Electronic Colloquium on Computational Complexity, Report No. 83 (2010)

Efficient Communication Using Partial Information

Mark Braverman* Anup Rao†

May 12, 2010

Abstract

We show how to efficiently simulate the sending of a message M to a receiver who has partial information about the message, so that the expected number of bits communicated in the simulation is close to the amount of additional information that the message reveals to the receiver.

We use our simulation method to obtain several results in communication complexity.

- We prove a new direct sum theorem for bounded round communication protocols. For every k, if the two party communication complexity of computing a function f is $C > k$, then the complexity of computing n copies of f using a k round protocol is at least $\Omega(n(C - O(k + \sqrt{Ck})))$. This is true in the distributional setting under any distribution on inputs, and also in the setting of worst case randomized computation.
- We prove that the internal information cost (namely the information revealed to the parties) involved in computing any functionality using a two party interactive protocol on any input distribution is *exactly* equal to the amortized communication complexity of computing independent copies of the same functionality. Here by amortized communication complexity we mean the average per copy communication in the best protocol for computing multiple copies of a functionality, with a fixed error in each copy of the functionality.
- Finally, we show that the only way to prove a strong direct sum theorem for communication complexity is by solving a particular variant of the pointer jumping problem that we define. If this problem has a cheap communication protocol, then a strong direct sum theorem must hold. On the other hand, if it does not, then the problem itself gives a counterexample for the direct sum question. In the process we show that a strong direct sum theorem for communication complexity holds if and only if efficient compression of communication protocols is possible.

*Microsoft Research New England, mbraverm@cs.toronto.edu.

†Computer Science and Engineering, University of Washington, anuprao@cs.washington.edu.

ISSN 1433-8092

1 Introduction

Suppose a sender wants to transmit a message M that is correlated with an input X to a receiver that has some information Y about X. What is the best way to carry out the communication in order to minimize the expected number of bits transmitted? A natural lowerbound for this problem is the mutual information between the message and X, given Y: $I(M;X|Y)$, i.e. the amount of new information M reveals to the receiver about X. In this work, we give a protocol that has the same effect as sending M, yet the expected number of bits communicated is asymptotically close to optimal — it is the same as the amount of new information that the receiving party learns from M, up to a sublinear additive term.

Our result is a generalization of classical data compression, where Y is empty (or constant), and M is a deterministic function of X. In this case, the information learnt by the receiver is equal to the entropy $H(M)$, and the compression result above corresponds to classical results on data compression first considered by Shannon [Sha48] — M can be encoded so that the expected number of bits required to transmit M is $H(M)+1$ (see for example the text [CT91]). In more recent decades, several works have considered the version where M is not necessarily determined by X [JRS03, HJMR07], but to our knowledge, there has been no successful study of the case described above, where the receiver has some partial information about the sender's message[1].

This problem has received a significant amount of attention in computer science because of its connection to one of the most basic questions in the field: the *direct sum* question. The direct sum question is: what is the complexity of computing n copies of a function $f(x,y)$, in terms of the complexity of computing one copy? A related variant is the direct product question: what is the success probability of computing n copies of f with few resources, in terms of the success probability of computing one copy? Famous examples of direct product theorems include Yao's XOR Lemma [Yao82] and Raz's Parallel Repetition Theorem [Raz95]. In the context of communication complexity, Shaltiel [Sha03] gave a direct product theorem for the discrepancy of a function, but it remains open to give such a theorem for the success probability of communication tasks. While the direct sum question for general models such as Boolean circuits has a long history (cf [Uhl74, Pau76, GF81]), no general results are known, and indeed they cannot be achieved by the standard reductions used in complexity theory, as a black-box reduction mapping a circuit C performing n tasks into a circuit C' performing a single task will necessarily make C' *larger* than C, rather than making it smaller. Indeed it is known that at least the most straightforward/optimistic formulation of a direct sum theorem for Boolean circuits is *false*.[2]

