

Now for the interesting part. We must show that

$$\sum_i I(X_i; \Pi(X)|X_{-i}, AND_k(X)) \geq \frac{1}{2^{k-1}}.$$

It will be sufficient to find a node of T satisfying the requirements of lemma 2.5.2.

Let f be the function labeling the least-depth node across all $T_{\mathbf{1}_{-i}}$ (for $1 \leq i \leq k$) so that $f|_{\mathbf{1}_{-i}}$ is nonconstant (such an f must exist, since the last message of Π is $AND_k(X)$ which is clearly nonconstant when restricted to $\mathbf{1}_{-i}$). Let $g = f|_{\mathbf{1}_{-i,j}}$. Let j be the index of the player who speaks g. Since g is the least-depth function nonconstant on $\mathbf{1}_{-i}$, clearly $g \in T_{\mathbf{1}} \cap T_{\mathbf{1}^i}$. If $g \notin T_{\mathbf{1}^j}$ then there is some lower-depth f' which is noncontant on 1_{-j}, contrary to construction. Finally, if $g \notin T_{\mathbf{1}^{i,j}}$ then there is some lower-depth f' which is noncontant on both $\mathbf{1}^j_{-i}$ and $\mathbf{1}^i_{-j}$, satisfying the hypotheses of the lemma, so without loss of generality we assume $f \in T_{\mathbf{1}_{-i,j}}$.

The combinatorial square of inputs $\mathbf{1}_{-i,j}$ is a *critical square* for f. We use the term loosely to mean that $g = f|\mathbf{1}_{-i,j}$ can be shown to satisfy the hypotheses of lemma 2.5.2 for either $\mathbf{1}^j_{-i}$ or $\mathbf{1}^i_{-j}$.

Write $g(a,b)$ for the value of g when $x_i = a$ and $x_j = b$ (note that this is abuse of notation since technically g does not have x_j as an input). If $g(0,0) \neq g(1,0)$ we are done. Suppose otherwise. We know $g(1,1) \neq g(0,1)$. Then either $g(1,1) \neq g(1,0)$ or $g(0,0) \neq g(0,1)$. If the latter, we are done. The former is impossible: if $g(1,1) \neq g(1,0)$, then $f_{\mathbf{1}_{-j}}$ is nonconstant, but since f does not have x_j as an input, there is some lower-depth f' which is nonconstant on $\mathbf{1}_{-j}$. This is contrary to construction, which completes the proof of the first bound.

The second result, that $R^{pub}IC^0_\mu(AND_k) \geq \frac{k+1}{2^{k-1}}$, now follows quickly. Let Π be a protocol with public randomness computing AND_k with zero error under μ. We think of Π as a distribution over deterministic protocols Π_r. Each deterministic protocol Π_r has $IC_\mu(\Pi_r) \geq \frac{k+1}{2^{k-1}}$ by the deterministic lower bound. Also, observe that where R is the public randomness in Π, we have that $X \to \Pi(X) \to R$ forms a Markov chain. Thus,

$$\begin{aligned}
IC_\mu(\Pi) &= \sum \mathbb{E}[I(X_i; \Pi(X)|X_{-i} = x_{-i}] \\
&\geq \sum \mathbb{E}[I(X_i; \Pi(X)|X_{-i} = x_{-i}, R)] \\
&= \sum \mathbb{E}_{X_i}[\mathbb{E}_R[I(X_i; \Pi(X)|X_{-i} = x_{-i}, R = r)]] \\
&= \mathbb{E}_R[\sum \mathbb{E}_{X_i}[I(X_i; \Pi_r(X)|X_{-i})]] \\
&\geq \frac{k+1}{2^{k-1}}.
\end{aligned}$$

□

2.5.2 Extension to Private Randomness

We conjecture that the bound we prove in the deterministic and public-coin cases also applies in the private-coin case.

