Average and Randomized Communication Complexity

ALON ORLITSKY AND ABBAS EL GAMAL, SENIOR MEMBER, IEEE

***Abstract*—The communication complexity of a two-variable function $f(x, y)$ is the number of information bits two communicators need to exchange to compute f when, initially, each knows only one of the variables. There are several communication-complexity measures corresponding to whether 1) the worst case or average number of bits is considered, 2) computation errors are allowed or not, and 3) randomization is allowed or not. Tight bounds are provided for the typical behavior of all bounded-error communication-complexity measures of Boolean functions. Let $\mathscr{F}_s$ be the set of functions f: $\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}\rightarrow\{0,1\}$ that contain s ones (i.e., $|\{(x,y): f(x,y)=1\}|=s$). It is shown that for every $n\le s\le n^2/2$, the communication-complexity measures fall into two classes:**

- ***$\log n$ class*—the error-free worst case randomized complexity and, more importantly, the error-free worst case deterministic complexity of most functions in $\mathscr{F}_s$ are between $\log n - 4$ and $\log n + 1$ bits (this holds even for $s = n$);**
- ***$\log(s/n)$ class*—the ϵ-error worst case randomized complexity and the ϵ-error average randomized complexity of most functions in $\mathscr{F}_s$ are between $(1-2\epsilon)(\log(s/n) - 2\log\log(s/n)$ and $(1-2\epsilon)(\log(s/n)+5.3\log\log n)$ bits. More importantly, the error-free average deterministic complexity of *all* functions in $\mathscr{F}_s$ is less than $\log(s/n)+8.3\log\log n$ bits. For most of these functions it is also $\ge\log(s/n)-2\log\log(s/n)$ bits.**

The difference between the complexities of the $\log n$ class and the $\log(s/n)$ class ranges from a constant (for $s\approx n^2/2$) to exponential (for $s\approx n\log n$). In particular, since most functions have about $n^2/2$ ones, all their complexity measures are around $\log n$ bits.

I. Introduction

COMMUNICATION complexity was first introduced by Abelson [1] and Yao [2]. It is concerned with different aspects of the following problem: n is a positive integer, f is a binary function defined on $\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}$, and $\epsilon\ge 0$. Two *communicators*, P_x having a random integer $X\in\{0,\cdots,n-1\}$ and P_y having a random integer $Y\in\{0,\cdots,n-1\}$, use a predetermined protocol to compute $f(X,Y)$ with probability of error $\le\epsilon$. How many bits must they exchange on the average? at worst?

The next two sections make these notions more precise: in Section II we formally define the deterministic model; in Section III we describe randomized protocols and compare them to deterministic ones. In Section IV we survey previous work and describe the results obtained in this paper. These results are proved in Section V.

II. Deterministic Protocols

Communication is performed over a binary channel that can carry only one bit at a time. Therefore, each communicator, in turn, transmits a (possible empty) sequence of bits that we call a *string* or a *message*. Since empty messages are allowed, we lose no generality by assuming that P_x transmits the first message. Communication proceeds according to a predetermined protocol that, at each time, specifies the message transmitted and, at the end of the communication, decides on the computed value that the communicators (P_x and P_y) assume is the correct value of the function. The protocol is *deterministic* if, whenever P_x and P_y have the same pair of random integers, they transmit the same sequence of messages and decide on the same computed value.

With this communication model in mind, we formally define deterministic protocols. The definition is similar to that of [3] and [4]. It differs from that of [3] in that 1) it views the protocol as a codebook (an approach that facilitates information-theoretic treatment), and 2) the computed value of the function does not have to be transmitted explicitly. An *input* is an ordered pair $(x,y)\in\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}$ thought of as a value assignment for X and Y. If $b_k,\cdots,b_l$ are strings, we let $\langle b_k,\cdots,b_l\rangle$ or $\langle b_j\rangle_{j=k}^{l}$ denote the $(l+1-k)$-element sequence whose ith element is b_{k+i-1}. A *deterministic protocol* ϕ for $\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}$ can therefore be defined as a mapping that associates with each input.

1) a sequence $\langle b_1(x,y), b_2(x,y),\cdots, b_{m(x,y)}(x,y)\rangle$ of transmitted messages, all the odd-indexed messages to be transmitted by P_x and all the even-indexed messages by P_y;
2) a *computed value* $v_\phi(x,y)$ that P_x and P_y assume is equal to $f(x,y)$.

Although the number of messages, $m(x,y)$, is finite for all protocols of interest, we extend the definition of $b_i(x,y)$ indefinitely and let $b_i(x,y)\overset{\text{def}}{=}\varnothing$, the empty string, for all $i\ge m(x,y)$. This simplifies notation and shortens proofs.

Remarks:

1) The messages depend, of course, on the protocol used, but to shorten notation we write $b_i(x,y)$ and $m(x,y)$ without the (sub/super) script ϕ.

Manuscript received August 1, 1987; revised May 18, 1989. This work was partially presented at the IEEE Symposium on Information Theory, Ann Arbor, MI, 1986.

A. Orlitsky is with AT&T Bell Laboratories, Room 2C-370, 600 Mountain Avenue, Murray Hill, NJ 07974.

A. El Gamal is with the Information Systems Laboratory, Stanford University, Stanford, CA 94305.

IEEE Log Number 8932717.

0018-9448/90/0100-0003$01.00 ©1990 IEEE

2) In general, each of P_x and P_y could decide on a different computed value. However, any protocol ensuring that a given communicator (say, P_x) knows $f(X,Y)$ for all inputs can be augmented by one bit to yield a protocol ensuring that both P_x and P_y know $f(X,Y)$ for all inputs. (Note that requiring merely that at least one communicator know $f(X,Y)$ for all inputs trivializes the problem as P_x can always decide that $f(X,Y)=0$ and P_y that $f(X,Y)=1$.)

Not every protocol can be carried out in a distributed environment where, initially, P_x knows only X, P_y only knows Y, and their only information about each other's random variable derives from previously exchanged messages. To enable communication in such an environment, we restrict consideration to protocols possessing the following properties.

Separate-Transmissions Property

All that a communicator knows prior to transmitting a message is a random variable and previous transmissions. To guarantee that his message will not depend on the other communicator's variable, we require that, for all x, x', y, y':

for odd i,

$\langle b_j(x,y)\rangle_{j=1}^{i-1} = \langle b_j(x,y')\rangle_{j=1}^{i-1}$ implies that

$$b_i(x,y) = b_i(x,y')$$

and for even i,

$\langle b_j(x,y)\rangle_{j=1}^{i-1} = \langle b_j(x',y)\rangle_{j=1}^{i-1}$ implies that

$$b_i(x,y) = b_i(x',y)$$

Prefix-Free-Messages Property

Let $\alpha_1,\cdots,\alpha_k$ and $\beta_1,\cdots,\beta_l$ be bits. The string $\alpha_1\cdots\alpha_k$ is a *prefix* of the string $\beta_1\cdots\beta_l$ if $k \le l$ and $\alpha_i = \beta_i$ for $i=1,\cdots,k$. It is a *proper prefix* if, in addition, $k \ne l$. A set of strings is *prefix-free* if no string in the set is a prefix of another. For each person to know when the message he receives terminates, we require that, given his random variable and previous transmissions, the set of all possible messages he can receive is prefix-free. Hence, for odd i,

$$\langle b_j(x,y)\rangle_{j=1}^{i-1} = \langle b_j(x',y)\rangle_{j=1}^{i-1}$$

implies that neither $b_i(x,y)$ nor $b_i(x',y)$ is a proper prefix of the other, and for even i,

$$\langle b_j(x,y)\rangle_{j=1}^{i-1} = \langle b_j(x,y')\rangle_{j=1}^{i-1}$$

implies that neither $b_i(x,y)$ nor $b_i(x,y')$ is a proper prefix of the other.

Separate-Decisions Property

When a communicator decides on the computed value, he must, again, base that decision on the random variable and the messages exchanged. Therefore,

$\langle b_j(x,y)\rangle_{j=1}^{\infty} = \langle b_j(x,y')\rangle_{j=1}^{\infty}$ implies that

$$v_\phi(x,y) = v_\phi(x,y'),$$

and

$\langle b_j(x,y)\rangle_{j=1}^{\infty} = \langle b_j(x',y)\rangle_{j=1}^{\infty}$ implies that

$$v_\phi(x,y) = v_\phi(x',y).$$

Remarks:

1) Note that the first two properties must apply to all i's, even those larger than $m(x,y)$.

2) It can be shown that, although empty messages are allowed, the prefix-free-messages property does not increase the amount of communication required. Every protocol containing empty messages can be trivially modified to obtain a protocol that transmits the same number of bits but does not contain any empty message.

We assume henceforth that all protocols possess these three properties. Now we turn to the complexity definitions.

Let $l_\phi(x,y) \overset{\text{def}}{=} \sum_{i=1}^{m(x,y)} |b_i(x,y)|$ denote the total number of bits transmitted according to the protocol ϕ when P_x knows x and P_y knows y. The worst case complexity of ϕ is defined as

$$\hat{L}_\phi \overset{\text{def}}{=} \max\left\{l_\phi(x,y) : (x,y) \in \{0,\cdots,n-1\} \cdot \{0,\cdots,n-1\}\right\}.$$

The *average complexity* of ϕ is defined as

$$\bar{L}_\phi \overset{\text{def}}{=} \frac{1}{n^2} \sum_{(x,y)\in\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}} l_\phi(x,y).$$

If f is a function defined on $\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}$ and ϕ is a protocol for $\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}$, we say that ϕ is *error-free* for f if $v_\phi(x,y) = f(x,y)$ for all $(x,y)\in\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}$. The worst case error-free communication complexity of f is then defined as

$$\hat{C}_D(f,0) \overset{\text{def}}{=} \min\left\{\hat{L}_\phi(x,y) : \phi \text{ is an error-free deterministic protocol for } f\right\}$$

and the average error-free communication complexity of f:

$$\bar{C}_D(f,0) \overset{\text{def}}{=} \min\left\{\bar{L}_\phi(x,y) : \phi \text{ is an error-free deterministic protocol for } f\right\}.$$

The subscript D stands for *deterministic*, whereas the zero indicates that no errors are allowed (zero probability of error). As might be suspected by the excessive notation, both requirements will soon be relaxed.

III. Randomized Protocols

Randomized protocols are like deterministic ones, except that P_x and P_y may use "coin flips" to determine their transmissions. Whereas deterministic protocols require that P_x base his transmissions and computed value only on X and preceding transmissions (similarly for P_y), randomized protocols require that P_x base the bias of the coin flips (which determine his transmissions and computed value) only on X and preceding transmissions (similarly, for P_y). Still, the set of all messages that have positive probability at any time must be prefix-free. Furthermore, the computed values must always agree (this requires a simple rejustification).

We denote randomized protocols by Φ (as opposed to ϕ for deterministic protocols). The number of bits transmitted when P_x knows x and P_y knows y is now a random variable denoted by $L_\Phi(x,y)$. Its expected value is denoted by $\bar{L}_\Phi(x,y)$.

The definitions of average and worst-case complexities of protocols can now be extended to randomized protocols. The *worst case complexity* $\hat{L}_\Phi$ of Φ is

$$\hat{L}_\Phi \overset{\text{def}}{=} \max\left\{\bar{L}_\Phi(x,y) : (x,y) \in \{0,\cdots,n-1\} \cdot \{0,\cdots,n-1\}\right\}.$$

The *average complexity* $\bar{L}_\Phi$ of Φ is

$$\bar{L}_\Phi \overset{\text{def}}{=} \frac{1}{n^2} \sum_{(x,y)\in\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}} \bar{L}_\Phi(x,y).$$

Similarly, the *computed value* $V_\Phi(x,y)$, which P_x and P_y assume is the value of $f(x,y)$, is now a 0–1 random variable. We let $\bar{E}_\Phi(x,y) \overset{\text{def}}{=} p(V_\Phi(x,y) \neq f(x,y))$ denote the probability that the computed value determined by Φ is wrong for (x,y), and define $\hat{E}_\Phi(f)$, the worst case error incurred by Φ in computing f, as

$$\hat{E}_\Phi(f) \overset{\text{def}}{=} \max\left\{\bar{E}_\Phi(x,y) : (x,y) \in \{0,\cdots,n-1\} \times \{0,\cdots,n-1\}\right\}.$$

Remarks:

1) The maximum in $\hat{L}_\Phi$ and $\hat{E}_\Phi(f)$ is taken over the inputs. For each input we still average over the "coin flips."