In communication complexity, a form of this question first appeared in a work by Karchmer, Raz, and Wigderson [KRW91], who conjectured a certain direct sum result for deterministic communication complexity of relations, and showed that it would imply that $\mathbf{P} \not\subseteq \mathbf{NC}^1$. Feder, Kushilevitz, Naor, and Nisan [FKNN91] gave a direct sum theorem for non-deterministic communication complexity, and deduced from it a somewhat weaker result for deterministic communication complexity

[1] We observe that if X, Y, M are arbitrary random variables, and the two parties are tasked with sampling M efficiently (as opposed to one party transmitting and the other receiving), it is impossible to succeed in communication comparable to the information revealed by M. For example, if $M = f(X,Y)$, where f is a boolean function with high communication complexity on average for X, Y, M reveals only one bit of information about the inputs, yet cannot be cheaply sampled.

[2] The example comes from fast matrix multiplication. By a counting argument, there exists an $n \times n$ matrix A over GF(2) such that the map $x \mapsto Ax$ requires a circuit of $\Omega(n^2/\log n)$ size. But the map $(x_1, \ldots, x_n) \mapsto (Ax_1, \ldots, Ax_n)$ is just the product of the matrices A and X (whose columns are $x_1, \ldots, x_n$) and hence can be carried out by a circuit of $O(n^{2.38}) \ll n \cdot (n^2/\log n)$. See Shaltiel's paper [Sha03] for more on this question.

— if a single copy of a function f requires C bits of communications, then n copies require $\Omega(\sqrt{C}n)$ bits. Feder et al also considered the direct sum question for *randomized* communication complexity (see also Open Problem 4.6 in [KN97]) and showed that the dependence of the communication on the error of the protocol for many copies can be better than that obtained by the naïve protocol for many copies.

Direct sum results for randomized communication complexity are closely related to results about compressing randomized communication protocols. Chakrabarti et al [CSWY01] were the first to relate the compression question to the direct sum question in this context. They introduced a notion that they called the *information cost* of a protocol, to measure the amount of information that an observer learns about the inputs of the parties by watching the messages and public randomness of the protocol. In our work we refer to this as the *external information cost*, to contrast it with the *internal information cost* (defined in [BBCR10]), that measures the information learnt by the parties in the protocol.

Formally, given a distribution on inputs X, Y to a communication problem, the external information cost is the mutual information $I(XY; \pi)$ between the inputs (XY), and the messages sent and the public randomness in the protocol π. The internal information cost is the sum $I(X; \pi|Y) + I(Y; \pi|X)$. It is the total new information learnt by each of the parties through participation in the protocol, over the information that they already knew via their inputs. The internal information cost is smaller or equal to the external information cost, which in turn does not exceed the communication complexity of a protocol.

The two measures of information are the same when X is independent of Y, and in this case an optimal direct sum theorem can be proved for the external information cost (and consequently for the internal information cost). Chakrabarti et al showed that from a protocol computing n copies of f on independent inputs with communication C, one can obtain a protocol computing f with external information cost C/n, as long as the inputs X, Y to f are independent of each other. Thus the problem of proving direct sum theorems for independent inputs reduces to the problem of simulating a protocol τ with small external information cost with a protocol ρ that has small communication. That is, the direct sum question reduces to the problem of protocol compression. Chakrabarti et al used this idea to give a direct sum theorem in the case that the communication involves one simultaneous round of messages. This was followed by a few works [JRS03, HJMR07] that obtained stronger direct sum theorems by designing more efficient compression algorithms that could compress each round of the communication in turn. These results applied to protocols where the number of rounds of communication for the protocol computing n copies was restricted to being sufficiently smaller than the communication complexity of computing one copy, and only applied when the inputs were independent of each other, since this is the only scenario where the external information cost could be bounded in the reduction for multiround protocols. Jain et al [JRS05] did manage to get results for arbitrary distributions on inputs as long as the protocols were restricted to being non-interactive (i.e. there is a single simultaneous round of communication). In this case they showed that computing n copies of f must require $\Omega(n)$ times the communication for computing a single copy.