Conjecture 2.5.3.

$$RIC^0_\mu(AND_k) \geq \frac{k+1}{2^{k-1}}.$$

It appears that the private-coin case is a great deal more complicated than the public-coin case. We are only able to prove partial results towards this conjecture, and they

involve somewhat more technical tools. We are able to prove that no private-coin protocol can compute AND_k with zero information, and in the case of bounded rounds we are able to prove a quantitative lower bound. However, our quantitative lower bound decays to zero as the number of rounds grows, so we are unable to rule out the possibility that the private-coin information complexity of AND_k is *asymptotically* zero.

The first difficulty is that the player who speaks the t-th message may vary on a fixed input across choices of random coins. We fix this by converting general public coin protocols to protocols which proceed by rounds. A protocol proceeds by rounds if the speaking order is round-robin: players always speak in the order $1, 2, 3, \ldots, k, 1, \ldots$. This means that the speaking order of the players is oblivious to both the input and the randomness in the protocol, which is what we desire.

Lemma 2.5.4. *Let Π be a randomized protocol. Then there is a protocol Π' which proceeds by rounds and for every input has output distributed exactly as those of Π so that for any distribution ν,*

$$IC_\nu(\Pi) = IC_\nu(\Pi').$$

Furthermore, the number of rounds in Π' is exactly the communication cost of Π.

Intuitively, lemma 2.5.4 says that the speaking order of the players does not itself carry information. The reason for this is that by definition at each point in the execution of a protocol every player knows whose turn it is to speak.

Proof of lemma 2.5.4. Fix a protocol Π. In our protocol Π', players will simulate Π. The protocol Π' has a round of communication for every message in Π. Suppose that for some fixing of the inputs and private randomness of Π, player j speaks the rth message. Then in round r of Π' (with the same inputs and private coins), every player but j sends a 0, and player j sends whatever she would have in protocol Π.

It is easy to see that the "filler" messages carry no information when conditioned on all preceding messages, so the information cost of Π' is precisely that of Π. Since there is a round in Π' for every message of Π, the number of rounds in Π' is exactly the communication cost of Π. □

The following lemma is the probabilistic analogue of lemma 2.5.2.

Lemma 2.5.5. *Fix a private-coin randomized protocol Π which proceeds by rounds and exactly computes AND_k. Denote its input vector by X. For a fixed input $y \in \{0,1\}^n$ there a distribution of transcripts $\Pi|X = y$. Furthermore, at each time t in the protocol (when there are t bits of $\Pi|X = t$ on the blackboard) there is an estimate of $\Pi|X = y$ using the information on the blackboard. Denote this estimate by $\Pi^t|X = y$. Suppose there are i, j and t so that*

$$h(\Pi^t|X = \mathbf{1}, \Pi^t|X = \mathbf{1}^i) - h(\Pi^t|X = \mathbf{1}, \Pi^t|X = \mathbf{1}^j) \geq \delta.$$

Then

$$IC_\mu(\Pi) \geq \frac{\delta^2/2 + k}{2^{k-1}}$$

Proof. The $k/2^{k-1}$ term is, as in the deterministic case, the information between the final output and the inputs. Similarly to the deterministic case, we will find $\delta^2/2$ bits of information in a very tightly concentrated distribution and lose a factor of 2^{k-1} in passing to the uniform distribution. Again as in the deterministic case, we will do this by looking at input distributions on which the output is known to most players before the protocol is even run and showing that players still learn about their inputs when the protocol is executed.

By hypothesis, there are i, j, t so that

$$h(\Pi^t|X = \mathbf{1}\,, \Pi^t|X = \mathbf{1}^i) - h(\Pi^t|X = \mathbf{1}\,, \Pi^t|X = \mathbf{1}^j) \geq \delta.$$