2) For brevity we consider only worst case errors. Average error results (when average errors are appropriately defined) can also be obtained.

Using these quantities, we define two more communication-complexity measures. The worst case randomized communication complexity of a function f with ϵ error:

$$\hat{C}_R(f,\hat{\epsilon}) \overset{\text{def}}{=} \min\left\{\hat{L}_\Phi : \hat{E}_\Phi(f) \le \epsilon\right\}$$

and the average randomized communication complexity of f with ϵ error:

$$\bar{C}_R(f,\hat{\epsilon}) \overset{\text{def}}{=} \min\left\{\bar{L}_\Phi : \hat{E}_\Phi(f) \le \epsilon\right\}.$$

Again, $\hat{C}$ denotes worst case complexity while $\bar{C}$ stands for average complexity, and the subscript R indicates that randomized protocols are allowed. The caret in $\hat{\epsilon}$ means that ϵ worst case error is permitted (we will later use $\bar{\epsilon}$ for ϵ average error).

A simple relationship between randomized and deterministic protocols is useful in proving lower bounds on randomized protocols. Let Φ be a randomized protocol. Each transmission is the outcome of a random experiment whose probability distribution is determined by Φ according to the integer known to the transmitter and previous transmissions. Conceptually, each communicator can perform all the random experiments (corresponding to all possible integers she might have and all possible transmissions) prior to the commencement of the communication. Then, given the communicator's value and previous transmissions, he consults the appropriate experiment and discards the rest.

These two huge random experiments, each performed by one communicator, are independent of the current input. Every combined outcome of these two random experiments induces a deterministic protocol. Let $p(\phi)$ be the total probability of outcomes that induce the deterministic protocol ϕ. As before, each ϕ determines $l_\phi(x,y)$ and $v_\phi(x,y)$. The expected communication length for (x,y), denoted earlier as $\bar{L}_\Phi(x,y)$, and the probability of error $\bar{E}_\Phi(x,y)$ can now be written as

$$\bar{L}_\Phi(x,y) = \sum_\phi p(\phi) l_\phi(x,y)$$

$$\bar{E}_\Phi(x,y) = \sum_{\{\phi:\, v_\phi(x,y) \neq f(x,y)\}} p(\phi). \tag{1}$$

These equations clearly imply that

$$\bar{L}_\Phi = \sum_\phi p(\phi) \bar{L}_\phi$$

$$\bar{E}_\Phi(f) = \sum_\phi p(\phi) \bar{E}_\phi(f). \tag{2}$$

Before proceeding further, we note that there are two possible models:

1) *shared random sources*: the communicators have access to each other's random experiment.

2) *separate random sources*: the communicators do not have access to each other's random experiment.

Clearly, $\hat{C}_R(f,\hat{\epsilon})$ and $\bar{C}_R(f,\hat{\epsilon})$ depend on the model assumed. The bounds we prove hold in a strong sense: all the lower bounds apply even with shared random sources, i.e., even when we combine the two random experiments into one that is known to both communicators. All the upper bounds apply even if the experiments are separated, each known to one communicator. Therefore, all bounds apply to both models.

We have so far defined four complexity measures: $\hat{C}_D(f,0)$, $\bar{C}_D(f,0)$, $\hat{C}_R(f,\hat{\epsilon})$, and $\bar{C}_R(f,\hat{\epsilon})$. Of special interest are the randomized complexities when no errors are allowed ($\epsilon = 0$)—the so-called "Las Vegas" complexities. There are two Las Vegas complexity measures ($\hat{C}_R(f,0)$ and $\bar{C}_R(f,0)$, but it can be easily seen that for all functions f, $\bar{C}_R(f,0) = \bar{C}_D(f,0)$. (We omit the carets because 0 worst case and 0 average errors are the same.) This leaves five different measures whose complexities we attempt to ascertain next.

Remark: As in the case of randomized complexities, one could define the deterministic measures $\bar{C}_D(f,\hat{\epsilon})$ and $\hat{C}_D(f,\hat{\epsilon})$. However, for deterministic protocols, $\hat{E}_\phi(f)$ is either zero or one, thus $\hat{C}_D(f,\hat{\epsilon}) = \hat{C}_D(f,0)$ and $\bar{C}_D(f,\hat{\epsilon}) = \bar{C}_D(f,0)$.

IV. Previous and New Results

Complexity measure A is smaller than complexity measure B if $A(f) \le B(f)$ for all functions f. For example, $\overline{C}_R(f,\hat{\epsilon}) \le \hat{C}_R(f,\hat{\epsilon})$ for all functions f. This relation induces a partial order on the five complexity measures which is depicted in Fig. 1.

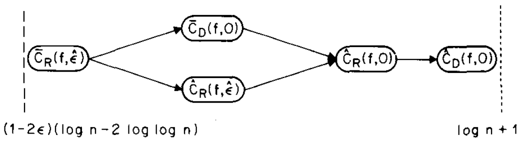
$\overline{C}_R(f,\hat{\epsilon})$
$\overline{C}_D(f,0)$
$\hat{C}_R(f,\hat{\epsilon})$
$\hat{C}_R(f,0)$
$\hat{C}_D(f,0)$
$(1-2\epsilon)(\log n - 2 \log\log n)$
$\log n + 1$

Fig. 1. Known relations for most functions.

The vertical lines bound the complexity of most functions. The dotted line therefore means that, for most functions, all complexity measures are smaller than $\log n + 1$ (in this case, it is trivially true for all functions). The dashed line and other known results are described in the next two paragraphs.

Yao [2] proved that, for most functions, $\hat{C}_D(f,0)$ is the worst possible: about $\log n$. The proof can be easily modified to show that the same is true for $\overline{C}_D(f,0)$. For randomized protocols, Yao showed that the equality function (1 if $x = y$, 0 otherwise), has $\hat{C}_D(\text{equ},0) = \log n + 1$, but, for all $\epsilon > 0$, $\hat{C}_R(\text{equ},\hat{\epsilon}) = \Theta(\log\log n)$. Mehlhorn and Schmidt demonstrated [5] a function (defined in Example 2) for which $\hat{C}_D(f,0) = M$, but $\hat{C}_R(f,0) = O(\sqrt{M}\log^2 M)$.

More recently, it has been shown [6], [7] that for most functions $\overline{C}_R(f,\hat{\epsilon}) > (1-2\epsilon)(\log n - \log\log n) - 2$. This result, indicated in Fig. 1 by a dashed vertical line, implies that for most functions it does not help to allow randomization or ϵ error or even to measure only the average complexity—the complexity remains about the same. In particular, it means that the equality and the function of [5] are more the exception than the rule.

Let $\mathbf{1}_f \overset{\text{def}}{=} \{(x,y)\colon\ f(x,y) = 1\}$ be the set of inputs for which f is 1 and $\mathbf{0}_f$ be the set of inputs for which f is 0. The vast majority of binary functions have about equal number of ones and zeros in their function table. Therefore, saying that a certain property holds for most functions, is about the same as saying that it holds for most functions with $|\mathbf{1}_f| \approx n^2/2$. A natural question, thus, is whether the complexity is reduced when the number of ones and zeros is not balanced, i.e., $|\mathbf{1}_f| \ll n^2/2$ or $|\mathbf{0}_f| \ll n^2/2$.

Similar questions have been asked concerning the compression of binary sequences. Considerable insight has been obtained by noting that sequences are compressible to their entropy, notwithstanding the fact that most sequences are incompressible because their entropy is maximal. Does a similar property hold for communication complexity?

Without loss of generality, we consider only functions with fewer ones than zeros, $\mathbf{1}_f \le n^2/2$. For every integer $s \le n^2/2$, define $\mathscr{F}_s \overset{\text{def}}{=} \{f\colon\ |\mathbf{1}_f| = s\}$ (the set of functions with s ones). We prove that for all $s \ge n$, the following hold true.

1) (Corollary 4) Most functions in $\mathscr{F}_s$ have worst case error-free (Las Vegas) complexities, $\hat{C}_D(f,0)$ and $\hat{C}_R(f,0)$, at least $\log n - 4$ (even when $s = n$).

2) (Corollary 7) However, *all* functions in $\mathscr{F}_s$ have $\overline{C}_D(f,0) \le \log(s/n) + 8.3\log\log n + c$. That is, a deterministic protocol exists that commits no errors and, on the average, exchanges $\le \log(s/n) + 8.3\log\log n + c$ bits.

3) (Corollary 6) Most of the functions in $\mathscr{F}_s$ have $\hat{C}_R(f,\hat{\epsilon}) \le (1-2\epsilon)(\log(s/n) + 5.3\log\log n + 2\log(1/\epsilon))$.

4) (Corollary 8) For most of the functions in $\mathscr{F}_s$, the bounds in (2) and (3) are tight:

$$\overline{C}_R(f,\hat{\epsilon}) \ge (1-2\epsilon)(\log(s/n) - 2\log\log(s/n)) - 9.$$

All the bounds proved apply to both models: shared and separate sources of randomness, and they are all tight up to an additive term. They are summarized in Fig. 2. As in Fig. 1, dotted vertical lines denote upper bounds and dashed lines denote lower bounds.

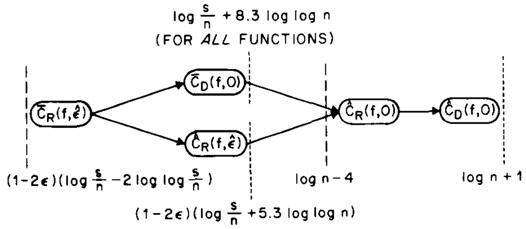
$\log \frac{s}{n} + 8.3 \log\log n$
(FOR *ALL* FUNCTIONS)
$\overline{C}_R(f,\hat{\epsilon})$
$\overline{C}_D(f,0)$
$\hat{C}_R(f,\hat{\epsilon})$
$\hat{C}_R(f,0)$
$\hat{C}_D(f,0)$
$(1-2\epsilon)(\log \frac{s}{n} - 2 \log\log \frac{s}{n})$
$\log n - 4$
$\log n + 1$
$(1-2\epsilon)(\log \frac{s}{n} + 5.3 \log\log n)$

Fig. 2. Bounds for most functions in s, $n < s < n^2/2$.

Some of the consequences of these results are as follows.

1) All the complexity measures fall in one of two distinctly different classes. One (consisting of $\hat{C}_D(f,0)$ and $\hat{C}_R(f,0)$) is at least $\log n - 4$ bits for every s. The other (consisting of $\overline{C}_D(f,0)$, $\hat{C}_R(f,\hat{\epsilon})$, and $\overline{C}_R(f,\hat{\epsilon})$) decreases with s and is about $\log(s/n)$ bits. For $s \approx n^2/2$, there is a small difference between the complexity measures in the two classes. For $s \approx n$ or $s \approx n\log n$, the complexity measures in the first class are exponentially larger than those in the second.

For most functions, $|\mathbf{1}_f| \approx n^2/2$. Therefore, all complexity measures coincide and are about $\log n$ bits. However, for most sparse functions ($|\mathbf{1}_f| \ll n^2/2$), the following hold.

2) In error-free computation, considerably fewer bits are needed on the average than at worst (most interesting, of course, is the difference between $\hat{C}_D(f,0)$ and $\overline{C}_D(f,0)$).

3) In error-free computation, randomization does not help reduce the worst case complexity. Even for $s \approx n$, most functions in $\mathscr{F}_s$ have $\hat{C}_R(f,0) \ge \log n - 4$. Hence $\hat{C}_R(f,0)$ is not much smaller than $\hat{C}_D(f,0)$, and the function described in [5] (see Example 2) is an exception.