For arbitrary input distributions, if n copies of f can be computed with communication C, then there is a protocol with internal information cost C/n and communication C that computes one copy of f, as shown implicitly in [BYJKS04] and explicitly in our prior work with Barak and Chen [BBCR10]. In that work we showed how to compress the communication globally, rather than round by round. As a result, we obtained direct sum results with no restrictions on the

dependence between inputs or on the number of rounds. Ignoring polylogarithmic factors, we showed that any protocol involving C bits of communication whose internal information cost is I can be compressed to give a protocol with communication complexity $\sqrt{C \cdot I}$. If the *external* information cost is $I_{ext} \geq I$, we gave a different compression scheme that gives a protocol with communication complexity $\tilde{O}(I_{ext})$. A consequence is a direct sum theorem proving that computing n copies of a function requires $\sqrt{n}$ times the communication, with no restrictions on the number of rounds of communication, under any input distribution, and even for worst case randomized computation. In the case that the inputs are independent of each other, our second compression scheme shows that n times the communication is required (again ignoring polylogarithmic factors), thus yielding a near-optimal direct sum theorem for distributional complexity over product distributions.

The main challenge that remains is to find a more efficient way to compress protocols whose internal information cost is small. Indeed, as we discuss below, in this work we show that this is essentially the *only* way to make progress on the direct sum question, in the sense that if there is some protocol that cannot be compressed well, then there is a way to compute n copies of some function surprisingly efficiently.

1.1 Our Results

Our main technical result is a protocol for two parties to efficiently sample from a distribution P that only the sender knows, by taking advantage of a distribution Q known only to the receiver. We obtain a protocol whose communication complexity can be bounded in terms of the informational divergence $\mathbf{D}(P||Q) = \sum_x P(x) \log(P(x)/Q(x))$.

Theorem 1.1. *Suppose that player A is given a distribution P and player B is given a distribution Q over a universe $\mathcal{U}$. There is a public coin protocol that uses an expected*

$$\mathbf{D}(P||Q) + \log(1/\varepsilon) + O\left(\sqrt{\mathbf{D}(P||Q)} + 1\right)$$

bits of communication such that at the end of the protocol:

- *Player A outputs an element a distributed according to P;*
- *Player B outputs b such that for each $x \in \mathcal{U}$, $\mathbf{P}[b = x |\ a = x] > 1 - \varepsilon$.*

As a corollary, we obtain the formulation discussed earlier. For any distribution X, Y and message M that is independent of Y once X is fixed, we can have the sender set P to be the distribution of M conditioned on her input x, and the receiver set Q to be the distribution of M conditioned on her input y. The expected divergence $\mathbf{D}(P||Q)$ turns out to be equal to the mutual information $I(M; X|Y)$. If we apply Theorem 1.1 to each round of communication in a multiround protocol, we can show the following corollary.

Corollary 1.2. *Let X, Y be inputs to an r round communication protocol π whose internal information cost is I. Then for every $\varepsilon > 0$, there exists a protocol τ such that at the end of the protocol, each party outputs a transcript for π. Furthermore, there is an event G with $\mathbf{P}[G] > 1 - r\varepsilon$ such that conditioned on G, the expected communication of τ is $I + O(\sqrt{rI} + 1) + r \log(1/\varepsilon)$, and both parties output the same transcript distributed exactly according to $\pi(X, Y)$.*

The proof appears in Section 5.1. Let f^n denote the function that computes n copies of f on n different inputs. Protocol compression yields the following direct sum theorem:

Corollary 1.3 (Direct Sum for Bounded Rounds)**.** *Let C be the communication complexity of the best protocol for computing f with error ρ on inputs drawn from μ. Then any r round protocol computing f^n on the distribution μ^n with error $\rho - \varepsilon$ must involve at least $\Omega(n(C - r\log(1/\varepsilon) - O(\sqrt{C \cdot r})))$ communication.*