Recall that h is a metric and therefore satisfies the triangle inequality. Therefore,

$$\begin{aligned} &h(\Pi^t|X = \mathbf{1}^j\,, \Pi^t|X = \mathbf{1}^{i,j}) + h(\Pi^t|X = \mathbf{1}^i\,, \Pi^t|X = \mathbf{1}^{i,j}) \\ &\geq h(\Pi^t|X = \mathbf{1}^j\,, \Pi^t|X = \mathbf{1}^i) \\ &\geq h(\Pi^t|X = \mathbf{1}\,, \Pi^t|X = \mathbf{1}^i) - h(\Pi^t|X = \mathbf{1}\,, \Pi^t|X = \mathbf{1}^j) \geq \delta \end{aligned}$$

and by Cauchy-Schwartz,

$$h^2(\Pi^i|X = \mathbf{1}^j\,, \Pi^i|X = \mathbf{1}^{i,j}) + h^2(\Pi^j|X = \mathbf{1}^i\,, \Pi^j|X = \mathbf{1}^{i,j}) \geq \frac{\delta^2}{2}.$$

By Lin's lemma (lemma 1.2.29),

$$I_{\nu_i}(X_i; \Pi^t|X_{-i} = \mathbf{1}^j) + I_{\nu_j}(X_j; \Pi^t|X_{-j} = \mathbf{1}^i) \geq \frac{\delta^2}{2}$$

where ν_i is uniform on $\mathbf{1}^j_{-i}$ and ν_j is uniform on $\mathbf{1}^i_{-j}$. Note that $\Pi^t|X = y$ can be obtained from $\Pi|X = y$ by dropping all but the first t bits and computing the estimate $\Pi^t|X = y$. Thus by the data processing inequality,

$$I_{\nu_i}(X_i; \Pi^i|X_{-i} = \mathbf{1}^j) \leq I_{\nu_i}(X_i; \Pi|X_{-i} = \mathbf{1}^j).$$

and similarly for j.

The result now follows by the same argument as in the deterministic case. □

Theorem 2.5.6 (restatement of 1.3.3)**.** *Let Π be a protocol with r rounds exactly computing AND_k with private coins. Then*

$$IC^0_\mu(\Pi) \geq \frac{1}{8(kr)^2 2^{k-1}} + \frac{k}{2^{k-1}}.$$

The proof will rely on the following lemma.

Lemma 2.5.7. *Let Π be a protocol with r rounds exactly computing AND_k with private coins. Then for all i,*

$$h(\Pi|X = \mathbf{1}^i, \Pi|X = \mathbf{1}) = 1$$

Proof of lemma 2.5.7. Let the random variables $\Pi|X = \mathbf{1}^i$ and $\Pi|X = \mathbf{1}$ be distributed according to ν_0 and ν_1, respectively. The length of transcripts of Π is exactly kr, so there are 2^{kr} possible transcripts. Without loss of generality, let the last bit of the transcript be the output. (this can be preserved in the rounds-based setting by slightly altering the transformation of the protocol to rounds). Let U_0 be the set of transcripts ending in a 0 and U_1 those ending in a 1. Then ν_0 is supported on U_0 and ν_1 is supported on U_1. Then ν_0 and ν_1 have disjoint support, so

$$\begin{aligned} h(\nu_0, \nu_1) &= \frac{1}{\sqrt{2}} \sqrt{\sum_{x \in U_0} (\sqrt{\nu_0(x)} - \sqrt{\nu_1(x)})^2 + \sum_{x \in U_1} (\sqrt{\nu_0(x)} - \sqrt{\nu_1(x)})^2} \\ &= \frac{1}{\sqrt{2}} \sqrt{\sum_{x \in U_0} \nu_0(x) + \sum_{x \in U_1} \nu_1(x)} \\ &= 1. \end{aligned}$$

□

Proof of theorem 2.5.6. For each player i, define the sequence $h_0^i \dots h_{kr}^i$ where

$$h_t^i = h(\Pi^t | X = \mathbf{1}^i, \Pi^t | X = \mathbf{1}).$$

That is, h_t^i is the Hellinger distance between the estimate at time t of the final transcript when $X = \mathbf{1}^i$ and her prediction when $X = \mathbf{1}$.