4) However, if errors are allowed, randomization does reduce the worst case complexity. In fact, in this case, it is close to average error-free deterministic complexity. Thus the behavior of the equality function is typical of functions in its class: $\mathscr{F}_n$.

V. Proofs

In this section we prove the results claimed. In Section V-A, we prepare the ground-work by exploring some information theoretic properties of protocols. In Section V-B, we show that for most functions, $\hat{C}_R(f,0) \geq \log n - 4$ (Corollary 4). In Section V-C we prove the separation between the classes by showing that for all functions $\overline{C}_D(f,0) \leq \log(|\mathbf{1}_f|/n) + 8.3 \log\log n$ and that, for most functions, $\hat{C}_R(f,\hat{\epsilon}) \leq (1-2\epsilon)(\log(|\mathbf{1}_f|/n) + 5.3 \log\log n) + 2$ (Corollaries 6 and 7). These results are deduced from a stronger bound that takes into consideration the distribution of ones in the function table. Last, in Section V-D Corollary 8, we show that these bounds are tight up to a negligible additive term by proving that for most functions $\overline{C}_R(f,\hat{\epsilon}) \geq (1-2\epsilon)(\log(|\mathbf{1}_f|/n) - 2\log\log(|\mathbf{1}_f|/n) - 9$.

A. Preliminary Results

If $b_k, \cdots, b_l$ are strings, we let $[b_k \cdots b_l]$ or $[b_j]_{j=k}^{l}$ denote the string obtained by their concatenation. We say that the sequence $\langle b_1, \cdots, b_l \rangle$ parses the string b if $b = [b_1 \cdots b_l]$. There usually are many ways to parse a string.

For every (x, y), we let $b_\phi(x, y) \overset{\text{def}}{=} [b_i(x, y)]_{i=1}^{m(x,y)}$, the codeword associated with (x, y), be the concatenation of all messages transmitted according to ϕ when (x, y) is the input. The next theorem shows that only one parsing of $b_\phi(x, y)$ complies with the separate-transmissions and the prefix-free-messages properties.

Theorem 1: If either $b_\phi(x, y)$ or $b_\phi(x', y')$ is a prefix of the other, then

$$\langle b_j(x, y) \rangle_{j=1}^{\infty} = \langle b_j(x', y) \rangle_{j=1}^{\infty} = \langle b_j(x, y') \rangle_{j=1}^{\infty} = \langle b_j(x', y') \rangle_{j=1}^{\infty}.$$

Proof: We show by induction on i that $\langle b_j(x, y) \rangle_{j=1}^{i} = \langle b_j(x', y) \rangle_{j=1}^{i} = \langle b_j(x, y') \rangle_{j=1}^{i} = \langle b_j(x', y') \rangle_{j=1}^{i}$. The induction hypothesis is clearly true for $i = 0$. Assume it is true for $i - 1$; then $[b_j(x, y)]_{j=1}^{i-1} = [b_j(x', y')]_{j=1}^{i-1}$. Hence one of $[b_j(x, y)]_{j=i}^{\infty}$ and $[b_j(x', y')]_{j=i}^{\infty}$ is a prefix of the other, which implies that one of $b_i(x, y)$ and $b_i(x', y')$ is a prefix of the other. If i is odd, then, by the separate-transmissions property and the induction hypothesis, $b_i(x, y') = b_i(x, y)$. By the prefix-free-messages property, neither $b_i(x, y')$ nor $b_i(x', y')$ can be a proper prefix of the other, and hence they must be the same. Using the same argument for (x', y), we get $b_i(x, y) = b_i(x', y) = b_i(x', y')$. The symmetric argument works for even i's, which completes the proof of the induction. Q.E.D.

Besides showing that $b_\phi(x, y)$ can be parsed into the individual messages in a unique way the theorem has three other implications. They are used extensively later so we restate them as individual corollaries. To do so, we need the following definitions.

A *rectangle* is a set of the form $R_x \times R_y$ where $R_x, R_y \subseteq \{0, \cdots, n-1\}$. If S is a subset of $\{0, \cdots, n-1\} \times \{0, \cdots, n-1\}$, we define its *closure* to be $\mathrm{Cl}(S) \overset{\text{def}}{=} \{(x, y)$: for some u and v in $\{0, \cdots, n-1\}$, both (x, u) and (v, y) are in $S\}$. It can be shown that $\mathrm{Cl}(S)$ is the intersection of all rectangles that contain S and is therefore a rectangle. The theorem implies that if $b_\phi(x, y) = b_\phi(x', y')$, then the "corner points" (x, y') and (x', y) have the same codeword. Therefore, if a protocol assigns the same codeword to all inputs in a set S, it assigns the same codeword to all inputs in $\mathrm{Cl}(S)$.

Let $R_\phi(x, y) \overset{\text{def}}{=} \{(u, v)\colon\ b_\phi(u, v) = b_\phi(x, y)\}$ be the set of inputs having the same code-word as (x, y). The first corollary of Theorem 1 states that $R_\phi(x, y)$ can assume only special forms.

Corollary 1: For all $(x, y) \in \{0, \cdots, n-1\} \times \{0, \cdots, n-1\}$, $R_\phi(x, y)$ is a rectangle.

Proof: By the theorem, $(u, v) \in \mathrm{Cl}(R_\phi(x, y))$ implies that $b_\phi(u, v) = b_\phi(x, y)$, i.e., $(u, v) \in R_\phi(x, y)$. Thus $\mathrm{Cl}(R_\phi(x, y)) \subseteq R_\phi(x, y)$. However, $R_\phi(x, y)$ is always contained in $\mathrm{Cl}(R_\phi(x, y))$. Hence $R_\phi(x, y) = \mathrm{Cl}(R_\phi(x, y))$ which is a rectangle. Q.E.D.

This corollary has an intuitive "near-proof." Consider the set S of inputs that can result in a given sequence of messages. By the separate-transmissions property, whenever P_x transmits a message, he partitions this set into *horizontal slices* (sets of the form $S \cap (A_x \times \{0, \cdots, n-1\})$ where the A_x partition $\{0, \cdots, n-1\}$), one slice corresponding to each message. (Note that this would not have been the case without the separate-transmissions property.) Similarly, when P_y transmits a message, he partitions S into *vertical slices* (sets of the form $S \cap (\{0, \cdots, n-1\} \times Ay)$). Since any horizontal (or vertical) slice of a rectangle is again a rectangle, it is clear, by induction, that $R_\phi(x, y)$ is a rectangle. The missing part in this "proof" is showing that different rectangles cannot correspond to the same concatenation of messages. This is the essence of Theorem 1.

The second corollary states that the codewords associated with the rectangles form a prefix-free set. Let $\mathscr{R}_\phi \overset{\text{def}}{=} \{R_\phi(x, y)\colon\ (x, y) \in \{0, \cdots, n-1\} \times \{0, \cdots, n-1\}\}$ be the set of rectangles over which ϕ has a fixed codeword. Any input (x, y) belongs to the rectangle $R_\phi(x, y)$, and any two intersecting rectangles in $\mathscr{R}_\phi$ coincide. Therefore, $\mathscr{R}_\phi$ partitions $\{0, \cdots, n-1\} \times \{0, \cdots, n-1\}$ into $|\mathscr{R}_\phi|$ rectangles where $|S|$ denotes the cardinality of the set S. For each rectangle $R \in \mathscr{R}_\phi$, define $b_\phi(R)$ to be the unique codeword that ϕ assigns to all inputs in R. Then Theorem 1 implies the following.

Corollary 2: $\{b_\phi(R)\colon\ R \in \mathscr{R}_\phi\}$ is prefix-free, and its cardinality is $|\mathscr{R}_\phi|$.

The third corollary combines the theorem with the separate-decisions property to show the following.

Corollary 3: For all $R \in \mathscr{R}_\phi$, v_ϕ is constant over R.

Proof: Let $(x, y), (x', y')$ be two inputs in a rectangle $R \in \mathscr{R}_\phi$. From the theorem, $\langle b_j(x, y) \rangle_{j=1}^{\infty} = \langle b_j(x', y) \rangle_{j=1}^{\infty} = \langle b_j(x', y') \rangle_{j=1}^{\infty}$. By the separate-decisions property, $v_\phi(x, y) = v_\phi(x', y) = v_\phi(x', y')$. Q.E.D.

In summary, the three corollaries show that every deterministic protocol ϕ determines

1) a partition $\mathscr{R}_\phi$ of $\{0,\cdots,n-1\}$ into rectangles. We let $R_\phi(x, y)$ denote the unique rectangle in $\mathscr{R}_\phi$ that contains (x, y);
2) a mapping b_ϕ: $\mathscr{R}_\phi \to \{0,1\}^*$ where $b_\phi(R)$ for $R \in \mathscr{R}_\phi$ is the unique codeword that ϕ associates with all inputs in R. That is, for all $(x, y) \in \{0,\cdots, n-1\} \times \{0,\cdots, n-1\}$, $b_\phi(x, y) = b_\phi(R_\phi(x, y))$. The mapping b_ϕ must satisfy several requirements; most germane to our purposes is that $\{b_\phi(R)$: $R \in \mathscr{R}_\phi\}$ is prefix-free with cardinality $|\mathscr{R}_\phi|$, with the length of $b_\phi(R)$ denoted by $l_\phi(R)$;
3) a mapping v_ϕ: $\mathscr{R}_\phi \to \{0,1\}$ where $v_\phi(R)$, for $R \in \mathscr{R}_\phi$, is the unique value that ϕ assigns to all the inputs (x, y) in R. That is, $v_\phi(x, y) = v_\phi(R_\phi(x, y))$ for all $(x, y) \in \{0,\cdots,n-1\} \times \{0,\cdots,n-1\}$.

Before proceeding to prove the bounds, we need a lemma from information theory. A probability distribution is a set of nonnegative reals that add to one. To allow multiple equal probabilities, we regard probability distributions as multisets, but for lack of alternative notation we denote them, as sequences, by angled brackets. The *entropy* of a probability distribution $\langle p(x)$: $x \in S\rangle$ is defined as

$$H\langle p(x): x \in S\rangle \overset{\text{def}}{=} \sum_{x \in S} p(x) \log \frac{1}{p(x)}.$$

The next information-theoretic lemma says that the number of bits needed to describe the outcome of a random variable is at least the entropy of the underlying probability distribution.

Lemma 1[8]: Let $\langle p_j$: $j=1,\cdots,J\rangle$ be a probability distribution and $\langle l_j$: $j=1,\cdots,J\rangle$ be the lengths of codewords in a prefix-free code.[1] Then,

$$\sum_{j=1}^{J} p_j l_j \geq H\langle p_j: j=1,\cdots,J\rangle.$$

B. Lower Bound on $\hat{C}_R(f, 0)$

In this section we prove that for most functions, however sparse, Las Vegas is not much better than determinism when worst case complexity is the measure. Slightly more precisely, for all $s \geq n$, most functions in $\mathscr{F}_s$ have $\hat{C}_R(f,0) \geq \log n - 4$.

As mentioned in Section III, every randomized protocol can be regarded as a collection of deterministic protocols; the randomness is confined to the choice of the protocol. Given any error-free deterministic protocol ϕ and any input (x, y), an error-free deterministic protocol ϕ' exists that exchanges only two bits when the input is (x, y) and at most two bits more than ϕ for every other input. It is therefore not clear *a priori* that there must be an input with a large expected communication $(\bar{L}_\Phi(x, y))$. We prove the lower bound by demonstrating a set S of inputs over which every deterministic protocol must exchange a large number of bits *on the average*.