Although the information cost of solving a problem may be much smaller than its communication complexity, we show that the internal information cost of computing a function f according to a fixed distribution is *exactly* equal to the amortized communication complexity of computing many copies of f. Specifically, let $\mathsf{IC}^{\mathsf{i}}{}_{\mu}(f, \rho)$ denote the smallest possible internal information cost of any protocol computing f with probability of failure at most ρ when the inputs are drawn from the distribution μ. Denote by $D_{\rho}^{\mu,n}(f)$ the communication complexity of the best protocol for computing f on n independent inputs drawn from μ so that the error with respect to each coordinate is at most ρ. We obtain the following:

Theorem 1.4. *For any f, μ, and ρ,*

$$\mathsf{IC}^{\mathsf{i}}{}_{\mu}(f, \rho) = \lim_{n \to \infty} \frac{D_{\rho}^{\mu,n}(f)}{n}.$$

Finally, we define a communication problem we call Correlated Pointer Jumping – $\mathsf{CPJ}(C, I)$ – that is parametrized by two parameters C and I such that $C \gg I$. $\mathsf{CPJ}(C, I)$ is designed in a way that the randomized communication complexity cost $I < \mathsf{CC}(\mathsf{CPJ}(C, I)) < C$. By combining prior work with new results, we show that determining the worst case randomized communication complexity $\mathsf{CC}(\mathsf{CPJ}(C, I))$ for $I = C/n$ is equivalent (up to poly-logarithmic factors) to determining the best parameter $k(n)$ for which a direct sum theorem $\mathsf{CC}(f^n) = \Omega(k(n) \cdot \mathsf{CC}(f))$ holds. For simplicity, we formulate only part of the result here, see Section 5.3 for more details.

Theorem 1.5. *If $\mathsf{CC}(\mathsf{CPJ}(C, C/n)) = \tilde{O}(C/n)$ for all C, then a near optimal direct sum theorem holds: $\mathsf{CC}(f^n) = \tilde{\Omega}(n \cdot \mathsf{CC}(f))$ for* all *f.*

On the other hand, if $\mathsf{CC}(\mathsf{CPJ}(C, C/n)) = \Omega((C \log^a C)/n)$ for all $a > 0$, then direct sum is violated by $\mathsf{CPJ}(C, C/n)$:

$$\mathsf{CC}(\mathsf{CPJ}(C, C/n)^n) = O(C \log C) = o(n \cdot \mathsf{CC}(\mathsf{CPJ}(C, C/n))/\log^a C),$$

for all a.

1.2 Techniques

The key technical contribution of our work is a sampling protocol that proves Theorem 1.1. Recall that the informational divergence $\mathbf{D}(P||Q)$ is equal to $\sum_x P(x) \log \frac{P(x)}{Q(x)}$. Intuitively, if this quantity is small, then typically the ratio $\frac{P(x)}{Q(x)}$ is close to 1. To illustrate our technique, let us focus on an easy special case: suppose Q is the uniform distribution on some subset S_Q of the universe $\mathcal{U}$, and P is the uniform distribution on some subset $S_P \subset S_Q$. Then the informational divergence $\mathbf{D}(P||Q)$ is exactly $\log(|S_Q|/|S_P|)$.

In this case, the players use an infinite public random tape that samples an infinite sequence of elements $a_1, a_2, \ldots$ uniformly at random from the universe $\mathcal{U}$. Player A then picks the first element that lies in S_P to be his sample. If this element is a_i, player A sends $k = \lceil i/|\mathcal{U}| \rceil$ to player B. In expectation k is only a constant, so the expected number of bits for this step is only a constant.

Next the players use the public randomness to sample a sequence of uniformly random boolean functions on the universe. A then sends the value of approximately $\log(1/\varepsilon)$ of these functions evaluated on his sample. B looks at her window of $|\mathcal{U}|$ elements and checks to see whether any of them agree with the evaluations sent by A and are in her set S_Q. If more than one agrees with A she asks A to send more evaluations of random functions. They continue this process until there is a unique element in the k'th interval that agrees with the evaluations and is in the set S_Q. For the analysis, note that the fraction of elements in the window that are in S_Q but not in S_P can be bounded in terms of the divergence between P and Q.