For all i, clearly $h_0^i = 0$, and by lemma 2.5.7, we know $h_{kr}^i = 1$. So for all i there is some t so that

$$h_t^i - h_{t-1}^i \geq \frac{1}{kr}$$

Fix a player i and let t be such that

$$h_t^i - h_{t-1}^i \geq \frac{1}{kr}$$

as above. Let player j speak message t. Then clearly $h_t^j = h_{t-1}^j$, so we get

$$h_t^i - h_t^j + h_{t-1}^j - h_{t-1}^i \geq \frac{1}{kr}$$

Therefore, either $h_t^i - h_t^j \geq 1/(2kr)$ or $h_{t-1}^j - h_{t-1}^i \geq 1/(2kr)$.

Now lemma 2.5.5 applies to give the result. □

Chapter 3

Conclusions and Open Problems

We believe that the results in this thesis demonstrate that there is hope for a useful theory of multiparty information complexity, and that furthermore the lower-bound techniques made available by such a theory are candidates to break the $\log n$ barrier and tighten bounds on DISJ in multiparty communication. Our results are, however, are merely proofs of concept. In the following section, we list some open problems whose solutions we believe to be probable next steps in the development of multiparty information complexity.

3.1 Open Problems

On our line of investigation, it is open whether any function has private-coin information complexity bounded away from zero independent of the number of rounds. Specifically, it is open to prove conjecture 2.5.3.

Our lower bounds are proved for a function whose input is a single bit to each player and whose sensitivity structure with respect to those bits is quite simple. It is open to prove an information complexity lower bound for a function on more than single bits, or whose sensitivity is less simple than AND_k.

We know that in a general randomness-on-forehead setting the information complexity of an arbitrary function goes to zero. However, it is unclear whether the same is true if the number of messages is limited. In particular, we conjecture that in the simultaneous model, where each player sends exactly one message to an external referee who computes the output (and who does not seem the inputs or the randomness), some function has nonzero information complexity even with randomness on the forehead.

On a more metatheoretical tack: the inspiration to investigate multiparty information comes from the two-player result that information bounds beat rectangle bounds. It is open whether the analogous result holds for more than two parties. Extending the two-party proof seems to require new multiparty rejection sampling techniques.

Bibliography

[AA80] Harold Abelson and Peter Andreae. Information transfer and area-time tradeoffs for vlsi multiplication. *Communications of the ACM*, 1980.

[AB09] Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009.

[AW08] Scott Aaronson and Avi Wigderson. Algebrization: A new barrier in complexity theory. *STOC*, 2008.

[BBCR10] Boaz Barak, Mark Braverman, Xi Chen, and Anup Rao. How to compress interactive communication. *STOC*, 2010.

[BOGW88] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. *STOC*, 1988.

[BR11] Mark Braverman and Anup Rao. Information equals amortized communication. In *FOCS*, pages 748–757, 2011.

[BW12] Mark Braverman and Omri Weinstein. A discrepancy lower bound for information complexity. *RANDOM*, 2012.

[BYJKS04] Ziv Bar-Yossef, T.S. Jayram, Ravi Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *JCSS*, 2004.

[CP] Arkadev Chattopadhyay and Toniann Pitassi. The story of set disjointness. *SIGACT News*.

[CT06] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wiley, 2006.

[JK] Rahul Jain and Hartmut Klauck. The partition bound for classical communication complexity and query complexity.

[KLL+12] Iordanis Kerenidis, Sophie Laplante, Virginie Lerays, Jérémie Roland, and David Xiao. Lower bounds on information complexity via zero-communication protocols and applications. *FOCS*, 2012.

[KN97] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, 1st edition, 1997.

[Lin91] Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 1991.

[LS] Troy Lee and Adi Shraibman. Lower bounds in communication complexit: A survey.

[Raz00] Ran Raz. The bns-chung criterion for multi-party communication complexity. *Computational Complexity*, 9:2000, 2000.

[She13] Alexander Sherstov. Communication lower bounds using directional derivatives. *STOC*, 2013.

[Sip] Michael Sipser. *Introduction to the Theory of Computation*.

[Tho79] C. D. Thompson. Area-time complexity for vlsi. *Ann. ACM SYmp. Thoeory of Computing*, 1979.

[Yao79] Andrew Yao. Some complexity questions related to distributive computing. *STOC*, 1979.