We define the average complexity of a protocol Φ over a set $S \subseteq \{0,\cdots,n-1\} \times \{0,\cdots,n-1\}$ to be

$$\bar{L}_\Phi^S \overset{\text{def}}{=} \frac{1}{|S|} \sum_{(x, y) \in S} \bar{L}_\Phi(x, y).$$

The definition applies of course also to deterministic protocols. If ϕ is deterministic, $\mathscr{R}_\phi$ (defined in Section V-A) induces a partition $S_1,\cdots, S_J$ of S such that each S_j is the intersection of S with some rectangle $R_j \in \mathscr{R}_\phi$. All inputs in each S_j have the same codeword—$b_\phi(R_j)$—whose length we denote by l_j. The next lemma relates $\bar{L}_\Phi^S$ to the sizes of the S_j.

Lemma 2: If S is partitioned by $\mathscr{R}_\phi$ into $S_1,\cdots, S_J$, then

$$\bar{L}_\Phi^S \geq H\langle |S_j|/|S|: j=1,\cdots,J\rangle.$$

Proof:

$$\begin{aligned}\bar{L}_\Phi^S &\overset{\text{def}}{=} \sum_{(x, y) \in S} \frac{1}{|S|} l_\phi(x, y)\\ &= \sum_{R \in \mathscr{R}_\phi} \sum_{(x, y) \in R \cap S} \frac{1}{|S|} l_\phi(x, y)\\ &= \sum_{j=1}^{J} \sum_{(x, y) \in S_j} \frac{1}{|S|} l_j\\ &= \sum_{j=1}^{J} l_j \frac{|S_j|}{|S|},\end{aligned}$$

but, from Lemma 2, $l_1,\cdots, l_J$ are lengths of codewords in a prefix-free code. Hence by Lemma 1

$$\sum_{j=1}^{J} l_j \frac{|S_j|}{|S|} \geq H\langle |S_j|/|S|: j=1,\cdots,J\rangle. \quad \text{Q.E.D.}$$

The rest of the proof applies this lemma to most functions in $\mathscr{F}_s$, for every $n \leq s \leq n^2/2$. We divide the range $n \leq s \leq n^2/2$ into three intervals: $n \leq s < 2n \log n$, $2n \log n \leq s < n^2/8$, and $n^2/8 \leq s \leq n^2/2$. The larger s, the more likely are functions in $\mathscr{F}_s$ to have high complexity. Therefore, the most interesting interval to prove is $n \leq s \leq 2n \log n$. We only prove this interval here and mention the stronger results pertaining to the other intervals of s in Corollary 4.

The closure of a set T was defined in Section V-A to be

$$\text{Cl}(T) \overset{\text{def}}{=} \{(x, y)\text{: for some } u \text{ and } v \text{ in } \{0,\cdots,n-1\},\ \text{both } (x, u) \text{ and } (v, y) \text{ are in } T\}.$$

If $S \subseteq \{0,\cdots,n-1\} \times \{0,\cdots,n-1\}$, we let λ_f^S be the size of the largest subset T of S such that f is constant over $\text{Cl}(T)$:

$$\lambda_f^S \overset{\text{def}}{=} \max\{|T|: T \subseteq S, \text{ and } (x_1, y_1), (x_2, y_2) \in \text{Cl}(T) \Rightarrow f(x_1, y_1) = f(x_2, y_2)\}.$$

[1] A set of strings is traditionally called a code book or a code. Its elements are called codewords.

(Note that $\mathrm{Cl}(T)$ does not have to be contained in S.) The next theorem uses λ_f^S to upper-bound the size of each S_j; then it uses Lemma 2 to lower-bound $\hat{C}_R(f,0)$.

Theorem 2: $\hat{C}_R(f,0) \geq \log|S|/\lambda_f^S$ for all $S \subseteq \{0,\cdots,n-1\} \times \{0,\cdots,n-1\}$.

Proof: Let S be any set in $\{0,\cdots,n-1\} \times \{0,\cdots,n-1\}$, Φ be any error-free randomized protocol, and ϕ be any deterministic protocol in Φ that occurs with nonzero probability.

Consider the partition $S_1,\cdots,S_J$ of S induced by $\mathscr{R}_\phi$. By definition, each S_j is contained in one rectangle of $\mathscr{R}_\phi$; hence $\mathrm{Cl}(S_j)$ is also contained in that rectangle. From Corollary 3, $v_\phi(x,y)$ must be constant over $\mathrm{Cl}(S_j)$. However, since Φ is error-free and ϕ occurs with positive probability, ϕ must be error-free too. Hence $f(x,y)$ must be constant over $\mathrm{Cl}(S_j)$. This implies that $|S_j| \leq \lambda_f^S$. A simple calculation can show that if $|S_j| \leq \lambda_f^S$ for all j, then $H\langle |S_j|/|S|\colon\ \underline{j}=1,\cdots,J\rangle \geq \log(|S|/\lambda_f^s)$. Therefore, by the last lemma, $\overline{L}_\phi^S \geq \log(|S|/\lambda_f^S)$. This is true for all positive-probability protocols $\phi \in \Phi$; therefore, $\overline{L}_\Phi^S \geq \log(|S|/\lambda_f^S)$. By definition of $\overline{L}_\Phi^S$ as an average, there must be an input $(x,y) \in S$ such that $\overline{L}_\Phi(x,y) \geq \log(|S|/\lambda_f^S)$. Therefore, for every error-free protocol Φ, $\hat{L}_\Phi \geq \log(|S|/\lambda_f^S)$ and the theorem follows. Q.E.D.

The next lemma uses a counting argument to show that, for s close to n, most functions in $\mathscr{F}_s$ contain a large set S with $\lambda_f^S \leq 2$.

Lemma 3: For all $s \geq n \geq 8$, a fraction larger than $1-2(s^9/n^{12})$ of the functions in $\mathscr{F}_s$ contain a set $S \subseteq \mathbf{1}_f$ of size $n/8$ such that for all subsets T of S, $|T| \geq 3$ implies that $\mathrm{Cl}(T) \not\subseteq \mathbf{1}_f$.

Proof: We define a *square* to be a product set of the form $A_x \times A_y$ where A_x, $A_y \subseteq \{0,\cdots,n-1\}$ are of equal size. We call $|A_x|$ (equivalently, $|A_y|$) the *side* of the square. The proof proceeds with two claims.

Claim 1: The fraction of functions f in $\mathscr{F}_s$ for which $\mathbf{1}_f$ contains a square of side 3 is at most s^9/n^{12}.

Proof of Claim 1: Fix a 3×3 square in $\{0,\cdots,n-1\} \times \{0,\cdots,n-1\}$. The number of sets of sizes s (thought of as $\mathbf{1}_f$) that contain this square, is $\binom{n^2-9}{s-9}$. Hence the number of sets of size s that contain any 3×3 square is at most $\binom{n}{3}\binom{n}{3}\binom{n^2-9}{s-9}$. The fraction of functions in $\mathscr{F}_s$ such that $\mathbf{1}_f$ contains a 3×3 rectangle is, therefore, at most

$$\binom{n}{3}\binom{n}{3}\binom{n^2-9}{s-9}\bigg/\binom{n^2}{s} \leq \frac{s^9}{n^{12}}.$$

Claim 2: For all $s \geq n$, the fraction of functions f in $\mathscr{F}_s$ such that $\mathbf{0}_f$ contains a square of side $\geq 7n/8$ is at most $1/2^n$.

Proof of Claim 2: As before, fix a square of side $7n/8$. The number of sets of size n^2-s (thought of as $\mathbf{0}_f$) that contain this square is $\binom{n^2-(7n/8)^2}{s}$. Therefore, the number of sets of size n^2-s that contain any square of side $7n/8$ is at most $\binom{n^2-(7n/8)^2}{s}\binom{n}{(7n/8)}^2$. Hence the fraction of functions in $\mathscr{F}_s$ such that $\mathbf{0}_f$ contains a square of side $\geq 7n/8$ is at most

$$\frac{\binom{n^2-(7n/8)^2}{s}\binom{n}{(7n/8)}^2}{\binom{n^2}{s}}$$

$$\leq \left(\frac{n^2-(7n/8)^2}{n^2}\right)^s \binom{n}{(7n/8)}^2$$

$$\leq \left(\frac{15}{64}\right)^s 1.458^{2n} < \frac{1}{2^n}.$$

Combined, the two claims imply that for a fraction of at least $1-0.5^n-(s^9/n^{12}) \geq 1-2(s^9/n^{12})$ of the functions f in $\mathscr{F}_s$, the set $\mathbf{0}_f$ does not contain a square of side $7n/8$, and the set $\mathbf{1}_f$ does not contain a square of side 3.

To complete the proof, we need one more definition. Call a set S *diagonal* if no two of its elements are in the same row or column. (That is, (x,y), $(x',y') \in S$ implies $x \neq x'$ and $y \neq y'$). It is easy to see that if $\mathbf{0}_f$ does not contain a square of side k $(1 \leq k \leq n)$, then $\mathbf{1}_f$ contains a diagonal set of size $n-k+1$.

We can therefore deduce that for a fraction of at least $1-2(s^9/n^{12})$ of the functions f in $\mathscr{F}_s$, the set $\mathbf{1}_f$ contains a diagonal set D_f of size $n/8$ but does not contain a square of side 3. D_f is clearly the desired set because if $T \subseteq D_f$ and $|T| \geq 3$ then $\mathrm{Cl}(T)$ contains a square of side 3. This implies that $\mathrm{Cl}(T) \not\subseteq \mathbf{1}_f$. Q.E.D.

The claimed bounds can now be proved easily:

Corollary 4: For any $s \geq n$, a fraction of at least $1-2[(2\log n)^9/n^3]$ of the functions in $\mathscr{F}_s$ have $\hat{C}_R(f,0) \geq \log n - 4$.

Proof: If $n \leq 16$ the claim is trivially true. Otherwise, as mentioned before, we divide the proof into three intervals of s: $n \leq s < 2n\log n$, $2n\log n \leq s < n^2/8$, and $n^2/8 \leq s \leq n^2/2$. We prove the result for the first interval and state the (stronger) results that can be proved for the others.

$n \leq s < 2n \log n$: A fraction of at least $1-2[(2\log n)^9/n^3]$ of the functions in $\mathscr{F}_s$ have $\hat{C}_R(f,0) \geq \log n - 4$.

Proof: According to Lemma 3, a fraction of at least $1-2[(2\log n)^9/n^3]$ of the functions in $\mathscr{F}_s$ contain a set S of size $n/8$ with $\lambda_f^S \leq 2$. Theorem 2 ensures that for all these functions, $\hat{C}_R(f,0) \geq \log[(n/8)/2] = \log n - 4$.

$2n\log n < s \leq n^2/8$: A fraction of at least $1-2/n^3$ of the functions in $\mathscr{F}_s$ have $\hat{C}_R(f,0) \geq \log n - 2$.

$n^2/8 \leq s \leq n^2/2$: A fraction of at least $1-0.365^n \cdot n^8$ of the functions in $\mathscr{F}_s$ have $\hat{C}_R(f,0) \geq \log n - 2$. Q.E.D.

Example 1: Let $d_H^N(x,y)$ denote the Hamming distance between the two N-bit sequences x and y. For $N \geq 0$ and $0 \leq k \leq N$, define the function H_k^N: $\{0,1\}^N \times \{0,1\}^N \to$

$\{0,1\}$ by

$$H_k^N(x,y) \overset{\text{def}}{=} \begin{cases} 1, & \text{if } d_H^N(x,y)=k \\ 0, & \text{otherwise.} \end{cases}$$

To determine the worst case Las Vegas complexity of H_k^N, consider $\mathbf{1}_{H_k^N}$ (the set of inputs for which H_k^N is 1). Its size is $2^N\binom{N}{k}$. Yet, [9] showed that for $0 \le k \le \lceil N/2-\sqrt{N/4}\,\rceil$, the largest rectangle contained in $\mathbf{1}_{H_k^N}$ is of size $\binom{N}{k}$. Taking S of Theorem 2 to be the union of $\mathbf{1}_{H_k^N}$ and any $\binom{N}{k}$ element subset of $\mathbf{0}_{H_k^N}$, we obtain $\hat{C}_R(H_k^N,0) \ge \lceil \log(2^N+1)\rceil = N+1$. This clearly implies that for all k's in the above range, $\hat{C}_R(H_k^N,0)=N+1$.