Of course in general P and Q are not distributions that take uniformly random points in sets. Still, our protocol for doing the sampling is simple, and is based on intuitions similar to those used in the example above.

2 Preliminaries

Notation. We reserve capital letters for random variables and distributions, calligraphic letters for sets, and small letters for elements of sets. Throughout this paper, we often use the notation $|b$ to denote conditioning on the event $B = b$. Thus $A|b$ is shorthand for $A|B = b$.

We use the standard notion of *statistical/total variation* distance between two distributions.

Definition 2.1. Let D and F be two random variables taking values in a set $\mathcal{S}$. Their *statistical distance* is

$$|D - F| \stackrel{def}{=} \max_{\mathcal{T} \subseteq \mathcal{S}} (|\Pr[D \in \mathcal{T}] - \Pr[F \in \mathcal{T}]|) = \frac{1}{2} \sum_{s \in \mathcal{S}} |\Pr[D = s] - \Pr[F = s]|$$

If $|D - F| \leq \varepsilon$ we shall say that D is ε-*close* to F. We shall also use the notation $D \stackrel{\varepsilon}{\approx} F$ to mean D is ε-close to F.

2.1 Information Theory

Definition 2.2 (Entropy)**.** The *entropy* of a random variable X is $H(X) \stackrel{def}{=} \sum_x \Pr[X = x] \log(1/\Pr[X = x])$. The *conditional entropy* $H(X|Y)$ is defined to be $\mathbf{E}_{y \in_{\mathrm{R}} Y} [H(X|Y = y)]$.

Fact 2.3. $H(AB) = H(A) + H(B|A)$.

Definition 2.4 (Mutual Information)**.** The *mutual information* between two random variables A, B, denoted $I(A; B)$ is defined to be the quantity $H(A) - H(A|B) = H(B) - H(B|A)$. The *conditional mutual information* $I(A; B|C)$ is $H(A|C) - H(A|BC)$.

In analogy with the fact that $H(AB) = H(A) + H(B|A)$,

Proposition 2.5. *Let C_1, C_2, D, B be random variables. Then*

$$I(C_1C_2; B|D) = I(C_1; B|D) + I(C_2; B|C_1D).$$

The previous proposition immediately implies the following:

Proposition 2.6 (Super-Additivity of Mutual Information)**.** *Let C_1, C_2, D, B be random variables such that for every fixing of D, C_1 and C_2 are independent. Then*

$$I(C_1; B|D) + I(C_2; B|D) \quad \leq \quad I(C_1C_2; B|D).$$

We also use the notion of *divergence*, which is a different way to measure the distance between two distributions:

Definition 2.7 (Divergence)**.** The informational divergence between two distributions is $\mathbf{D}(A||B) \stackrel{def}{=} \sum_x A(x) \log(A(x)/B(x))$.

For example, if B is the uniform distribution on $\{0,1\}^n$ then $\mathbf{D}(A||B) = n - H(A)$.

Proposition 2.8. *Let A, B, C be random variables in the same probability space. For every a in the support of A and c in the support of C, let B_a denote $B|A = a$ and B_{ac} denote $B|A = a, C = c$. Then $I(A; B|C) = \mathbf{E}_{a,c \in_R A,C} [\mathbf{D}(B_{ac}||B_c)]$*

Lemma 2.9.

$$\mathbf{D}(P_1 \times P_2||Q_1 \times Q_2) = \mathbf{D}(P_1||Q_1) + \mathbf{D}(P_2||Q_2).$$

2.2 Communication Complexity

Let $\mathcal{X}, \mathcal{Y}$ denote the set of possible inputs to the two players, who we name P_x, P_y. In this paper[3], we view a *private coins protocol* for computing a function $f : \mathcal{X} \times \mathcal{Y} \to \mathbf{Z}_K$ as a rooted tree with the following structure:

- Each non-leaf node is *owned* by P_x or by P_y.
- Each non-leaf node owned by a particular player has a set of children that are owned by the other player. Each of these children is labeled by a binary string, in such a way that this coding is prefix free: no child has a label that is a prefix of another child.
- Every node is associated with a function mapping $\mathcal{X}$ to distributions on children of the node and a function mapping $\mathcal{Y}$ to distributions on children of the node.
- The leaves of the protocol are labeled by output values.