When n is a power of 2, the equality function is equivalent to H_0^N. Therefore, $\hat{C}_R(\text{equ}, 0) = N+1 = \log n + 1$.

C. Upper Bound on $\overline{C}_D(f,0)$ and $\hat{C}_R(f,\hat{\epsilon})$

The upper bound proved is stronger than described in Section IV. It bounds the complexity in terms of the distribution of the ones and zeros (rather than just their number). For "non-regular" distributions it is even lower than $\log(|\mathbf{1}_f|/n)$.

For notational convenience we define $\log 0$ to be -1 and define $G(f)$ as follows:

$$G_{Y|X}(f) \overset{\text{def}}{=} 1 + \frac{1}{n}\min\left(\sum_{x=0}^{n-1}\log|\{y\colon f(x,y)=1\}|,\ \sum_{x=0}^{n-1}\log|\{y\colon f(x,y)=0\}|\right)$$

$$G_{X|Y}(f) \overset{\text{def}}{=} 1 + \frac{1}{n}\min\left(\sum_{y=0}^{n-1}\log|\{x\colon f(x,y)=1\}|,\ \sum_{y=0}^{n-1}\log|\{x\colon f(x,y)=0\}|\right)$$

$$G(f) \overset{\text{def}}{=} \min\left(G_{Y|X}(f), G_{X|Y}(f)\right).$$

We show that for *all* functions, $\overline{C}_R(f,\hat{\epsilon}) \le (1-2\epsilon)(G(f) + 6.3\log\log n + 2\log(1/\epsilon) + c)$ and use this result to establish the upper bounds claimed in the introduction.

One interpretation of this bound is that each row (column) in the function table contributes about log of the number of ones (zeros) it contains to the complexity of the function. If a function has about the same number of ones in each row, then a protocol that requires about twice that many bits in the worst case can be obtained by generalizing and refining the equality-function protocol of [10] (see [2]).

Lemma 4: Let $n \ge 2$, $m \ge 1$, and $f\colon \{0,\cdots,n-1\} \times \{0,\cdots,n-1\} \to \{0,1\}$ satisfy $|\{y\colon f(x,y)=1\}| \le m$ for all $x \in \{0,\cdots,n-1\}$. Then, for all $0 < \epsilon \le 1/2$,

$$\hat{C}_R(f,\hat{\epsilon}) \le 2(\log m + \log\ln n + \log(1/\epsilon) + 1).$$

Proof: The trivial protocol Φ that guesses with probability 2ϵ and transmits all bits otherwise has $\hat{E}_\Phi(f) \le \epsilon$ and $\hat{L}_\Phi \le 1 + 2\epsilon + (1-2\epsilon)(\log n + 2) = 3 - 2\epsilon + (1-2\epsilon)\log n$. For $n \le 60\,000$, this is less than the upper bound for all ϵ. Thus, from here on, we assume that $n > 60\,000$.

For every $x \in \{0,\cdots,n-1\}$, let $m(x) \overset{\text{def}}{=} |\{y\colon f(x,y)=1\}|$ and for $i=1,\cdots,m(x)$, let $y_i(x)$ be the ith smallest integer in $\{y\colon f(x,y)=1\}$. Clearly, for all $x \in \{0,\cdots,n-1\}$, $m(x) \le m$ and $\{y_1(x),\cdots,y_{m(x)}(x)\} = \{y\colon f(x,y)=1\}$.

Let $\sigma \overset{\text{def}}{=} \sqrt{2}(m\ln n/\epsilon)$. Then $n > 60\,000$ implies $\sigma > 31.1$ so the number of primes between σ and 2σ is at least $\sigma/(\sqrt{2}\ln\sigma)$ and at most $\sigma/\ln\sigma$. That is, $\sigma/(\sqrt{2}\ln\sigma) < \Pi(2\sigma)-\Pi(\sigma) < \sigma/\ln\sigma$ where $\Pi(x)$ is the number of primes not exceeding x. The number of primes between σ and 2σ that divide any positive integer $\le n$ is always at most $\log_\sigma n$.

For brevity, let $(x)_\alpha$ denote $x \bmod \alpha$. We show that the following protocol, denoted Φ_1, achieves the claimed bound.

1) P_y picks at random a prime α such that $\sigma < \alpha < 2\sigma$. He transmits α and $(y)_\alpha$ to P_x.
2) P_x computes and transmits
$$g(x,\alpha,(y)_\alpha) \overset{\text{def}}{=} \begin{cases} 1, & \text{if } (y)_\alpha = (y_i(x))_\alpha \\ & \text{for some } 1 \le i \le m(x) \\ 0, & \text{otherwise.} \end{cases}$$
3) Both P_x and P_y accept $g(x,\alpha,(y)_\alpha)$ as the value of $f(x,y)$.

Now, if $f(x,y)=1$, then $y=y_i(x)$ for some $1 \le i \le m(x)$, thus $g(x,\alpha,(y)_\alpha)=1$. Hence $P(V_{\Phi_1}(x,y)=0|(x,y)\in\mathbf{1}_f)=0$. On the other hand,

$$\begin{aligned} P\left(V_{\Phi_1}(x,y)=1|(x,y)\in\mathbf{0}_f\right) &= P\left(\exists i \in \{1,\cdots,m(x)\} \text{ such that } (y_i(x))_\alpha = (y)_\alpha | y \ne y_j(x) \text{ for all } j \in \{1,\cdots,m(x)\}\right) \\ &\le \sum_{i=1}^{m(x)} P\left(y - y_i(x) \equiv 0 \ (\mathrm{mod}\ \alpha) | y - y_i(x) \ne 0\right) \\ &= \sum_{i=1}^{m(x)} \frac{\text{number of primes between } \sigma \text{ and } 2\sigma \text{ dividing the nonzero integer } (y-y_i(x))}{\text{number of primes between } \sigma \text{ and } 2\sigma} \\ &\le \sum_{i=1}^{m(x)} \log_\sigma n \Big/ \frac{\sigma}{\sqrt{2}\ln\sigma} \\ &\le \sqrt{2}\,m(\ln n/\sigma) = \epsilon. \end{aligned}$$

The average number of bits transmitted is upper-bounded by

- transmitting α: $\log(\sigma/\ln\sigma) \le \log m + \log\ln n + \log(1/\epsilon) + 0.5 - \log\ln 31.1$ bits;
- transmitting $y \bmod \alpha$: $\log 1.7\sigma \le \log m + \log\ln n + \log(1/\epsilon) + 0.5 + 0.77$ bits;
- rounding off the above: 1 bit;
- transmitting $g(x, \alpha, y \bmod \alpha)$: 1 bit.

Hence the total communication is at most $2(\log m + \log\ln n + \log(1/\epsilon) + 1)$ bits. Q.E.D.

In the next example we use the lemma to improve the bound on a function defined in [5].

Example 2: Let $M \ge 0$, and $x^1, \cdots, x^M$, $y^1, \cdots, y^M \in \{0,1\}^M$ (we index the sequences with superscripts because subscripts were used earlier to denote a bit in a sequence). If $n = 2^{M^2}$, then each $(x^1, \cdots, x^M)$ can be identified with an integer in $\{0, \cdots, n-1\}$. The *component equality* (CE) function is defined as

$$\mathrm{CE}\big((x^1, \cdots, x^M), (y^1, \cdots, y^M)\big) \overset{\mathrm{def}}{=} \begin{cases} 1, & \text{if } x^i = y^i \text{ for some } 1 \le i \le M \\ 0, & \text{otherwise.} \end{cases}$$

It was shown in [5] that $\hat{C}_R(f,0) \le O(M \log^2 M)$. By choosing $\epsilon = \log M/M$ in the last lemma, and proceeding as in the original proof, we get, $\hat{C}_R(f,0) \le 5M\log M + c$.

However, to meet the lower bound, Lemma 4 needs to be improved even for functions with equally dense rows. The reason is that the protocol uses about as many bits describing the results of the random experiments as it does describing the values. One way around this, is using less randomization. The following lemma, proven in [4], shows that in the initial phases of the communication, deterministic protocols can be efficient.

Lemma 5: Let $n \ge 2$, $m \ge 1$, and f: $\{0, \cdots, n-1\} \times \{0, \cdots, n-1\} \to \{0,1\}$ satisfy

$$|\{y\colon f(x,y) = 1\}| \le m \text{ for all } x \in \{0, \cdots, n-1\}.$$

Then, there exists a partition $B_1, \cdots, B_{\lceil cm/(\ln n)^{1.1}\rceil}$ of $\{0, \cdots, n-1\}$ such that for $x = 0, \cdots, n-1$ and $j = 1, \cdots, \lceil cm/(\ln n)^{1.1}\rceil$, $|\{y \in B_j\colon f(x,y) = 1\}| < (\ln n)^{1.1}$ where c is a constant independent of n, m, and f.

Combined, Lemmas 4 and 5 yield the following.

Lemma 6: For all $n \ge 2$, $0 < \epsilon \le 1/2$, and f: $\{0, \cdots, n-1\} \times \{0, \cdots, n-1\} \to \{0,1\}$,

$$\bar{C}_R(f, \hat{\epsilon}) \le G(f) + 4.1 \log\log n + 2\log\frac{1}{\epsilon} + c$$

where c is a constant independent of n, ϵ, and f.

Proof: Without loss of generality assume that

$$G(f) = G_{Y|X}(f) = \frac{1}{n}\sum_{x=0}^{n-1} \log|\{y\colon f(x,y) = 1\}|.$$

For $i = -1, 0, 1, \cdots, \lceil \log n\rceil$ define $A_i \overset{\mathrm{def}}{=} \{x\colon\ i - 1 < \log m(x) \le i\}$. Clearly, the sets A_{-1}, A_0, $A_1, \cdots, A_{\lceil\log n\rceil}$ partition $\{0, \cdots, n-1\}$.

By Lemma 5, for $i = \lceil\log(\ln n)^{1.1}\rceil, \cdots, \lceil\log n\rceil$, there exists a partition $\{B_{i,j}\colon\ j = 1, \cdots, \lceil c(2^i/(\ln n)^{1.1})\rceil\}$ of $\{0, \cdots, n-1\}$ such that for all $x \in A_i$ and all $j \in \{1, \cdots, \lceil c(2^i/(\ln n)^{1.1})\rceil\}$, $|\{y \in B_{ij}\colon f(x,y) = 1\}| < (\ln n)^{1.1}$ where c is a constant.

P_x and P_y agree on such a collection of partitions and conduct a protocol Φ_2 defined as follows

1) P_x transmits $\lceil\log m(x)\rceil$ to P_y (thereby telling him of the index of the set A_i that x is in).

2) If $\lceil\log m(x)\rceil = -1$, they decide that the value of the function is zero and stop. If $-1 < \lceil\log m(x)\rceil < \lceil\log(\ln n)^{1.1}\rceil$, they move to step 3. Otherwise, P_y transmits the index j of the set $B_{\lceil\log m(x)\rceil, j}$ that contains y.

From now on, they know that (x, y) is in the generalized rectangle $A_{\lceil\log m(x)\rceil} \times B_{\lceil\log m(x)\rceil, j}$. This rectangle has the property that each of its rows contains at most $(\ln n)^{1.1}$ elements of $\mathbf{1}_f$. That is, $|\{y \in B_{\lceil\log m(x)\rceil, j}\colon f(x,y) = 1\}| < (\ln n)^{1.1}$ for all x in $A_{\lceil\log m(x)\rceil}$.

3) They use the protocol Φ_1 of Lemma 4 to find $f(x, y)$ over this rectangle with probability of error $\le \epsilon$.

The number of bits transmitted is upper-bounded by the following steps.

Step 1 $\lceil\log(\log n + 2)\rceil$.
Step 2 0 bits if $m(x) < (\ln n)^{1.1}$ and $\lceil\log\lceil c(2\lceil\log m(x)\rceil/(\ln n)^{1.1})\rceil\rceil$
$\le \lceil\log(c \cdot 2m(x)/(\ln n)^{1.1})\rceil$, otherwise.
Step 3 0 bits if $m(x) = 0$, $2(\log m(x) + \log\ln n + \log(1/\epsilon) + 1)$ if $0 < m(x) < (\ln n)^{1.1}$, and $2(\log(\ln n)^{1.1} + \log\ln n + \log(1/\epsilon) + 1)$, otherwise.

Hence for every (x, y), the total number of bits transmitted is at most $\log m(x) + 2\log(1/\epsilon) + 4.1\log\ln n + c$ (where c is a new constant). The average, therefore, is $\le (1/n)\sum_{x=0}^{n-1} \log m(x) + 2\log(1/\epsilon) + 4.1\log\ln n + c = G(f) + 2\log(1/\epsilon) + 4.1\log\ln n + c$. Q.E.D.

Decreasing ϵ in the lemma increases the complexity only moderately, so if ϵ is large, some bits can be saved by incurring an error smaller than ϵ most of the time and just guessing the result in the rest, yielding the following.

Theorem 3: For all $n \ge 2$, $0 < \epsilon \le 1/2$, and all f: $\{0, \cdots, n-1\} \times \{0, \cdots, n-1\} \to \{0,1\}$,

$$\bar{C}_R(f, \hat{\epsilon}) \le 2 + (1 - 2\epsilon) \cdot (G(f) + 6.3\log\log n - 2\log(1/\epsilon) + c)$$

where c is a constant independent of n, ϵ, and f.

Proof: Let $\delta \overset{\mathrm{def}}{=} \min(\epsilon, (1/G(f)), 1/101)$ and $\epsilon' \overset{\mathrm{def}}{=} (\epsilon - \delta)/(1/2 - \delta)$. The protocol Φ_3 is defined as follows

1) P_y performs a random experiment that is 1 with probability ϵ' and transmits the outcome to P_x.

2) If the outcome of the experiment is 1, P_y performs an unbiased binary experiment and transmits the result to P_x. They both accept the outcome of this experiment as the value of $f(x, y)$. If the outcome of the experiment in the first phase is 0 (probability $1 - \epsilon'$), they use protocol Φ_2

(Lemma 6) to compute $f(x, y)$ with probability of error $\leq \delta$.

The total probability of error is $\leq \epsilon'/2+(1-\epsilon')\delta = \epsilon'((1/2)-\delta)+\delta=\epsilon$. The average communication length is

$$\bar{L}_{\Phi_3}=1+\epsilon'+(1-\epsilon')$$
$$\cdot(G(f)+4.1\log\log n+2\log(1/\delta)+c)$$
$$\leq 2+(1-2\epsilon)(G(f)+6.3\log\log n+2\log(1/\epsilon)+c).$$

Q.E.D.

By the convexity of the log function, $G(f)\leq \log(|\mathbf{1}_f|/n)$. We have therefore proved the upper bound for the average case.

Corollary 5: For all $n\geq 2$, $0<\epsilon\leq 1/2$, and all functions f: $\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}\rightarrow\{0,1\}$,

$$\bar{C}_R(f,\hat{\epsilon})\leq 2+(1-2\epsilon)\big(\log\big(|\mathbf{1}_f|/n\big)+6.3\log\log n$$
$$+2\log(1/\epsilon)+c\big).$$

For all $s\geq n$, most functions in $\mathscr{F}_s$ have about the same number of ones in every row and column. A quick analysis of Lemma 6 can show that the expected number of bits transmitted for every input is about the same (and, that step 1 can be skipped). Hence we have the following.

Corollary 6: Let $s\geq n\geq 2$ and $0<\epsilon\leq 1/2$. For most functions f in $\mathscr{F}_s$,

$$\hat{C}_R(f,\hat{\epsilon})\leq 2+(1-2\epsilon)\big(\log\big(|\mathbf{1}_f|/n\big)$$
$$+5.3\log\log n+2\log(1/\epsilon)+c\big).$$

The protocol of Theorem 3 can incur an error only when its computed value is one. This happens with probability $\leq\epsilon+(|\mathbf{1}_f|/n^2)$. Verifying the result whenever that is the case, we obtain a randomized Las Vegas protocol with $\bar{L}_\Phi\leq\log(|\mathbf{1}_f|/n)+8.3\log\log n+c$. This, in turn, implies the existence of a deterministic protocol with the same average length. Thus we have the next corollary.

Corollary 7: For *all* functions f: $\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}\rightarrow\{0,1\}$,

$$\bar{C}_D(f,0)\leq\log\big(|\mathbf{1}_f|/n\big)+8.3\log\log n+c.$$

Proof: If $|\mathbf{1}_f|/n^2>1/(\log n)^{8.3}$, then $\log(|\mathbf{1}_f|/n)+8.3\log\log n+c\geq\log n+c$. For $c\geq 2$, this is more than the complexity of the trivial protocol. Else, let $\epsilon\overset{\text{def}}{=}1/\log n$. Corollary 5 says that $\bar{C}_R(f,\hat{\epsilon})\leq\log|\mathbf{1}_f|/n+8.3\log\log n+c$. The protocol used to derive this bound (Φ_3) errs only when its computed value is one. Assuming a uniform probability distribution over the inputs, this happens with probability $\leq\epsilon+(|\mathbf{1}_f|/n^2)\leq 2/\log n$. By verifying the result whenever this is the case, we add at most 2 bits on the average. Therefore, $\bar{C}_D(f,0)=\bar{C}_R(f,0)\leq\log(|\mathbf{1}_f|/n)+8.3\log\log n+c$. Q.E.D.

Example 3: Consider H_k^N again. Example 1 showed that for $0\leq k\leq\lceil N/2-\sqrt{N/4}\rceil$, $\hat{C}_R(H_k^N,0)\geq\log N$. The following protocol, whose main advantage is amenability to simple analysis, shows that the average complexity of H_k^N is much smaller.

First, P_x transmits $x_1,\cdots,x_k$. Then, starting with $i=k+1$ and continuing with consecutive i's, P_x transmits x_i and P_y responds by transmitting 1 if the Hamming distance between $(x_1,\cdots,x_i)$ and $(y_1,\cdots,y_i)$ is $k+1$, and transmitting 0, otherwise. To the Nth bit, P_y responds by transmitting the value of H_k^N. They stop after P_y either transmits a one or responds to x_N. Let $i_j\overset{\text{def}}{=}\min\{\{i: d_H((x_1,\cdots,x_i),(y_1,\cdots,y_i))=j\}\cup\{N\}\}$ and let I_j be the random variable whose value is i_j. If L is the random variable denoting the length of the communication then, $L=k+2(I_{k+1}-k)=2I_{k+1}-k$, so the expected value of L is $\bar{L}=2\bar{I}_{k+1}-k$.

The expected value of I_{k+1} can be determined via a tedious calculation or by using an old trick. For $j=1,\cdots,k+1$, let $D_j\overset{\text{def}}{=}I_j-I_{j-1}$ (I_0 is defined to be zero). Then each D_j is distributed geometrically with parameter 1/2. Therefore, the expected value of each D_j is 2, and the expected value of I_{k+1} is $E(I_{k+1})=E(\Sigma_{j=1}^{k+1}D_j)=\Sigma_{j=1}^{k+1}2=2(k+1)$. Hence $\bar{L}=3k+4$.

Note that this protocol achieves a much smaller complexity than that promised by Corollary 7. The next subsection, however, shows that this is a rare exception. For most functions, the upper bound of the corollary is quite tight. The reason for the discrepancy here is that $\mathbf{0}_{H_k^N}$ contains unusually large generalized rectangles.

D. Lower Bound on $\bar{C}_R(f,\hat{\epsilon})$

In Section V-B we defined the average complexity of a protocol over a set $S\subseteq\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}$. We now define the average error of Φ over S to be

$$\bar{E}_\Phi^S(f)\overset{\text{def}}{=}\frac{1}{|S|}\sum_S\bar{E}_\Phi(x,y)$$
$$=\frac{1}{|S|}\sum_S P\big(V_\Phi(x,y)\neq f(x,y)\big).$$

The average error of Φ, $\bar{E}_\Phi(f)$, is defined to be the arithmetic average of the error over the zeros and the errors over the ones: $\bar{E}_\Phi(f)\overset{\text{def}}{=}(\bar{E}_\Phi^{\mathbf{0}_f}(f)+\bar{E}_\Phi^{\mathbf{1}_f}(f))/2$.

Remark: In general, $\bar{E}_\Phi(f)$ can be taken as the maximum of $\bar{E}_\Phi^{\mathbf{0}_f}(f)$ and $\bar{E}_\Phi^{\mathbf{1}_f}(f)$, or any weighted average of the two. Here we chose half–half weights. Other weights, though possible, yield an inferior lower bound.

We can now define average-error complexity in much the same way we did in the introduction. Two measures prove particularly useful. The *average deterministic complexity with ϵ average error* is defined as

$$\bar{C}_D(f,\bar{\epsilon})\overset{\text{def}}{=}\min\big\{\bar{L}_\phi:\ \phi\text{ is deterministic}$$
$$\text{and }\bar{E}_\Phi(f)\leq\epsilon\big\},$$

and the *average randomized complexity* with ϵ average error is defined as

$$\bar{C}_R(f,\bar{\epsilon})\overset{\text{def}}{=}\min\big\{\bar{L}_\Phi:\ \bar{E}_\Phi(f)\leq\epsilon\big\}.$$

The next lemma shows that, although the average error is defined in a peculiar way, ϵ average error is still easier to achieve than ϵ worst error.

Lemma 7: For all functions f and $\epsilon > 0$, $\bar{C}_R(f, \hat{\epsilon}) \geq \bar{C}_R(f, \bar{\epsilon})$.

Proof:

$$\bar{E}_\Phi^{\mathbf{0}_f}(f) \overset{\text{def}}{=} \frac{1}{|\mathbf{0}_f|} \sum_{(x,y) \in \mathbf{0}_f} \bar{E}_\Phi(x,y)$$
$$\leq \max\left\{ \bar{E}_\Phi(x,y) : (x,y) \in \mathbf{0}_f \right\}$$

and

$$\bar{E}_\Phi^{\mathbf{1}_f}(f) \overset{\text{def}}{=} \frac{1}{|\mathbf{1}_f|} \sum_{(x,y) \in \mathbf{1}_f} \bar{E}_\Phi(x,y)$$
$$\leq \max\left\{ \bar{E}_\Phi(x,y) : (x,y) \in \mathbf{1}_f \right\}.$$

Therefore,

$$\begin{aligned}\bar{E}_\Phi(f) &= \left(\bar{E}_\Phi^{\mathbf{0}_f}(f) + \bar{E}_\Phi^{\mathbf{1}_f}(f)\right)/2 \\ &\leq \max\left(\bar{E}_\Phi^{\mathbf{0}_f}(f),\ \bar{E}_\Phi^{\mathbf{1}_f}(f)\right) \\ &\leq \max\Big(\max\left\{\bar{E}_\Phi(x,y) : (x,y) \in \mathbf{0}_f\right\}, \\ &\qquad \max\left\{\bar{E}_\Phi(x,y) : (x,y) \in \mathbf{1}_f\right\}\Big) \\ &= \max\big\{\bar{E}_\Phi(x,y) : (x,y) \\ &\qquad \in \{0,\cdots,n-1\} \times \{0,\cdots,n-1\}\big\} \\ &= \hat{E}_\Phi(f). \end{aligned}$$

Q.E.D.

Any lower bound for $\bar{C}_R(f, \bar{\epsilon})$ is therefore also a lower bound for $\bar{C}_R(f, \hat{\epsilon})$. It then suffices to prove the result for average error. We begin by proving a lower bound for the deterministic complexity $\bar{C}_D(f, \bar{\epsilon})$.