On input x, y, the protocol π is executed as in Figure 1.

A public coin protocol is a distribution on private coins protocols, run by first using shared randomness to sample an index r and then running the corresponding private coin protocol π_r. Every private coin protocol is thus a public coin protocol. The protocol is called deterministic if all distributions labeling the nodes have support size 1.

Definition 2.10. The *communication complexity* of a public coin protocol π, denoted $\mathsf{CC}(\pi)$, is the maximum number of bits that can be transmitted in any run of the protocol.

Definition 2.11. The *number of rounds* of a public coin protocol is the maximum depth of the protocol tree π_r over all choices of the public randomness.

[3]The definitions we present here are equivalent to the classical definitions and are more convenient for our proofs.

Generic Communication Protocol

1. Set v to be the root of the protocol tree.
2. If v is a leaf, the protocol ends and outputs the value in the label of v. Otherwise, the player owning v samples a child of v according to the distribution associated with her input for v and sends the label to indicate which child was sampled.
3. Set v to be the newly sampled node and return to the previous step.

Figure 1: A communication protocol.

Given a protocol π, $\pi(x, y)$ denotes the concatenation of the public randomness with all the messages that are sent during the execution of π. We call this the *transcript* of the protocol. We shall use the notation $\pi(x, y)_j$ to refer to the j'th transmitted message in the protocol. We write $\pi(x, y)_{\leq j}$ to denote the concatenation of the public randomness in the protocol with the first j message bits that were transmitted in the protocol. Given a transcript, or a prefix of the transcript, v, we write $\mathsf{CC}(v)$ to denote the number of message bits in v (i.e. the length of the communication).

Definition 2.12 (Communication Complexity notation)**.** For a function $f : \mathcal{X} \times \mathcal{Y} \to \mathbf{Z}_K$, a distribution μ supported on $\mathcal{X} \times \mathcal{Y}$, and a parameter $\rho > 0$, $D^{\mu}_{\rho}(f)$ denotes the communication complexity of the cheapest deterministic protocol for computing f on inputs sampled according to μ with error ρ. $R_{\rho}(f)$ denotes the cost of the best randomized public coin protocol for computing f with error at most ρ on *every* input.

We shall use the following theorem due to Yao:

Theorem 2.13 (Yao's Min-Max)**.** $R_{\rho}(f) = \max_{\mu} D^{\mu}_{\rho}(f)$.

Recall that the internal information cost $\mathsf{IC}^{\mathsf{i}}{}_{\mu}(\pi)$ of a protocol π is defined to be $I(\pi(X, Y); X|Y) + I(\pi(X, Y); Y|X)$.

Lemma 2.14. *Let R be the public randomness used in the protocol π. Then $\mathsf{IC}^{\mathsf{i}}{}_{\mu}(\pi) = \mathbf{E}_R \left[\mathsf{IC}^{\mathsf{i}}{}_{\mu}(\pi_R)\right]$*

Proof.

$$\begin{aligned}
\mathsf{IC}^{\mathsf{i}}{}_{\mu}(\pi) &= I(\pi(X, Y); X|Y) + I(\pi(X, Y); Y|X) \\
&= I(R; X|Y) + I(R; Y|X) + I(\pi(X, Y); X|YR) + I(\pi(X, Y); Y|XR) \\
&= I(\pi(X, Y); X|YR) + I(\pi(X, Y); Y|XR) \\
&= \mathop{\mathbf{E}}_{R} \left[\mathsf{IC}^{\mathsf{i}}{}_{\mu}(\pi_R)\right]
\end{aligned}$$

□

The following theorem was proved in [BYJKS04]. Here we cite a version appearing in [BBCR10]:

Theorem 2.15. *For every μ, f, ρ there exists a protocol τ computing f on inputs drawn from μ with probability of error at most ρ and communication at most $\mathsf{D}^{\mu^n}_{\rho}(f^n)$ such that $\mathsf{IC}^{\mathsf{i}}{}_{\mu}(\tau) \leq \frac{D^{\mu^n}_{\rho}(f^n)}{n}$.*