Each rectangle $R \subseteq \{0,\cdots,n-1\} \times \{0,\cdots,n-1\}$ contains a certain proportion $\rho_f(R)$ of ones defined as

$$\rho_f(R) \overset{\text{def}}{=} \frac{|R \cap \mathbf{1}_f|}{|R|} = \frac{|\{(x,y) \in R : f(x,y) = 1\}|}{|R|}$$

(the number of ones in the rectangle divided by its size). For deterministic ϕ, define

$$R_{\phi,f}(\delta) \overset{\text{def}}{=} \left\{(x,y) : (1-\delta)\frac{|\mathbf{1}_f|}{n^2} \leq \rho_f\left(R_\phi(x,y)\right) \leq (1+\delta)\frac{|\mathbf{1}_f|}{n^2}\right\}$$

to be the inputs covered by rectangles $R \in \mathscr{R}_\phi$ with $\rho_f(R)$ deviating from $|\mathbf{1}_f|/n^2$ by a factor of at most 1 ± δ.

We assumed in the beginning of Section IV that $|\mathbf{1}_f| \leq n^2/2$. Therefore, if $R \in \mathscr{R}_\phi$ satisfies $R \subseteq R_{\phi,f}(\delta)$, then

$$\begin{aligned}|R \cap \mathbf{0}_f|/|R| &= \left(|R| - |R \cap \mathbf{1}_f|\right)/|R| \\ &= 1 - \rho_f(R) \\ &\geq 1 - \frac{|\mathbf{1}_f|}{n^2}(1+\delta) \\ &\geq (1-\delta)\frac{|\mathbf{0}_f|}{n^2}. \end{aligned} \tag{3}$$

A similar calculation can show that $|R \cap \mathbf{0}_f|/|R| \leq (1+\delta)|\mathbf{0}_f|/n^2$.

Intuitively, rectangles with proportion of ones similar to that of the original function table cannot reduce the uncertainty about the function's value. The next lemma says that if the protocol errs with low probability, then $R_{\phi,f}(\delta)$ must be small.

Lemma 8: For every deterministic protocol ϕ, and all $\delta > 0$,

$$\bar{E}_\phi(f) \geq \frac{1-\delta}{2n^2}\left|R_{\phi,f}(\delta)\right|.$$

Proof: Let $0_\phi \overset{\text{def}}{=} R_{\phi,f}(\delta) \cap \{(x,y) : V_\phi(x,y) = 0\}$ be the set of inputs in $R_{\phi,f}(\delta)$ that ϕ declares zero, and $1_\phi \overset{\text{def}}{=} R_{\phi,f}(\delta) \cap \{(x,y) : V_\phi(x,y) = 1\}$—the set of inputs in $R_{\phi,f}(\delta)$ that ϕ declares one. Clearly, $0_\phi \cup 1_\phi = R_{\phi,f}(\delta)$, and since ϕ is deterministic, $\bar{E}_\phi^S(f) = |\{(x,y) : V_\phi(x,y) \neq f(x,y)\}|/|S|$. Therefore,

$$\begin{aligned}\bar{E}_\phi^{\mathbf{1}_f}(f) &= \frac{1}{|\mathbf{1}_f|}\sum_{\mathbf{1}_f} \chi_{\{(x,y) : V_\phi(x,y) = 0\}}(x,y) \\ &\geq \frac{1}{|\mathbf{1}_f|}\sum_{\{R \in \mathscr{R}_\phi : R \subseteq 0_\phi\}} \sum_{\mathbf{1}_f \cap R} 1 \\ &\geq \frac{1}{|\mathbf{1}_f|}\sum_{\{R \in \mathscr{R}_\phi : R \subseteq 0_\phi\}} (1-\delta)\left(|\mathbf{1}_f|/n^2\right)|R| \\ &= \frac{1}{n^2}(1-\delta)|0_\phi|. \end{aligned}$$

We showed in (3) that all rectangles $R \in \mathscr{R}_\phi$ that are contained in $R_{\phi,f}(\delta)$, satisfy $|R \cap \mathbf{0}_f| > |R|(1-\delta)(|\mathbf{0}_f|/n^2)$. Thus, we also have

$$\bar{E}_\phi^{\mathbf{0}_f}(f) \geq \frac{1}{n^2}(1-\delta)|1_\phi|.$$

Combining the two,

$$\begin{aligned}\bar{E}_\phi(f) &= \frac{1}{2}\left(\bar{E}_\phi^{\mathbf{0}_f}(f) + \bar{E}_\phi^{\mathbf{1}_f}(f)\right) \\ &\geq \frac{1-\delta}{2n^2}\left(|0_\phi| + |1_\phi|\right) = \frac{1-\delta}{2n^2}\left|R_{\phi,f}(\delta)\right|. \end{aligned}$$

Q.E.D.

Let $r_f(\delta) \overset{\text{def}}{=} \max\{|R| : |\rho_f(R) - (|\mathbf{1}_f|/n^2)| > \delta(|\mathbf{1}_f|/n^2)\}$ be the size of the largest rectangle in $\{0,\cdots,n-1\} \times \{0,\cdots,n-1\}$ with proportion of ones deviating from (s/n^2) by a factor of more than $1 \pm \delta$. ($r_f(\delta)$ is independent of ϕ.) Then, for all deterministic protocols ϕ, if R is a rectangle in $\mathscr{R}_\phi$ and $|R| > r_f(\delta)$ then R must be contained in $R_{\phi,f}(\delta)$. From the last lemma,

$$\sum_{\{R \in \mathscr{R}_\phi : |R| > r_f(\delta)\}} \frac{|R|}{n^2} \leq \frac{|R_{\phi,f}(\delta)|}{n^2} \leq \frac{2\bar{E}_\phi(f)}{1-\delta}. \tag{4}$$

The inequality implies that the total area occupied by rectangles of size $> r_f(\delta)$ is small if $\bar{E}_\phi(f)$ is small. (In Lemma 12, we show that for most functions $r_f(\delta)$ is also small.) The next simple lemma says that when this is the case (the larger probabilities occupy a small part of the probability space), the entropy is high.

Lemma 9: If $\langle r_j\colon\ j=1,\cdots,J\rangle$ is a probability distribution such that $\Sigma_{\{j:\ r_j>r\}} r_j \le A$, then $H\langle r_j\rangle \ge (1-A)\log(1/r)$.

Proof: Compute the average number of bits needed to describe the events $r_j \le r$. Q.E.D.

When Lemma 2 is applied to $S=\{0,\cdots,n-1\}\times\{0,\cdots,n-1\}$, it yields $\overline{L}_\phi \ge H\langle |R|/n^2\colon\ R\in\mathscr{R}_\phi\rangle$. Combining it with (4) and Lemma 9, we obtain, for all ϕ and all $\delta>0$,

$$\overline{L}_\phi \ge H\langle |R|/n^2\colon\ R\in\mathscr{R}_\phi\rangle \ge \left(1-\frac{2\overline{E}_\phi(f)}{1-\delta}\right)\log\frac{n^2}{r_f(\delta)}. \tag{5}$$

Next, we need to relate $r_f(\delta)$ to $|\mathbf{1}_f|$. First, we use a basic result from probability theory to reduce the problem of functions in $\mathscr{F}_s$ that behave hypergeometrically to that of Bernoulli random variables which are easier to analyze.

Lemma 10: Let H be distributed hypergeometrically with parameters N, s, and r. That is,

$$P(H=k)=H_{N,s,r}(k)\stackrel{\text{def}}{=}\frac{\binom{r}{k}\binom{N-r}{s-k}}{\binom{N}{s}}=\frac{\binom{s}{k}\binom{N-s}{r-k}}{\binom{N}{r}}$$

and let B be distributed binomially with probability s/N over r variables. That is,

$$P(B=k)=B_{s/N,r}(k)\stackrel{\text{def}}{=}\binom{r}{k}\left(\frac{s}{N}\right)^k\left(1-\frac{s}{N}\right)^{r-k}.$$

Then, $r(s/N)$ is the expected value of both H and B, and, for $a\le\lfloor r(s/N)\rfloor\le\lceil r(s/N)\rceil\le b$,

$$P(a\le H\le b)\ge P(a\le B\le b).$$

Proof: The lemma follows from a result of Uhlmann [11] showing that for all N, s, and r,

$$P(0\le H\le x)-P(0\le B\le x)$$
$$\begin{cases}\ge 0, & \text{for } x\ge \dfrac{s}{N}(r-1)\dfrac{N+1}{N}\\ \le 0, & \text{for } x\le \dfrac{s}{N}(r-1)\dfrac{N+1}{N}-\dfrac{r-1}{N}\end{cases}$$

(the original result was phrased slightly differently). Note that $(s/N)(r-1)((N+1)/N)<(s/N)r$, hence

$$P(0\le H\le b)\ge P(0\le B\le b).$$

Also, $(s/N)(r-1)((N+1)/N)-(r-1)/N>(s/N)r-1$, hence

$$P(0\le H\le a-1)\le P(0\le B\le a-1)$$

and the lemma follows. Q.E.D.

To bound the size of "nontypical" rectangles, we use Bernstein's lemma.

Lemma 11 [12]: Let $X_1,\cdots,X_r$ be independent, Bernoulli-p, random variables, and let $\delta<1-p$. Then,

$$P\left(\left|\frac{\sum_{i=1}^{r}X_i}{r}-p\right|>\delta p\right)\le e^{-\delta^2 rp/4.5.}$$

Therefore, if f is a random function in $\mathscr{F}_s$, the probability that a given rectangle of size $>r$ will have a proportion of f-ones deviating from $(s/n)^2$ by a factor of more than $1\pm\delta$ is $<e^{-\delta^2 rs/4.5n^2}$. The probability that any rectangle of size $>r$ will have a proportion of f-ones deviating from $(s/n)^2$ by more than δ is $<2^{2n}\cdot e^{-\delta^2 rs/4.5n^2}$. If $r\ge 9n^3/(s\cdot\delta^2)$, this probability tends to zero faster than $(2/e)^n$. Therefore, a fraction of a least $1-(2/e)^n$ of the functions in $\mathscr{F}_s$ do not contain any rectangle R of size $\ge 9n^3/s\cdot\delta^2$ with a proportion of f-ones deviating from $(s/n)^2$ by a factor of more than $1\pm\delta$. We have thus proved the following.

Lemma 12: For all s and $0<\delta<1-(s/n)^2$, a fraction of at least $1-(2/e)^n$ of the functions in $\mathscr{F}_s$ have $r_f(\delta)\le 9n^3/s\cdot\delta^2$.

Substituting in (5) we get that for $0<\delta<1-(s/n)^2$,

$$\overline{L}_\phi\ge\left(1-\frac{2\overline{E}_\phi(f)}{1-\delta}\right)\log\frac{n^2}{r_f(\delta)}\ge\left(1-\frac{2\overline{E}_\phi(f)}{1-\delta}\right)\log\frac{s\delta^2}{9n}.$$

We want results tight up to an additive term, so we can think of δ as being very small (in fact, decreasing to 0). For $\delta<1/2$, $1/(1-\delta)<1+2\delta$, and, for very small δ's, the difference (always less than δ) is negligible. Therefore, we approximate

$$\begin{aligned}\overline{L}_\phi &\ge \left(1-2\overline{E}_\phi(f)\right)\log\frac{s\delta^2}{9n}-4\delta\overline{E}_\phi(f)\log\frac{s\delta^2}{9n}\\ &> \left(1-2\overline{E}_\phi(f)\right)\log\frac{s\delta^2}{9n}-4\delta\log\frac{s\delta^2}{9n}\\ &> \left(1-2\overline{E}_\phi(f)\right)\log\frac{s}{n}+2\left(1-2\overline{E}_\phi(f)\right)\\ &\quad\cdot\log\delta-4\delta\log\frac{s}{n}-4.\end{aligned}$$

Since $s\le n^2/2$, this inequality is valid for all $\delta<1/2$. The (negative) expression $2(1-2\overline{E}_\phi(f))\log\delta-4\delta\log(s/n)$ is maximized for $\delta=(1-2\overline{E}_\phi(f))/2\ln 2\cdot\log(s/n)$. Substituting, we obtain:

$$\overline{L}_\phi>\left(1-2\overline{E}_\phi(f)\right)\left(\log\frac{s}{n}-2\log\log\frac{s}{n}\right)-9.$$

This result suffices to bound $\overline{C}_D(f,\bar{\epsilon})$ for most functions. To bound $\overline{C}_R(f,\bar{\epsilon})$, we return to the notations introduced at the beginning of the section. Note that for every randomized protocol Φ,

$$\bar{E}_{\Phi}^{\mathbf{0}_f}(f) \overset{\mathrm{def}}{=} \frac{1}{|\mathbf{0}_f|} \sum_{\mathbf{0}_f} \bar{E}_{\Phi}(x, y)$$

$$\overset{\mathrm{def}}{=} \frac{1}{|\mathbf{0}_f|} \sum_{\mathbf{0}_f} p(V_{\Phi}(x, y) \neq f(x, y))$$

$$= \frac{1}{|\mathbf{0}_f|} \sum_{\mathbf{0}_f} \sum_{\phi \in \Phi} p(\phi) \chi_{\{(x, y):\, V_{\phi}(x, y) \neq f(x, y)\}}(x, y)$$

$$= \sum_{\phi \in \Phi} p(\phi) \frac{1}{|\mathbf{0}_f|} \sum_{\mathbf{0}_f} \chi_{\{(x, y):\, V_{\phi}(x, y) \neq f(x, y)\}}(x, y)$$

$$\overset{\mathrm{def}}{=} \sum_{\phi \in \Phi} p(\phi) \bar{E}_{\phi}^{\mathbf{0}_f}(f).$$

Similarly,

$$\bar{E}_{\Phi}^{\mathbf{1}_f}(f) = \sum_{\phi \in \Phi} p(\phi) \bar{E}_{\phi}^{\mathbf{1}_f}(f).$$

Therefore,

$$\bar{E}_{\Phi}(f) \overset{\mathrm{def}}{=} \left(\bar{E}_{\Phi}^{\mathbf{0}_f}(f) + \bar{E}_{\Phi}^{\mathbf{1}_f}(f)\right)/2$$

$$= \left(\sum_{\phi \in \Phi} p(\phi) \bar{E}_{\phi}^{\mathbf{0}_f}(f) + \sum_{\phi \in \Phi} p(\phi) \bar{E}_{\phi}^{\mathbf{1}_f}(f)\right) \Big/ 2$$

$$= \sum_{\phi \in \Phi} p(\phi) \left[\left(\bar{E}_{\phi}^{\mathbf{0}_f}(f) + \bar{E}_{\phi}^{\mathbf{1}_f}(f)\right)/2\right]$$

$$= \sum_{\phi \in \Phi} p(\phi) \bar{E}_{\phi}(f)$$

and

$$\bar{L}_{\Phi} \overset{\mathrm{def}}{=} \frac{1}{n^2} \sum_{(x, y)} \bar{L}_{\Phi}(x, y)$$

$$= \sum_{\phi \in \Phi} p(\phi) \frac{1}{n^2} \sum_{(x, y)} \bar{L}_{\phi}(x, y)$$

$$= \sum_{\phi \in \Phi} p(\phi) \bar{L}_{\phi}$$

$$> \sum_{\phi \in \Phi} p(\phi) \left[\left(1 - 2\bar{E}_{\phi}(f)\right) \log \frac{s}{n} - 2 \log\log \frac{s}{n} - 9\right]$$

$$= \left(1 - 2\bar{E}_{\Phi}(f)\right) \log \frac{s}{n} - 2 \log\log \frac{s}{n} - 9.$$

We have proved the following.

Corollary 8: For all $0 < \epsilon \leq 1/2$, a fraction of at least $1 - (2/e)^n$ of the functions in $\mathscr{F}_s$ satisfy

$$\bar{C}_R(f, \bar{\epsilon}) \geq \bar{C}_R(f, \hat{\epsilon})$$
$$> (1 - 2\epsilon)(\log(s/n) - 2 \log\log(s/n)) - 9.$$

This completes the proof of the lower bound. We conclude the paper by applying some of the techniques developed to find a tight lower bound for the equality function.

Example 4: Consider the equality function once more. Example 3 showed that $\bar{C}_D(\mathrm{equ}, 0) \leq 4$. Consequently, we can only expect high worst case lower bounds. In this example, we prove that $\hat{C}_R(\mathrm{equ}, \bar{\epsilon}) \geq [\log(n/(1 + 2\epsilon(n-1)))] \approx \log(1/\epsilon)$. First, we relate $\bar{E}_{\phi}$ to $\bar{L}_{\phi}^{\mathbf{1}_{\mathrm{equ}}}$ in deterministic protocols.

Let ϕ be any deterministic protocol. Then ϕ induces the partition $\mathscr{R}_{\phi}$ of $\{0, \cdots, n-1\} \times \{0, \cdots, n-1\}$ described in Section V-A. Let $R_1, \cdots, R_J$ be the rectangles in $\mathscr{R}_{\phi}$ that have a nonempty intersection with $\mathbf{1}_{\mathrm{equ}} = \{(0,0), \cdots, (n-1, n-1)\}$. To minimize the error, all other rectangles should have a computed value of 0, so we concentrate on $R_1, \cdots, R_J$.

For $j = 1, \cdots, J$ define $S_j \overset{\mathrm{def}}{=} R_j \cap \mathbf{1}_{\mathrm{equ}}$. Then, $\Sigma_{j=1}^{J} |S_j| = n$ and, since each R_j is a rectangle and S is a diagonal, $|S_j|^2 \leq |R_j|$. The average error over $\mathbf{0}_{\mathrm{equ}}$ is

$$\bar{E}_{\phi}^{\mathbf{0}_{\mathrm{equ}}} = \sum_{\{R \in \mathscr{R}_{\phi}:\, V_{\phi}(R) = 1\}} \frac{|R \cap \mathbf{0}_{\mathrm{equ}}|}{|\mathbf{0}_{\mathrm{equ}}|}$$

$$\geq \sum_{\{j:\, V_{\phi}(R_j) = 1\}} \frac{|R_j - S_j|}{|\mathbf{0}_{\mathrm{equ}}|} \geq \sum_{\{j:\, V_{\phi}(R_j) = 1\}} \frac{|S_j|^2 - |S_j|}{n(n-1)}.$$

For the average error over $\mathbf{1}_{\mathrm{equ}}$ we get,

$$\bar{E}_{\phi}^{\mathbf{1}_{\mathrm{equ}}} = \sum_{\{R \in \mathscr{R}_{\phi}:\, V_{\phi}(R) = 0\}} \frac{|R \cap \mathbf{1}_{\mathrm{equ}}|}{|\mathbf{1}_{\mathrm{equ}}|}$$

$$\geq \sum_{\{j:\, V_{\phi}(R_j) = 0\}} \frac{|S_j|}{n} \geq \sum_{\{j:\, V_{\phi}(R_j) = 0\}} \frac{|S_j|^2 - |S_j|}{n(n-1)}.$$

Incorporating these equations into the average error calculation yields

$$\bar{E}_{\phi} = \frac{1}{2}\left(\bar{E}_{\phi}^{\mathbf{0}_{\mathrm{equ}}} + \bar{E}_{\phi}^{\mathbf{1}_{\mathrm{equ}}}\right) \geq \frac{1}{2n(n-1)} \sum_{j=1}^{J} \left(|S_j|^2 - |S_j|\right)$$

$$= \frac{1}{2n(n-1)} \left(\sum_{j=1}^{J} |S_j|^2 - n\right).$$

Hence

$$\sum_{j=1}^{J} \left(|S_j| / |\mathbf{1}_{\mathrm{equ}}|\right)^2 \leq \left(2\bar{E}_{\phi}(n-1) + 1\right)/n.$$

Using the inequality

$$H\langle p_j\colon j = 1, \cdots, J\rangle \geq \log \frac{1}{\sum_{j=1}^{J} p_j^2},$$

we get

$$\bar{L}_{\phi}^{\mathbf{1}_{\mathrm{equ}}} \geq H\langle |S_j| / |\mathbf{1}_{\mathrm{equ}}|\colon j = 1, \cdots, J\rangle$$

$$\geq \log\left(\frac{n}{1 + 2\bar{E}_{\phi}(n-1)}\right).$$

By the convexity of the logarithm,

$$\begin{aligned}\overline{L}_{\Phi}^{1_{\mathrm{equ}}} &\geq \sum_{\phi\in\Phi} p(\phi)\log\left(\frac{n}{1+2\overline{E}_{\phi}(n-1)}\right)\\ &\geq \log\left(\frac{n}{1+2\sum_{\phi\in\Phi} p(\phi)\overline{E}_{\phi}(n-1)}\right)\\ &= \log\left(\frac{n}{1+2\overline{E}_{\Phi}(n-1)}\right).\end{aligned}$$

Therefore, $\hat{C}_R(\mathrm{equ},\bar{\epsilon}) \geq \log(n/(1+2\epsilon(n-1))$. To prove tightness we now describe a protocol whose complexity is two bits more than the lower bound. Note, though, that the complexity of the protocol is computed under the shared random sources model.

Let $\epsilon > 0$. Consider all partitions of $\{0,\cdots,n-1\}$ into $\lceil n/(1+\lfloor 2\epsilon(n-1)\rfloor)\rceil$ sets of size $\leq 1+2\epsilon(n-1)$ each. P_x picks one such partition at random (all partitions being equally likely) and transmits the index of the set that contains x. By the model assumption, he need not transmit which partition was picked. P_y then transmits 1 if y belongs to the same set in the partition and 0 otherwise.

The number of bits transmitted is $1+\lceil\log(n/\lfloor 1+2\epsilon(n-1))\rfloor\rceil$. The error probability is clearly zero if $x = y$ and, by symmetry or a more cumbersome argument, at most 2ϵ for $x \neq y$.

As mentioned, the protocol relies heavily on the assumption that the random experiment is shared by P_x and P_y. There are far more than n partitions of $\{0,\cdots,n-1\}$ into sets of size $1+2\epsilon(n-1)$. Thus, if P_x has to specify which one he uses, more than $\log n + 1$ bits must be transmitted. In fact, it is stated in [2] that, under the separation of random sources assumption, $\hat{C}_R(\mathrm{equ},\hat{\epsilon}) = \Theta(\log\log n)$. We note that, for the equality function, worst case and average errors yield similar complexities.

References

[1] H. Abelson, "Lower bounds on information transfer in distributed computations," in *Proc. 19th Annu. Symp. Foundations of Computer Science*, 1978.

[2] A. C. Yao, "Some complexity questions related to distributive computing," in *Proc. 11th Annu. ACM Symp. Theory of Computing*, 1979.

[3] C. H. Papadimitriou and M. Sipser, "Communication complexity," in *Proc. 14th Annu. ACM Symp. Theory of Computing*, Apr. 1982.

[4] A. El Gamal and A. Orlitsky, "Interactive data compression," in *Proc. 25th Annu. Symp. Foundations of Computer Science*, Oct. 1984.

[5] K. Mehlhorn and E. M. Schmidt, "Las Vegas is better then determinism in VLSI and distributed computing," in *Proc. 14th Annu. ACM Symp. Theory of Computing*, Apr. 1982.

[6] A. Orlitsky and A. El Gamal, "Randomized communication complexity," presented at the IEEE Int. Symp. Information Theory, June 1985.

[7] B. Chor and O. Goldreich, "Unbiased bits from sources of weak randomness and probabilistic communication complexity," in *Proc. 26th Annu. Symp. Foundations of Computer Science*, Oct. 1985.

[8] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, July 1948.

[9] K. F. Pang and A. El Gamal, "Communication complexity of computing the Hamming distance," *SIAM J. Computing*, to appear.

[10] M. O. Rabin and A. Yao, unpublished manuscript (see [2]).

[11] W. Uhlmann, "Vergleich der hypergeometrischen mit der Binomial Verteilung," *Metrika*, vol. 10, 1966.

[12] A. Rényi, *Probability Theory*. Amsterdam, The Netherlands: North Holland, 1970.