For our results on amortized communication complexity, we need the following definition: we shall consider the problem of computing n copies of f, with error ρ in each coordinate of the computation, i.e. the computation must produce the correct result in any single coordinate with probability at least $1-\rho$. We denote the communication complexity of this problem by $D_\rho^{\mu,n}(f) \leq \mathsf{D}_\rho^{\mu^n}(f^n)$. Formally,

Definition 2.16. Let μ be a distribution on $X \times Y$ and let $0 < \rho < 1$. We denote by $D_\rho^{\mu,n}(f)$ the distributional complexity of computing f on each of n independent pairs of inputs drawn from μ, with probability of failure at most μ on each of the inputs.

The result above is actually much stronger, the same proof that appears in [BBCR10] shows the following theorem:

Theorem 2.17. *For every μ, f, ρ there exists a protocol τ computing f on inputs drawn from μ with probability of error at most ρ and communication at most $\mathsf{D}_\rho^{\mu,n}(f^n)$ such that $\mathsf{IC}^{\mathsf{i}}{}_\mu(\tau) \leq \frac{D_\rho^{\mu,n}(f)}{n}$.*

3 Sampling From Correlated Distributions

Here we prove the following theorem

Theorem 3.1. *Suppose that player A is given a distribution P and player B is given a distribution Q over a universe $\mathcal{U}$. There is a protocol that uses an expected*

$$\mathbf{D}\left(P||Q\right) + \log 1/\varepsilon + O(\mathbf{D}\left(P||Q\right)^{1/2} + 1)$$

bits of communication such that at the end of the protocol:

- *player A outputs an element a distributed according to P;*
- *for each x, $\mathbf{P}[b = a|\ a = x] > 1 - \varepsilon$.*

Note that the second condition implies in particular that player B outputs an element b such that $b = a$ with probability $> 1 - \varepsilon$. The protocol requires no prior knowledge or assumptions on $\mathbf{D}\left(P||Q\right)$.

Proof. We prove the theorem by exhibiting such a protocol. The protocol runs as follows. Both parties interpret the shared random tape as a sequence of uniformly selected elements $\{a_i\}_{i=1}^\infty = \{(x_i, p_i)\}_{i=1}^\infty$ from the set $\mathcal{A} := \mathcal{U} \times [0,1]$. Denote the subset

$$\mathcal{P} := \{(x,p)\ :\ P(x) < p\}$$

of $\mathcal{A}$ as the set of points under the histogram of the distribution P. Similarly, define

$$\mathcal{Q} := \{(x,p)\ :\ Q(x) < p\}.$$

For a constant $C \geq 1$ we will define the C-multiple of $\mathcal{Q}$ as

$$C \cdot \mathcal{Q} := \{(x,p) \in \mathcal{A}\ :\ (x, p/C) \in \mathcal{Q}\}.$$

We will also use a different part of the shared random tape to obtain a sequence of random hash functions $h_i : \mathcal{U} \to \{0,1\}$ so that for any $x \neq y \in \mathcal{U}$, $\mathbf{P}[h_i(x) = h_i(y)] = 1/2$.

We are now ready to present the protocol:

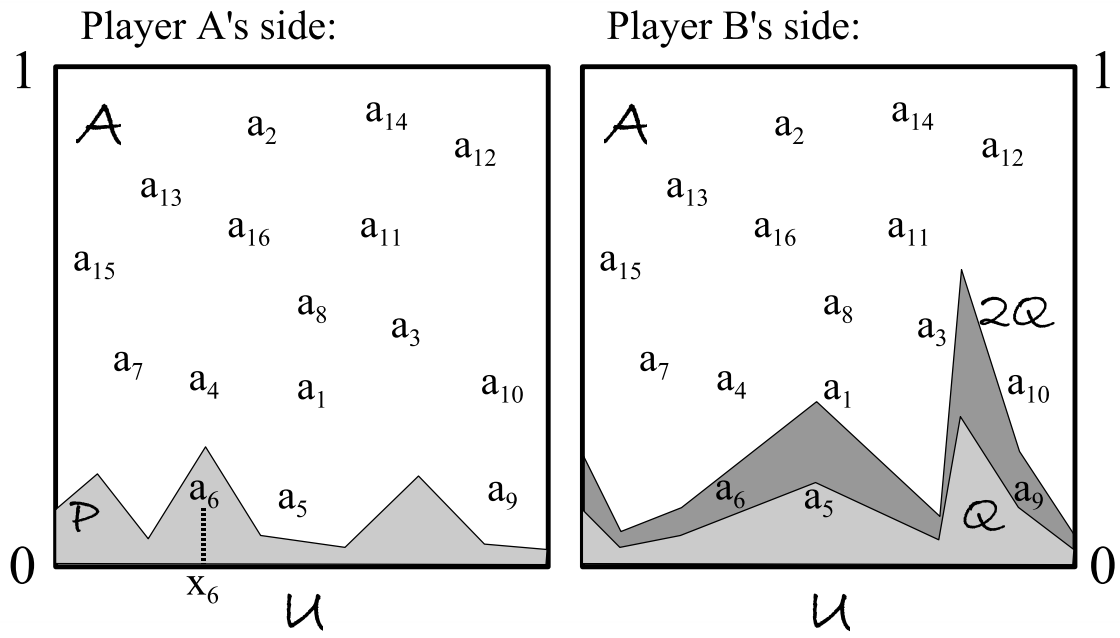

Figure 2: An illustration on the execution of the protocol. The elements a_i are selected uniformly from $\mathcal{A} = \mathcal{U} \times [0,1]$. The first a_i to fall in $\mathcal{P}$ is a_6, and thus player A outputs x_6. Player A sends hashes of a_6, which do not match the hashes of a_5, the only a_i in $\mathcal{Q}$. Player B responds 'failure', and considers surviving elements in $2\mathcal{Q}$, which are a_6 and a_9. After a few more hashes from A, a_6 is selected by B with high probability.

1. Player A selects the first index i such that $a_i \in \mathcal{P}$, and outputs x_i;
2. Player A sends Player B the binary encoding of $k := \lceil i/|\mathcal{U}| \rceil$;
3. Player A sends the values of $h_j(x)$ for $j = 1, .., s_0$, where $s_0 := 2 + \lceil \log 1/\varepsilon \rceil$;
4. Repeat, until Player B produces an output, beginning with iteration $t = 0$:
   (a) set $C := 2^{t^2}$;
   (b) if there is an $a_r = (y, q)$ with $r \in \{(k-1) \cdot |\mathcal{U}| + 1, \ldots, k \cdot |\mathcal{U}|\}$ in $C \cdot \mathcal{Q}$ such that $h_j(y) = h_j(x)$ for $j = 1, .., s_t$, Player B responds 'success' and outputs y; if there is more than one such a, player B selects the first one;
   (c) otherwise, Player B responds 'failure' and Player A sends $2t + 3$ more hash values $h_{s_t+1}(x), \ldots, h_{s_t+t+1}(x)$ and sets $s_{t+1} := s_t + 2t + 3 = 1 + \lceil \log 1/\varepsilon \rceil + (t+2)^2$, $t := t + 1$.

It is easy to see that the output of Player A is distributed according to the distribution P. We will show that for any choice of i and the pair (x_i, p_i) by A, Player B outputs the same x_i with probability $> 1 - \varepsilon$. In addition, we will show that the expected amount of communication is $\mathbf{D}(P||Q) + \log 1/\varepsilon + O(\mathbf{D}(P||Q)^{1/2} + 1)$. Hence, in particular, if $\mathbf{D}(P||Q)$ is finite, the protocol terminates with probability 1. We start with the following claim.

Claim 3.2. *For each n, $\mathbf{P}[k > n] < e^{-n}$.*

Proof. For each n, we have

$$\mathbf{P}[k > n] = \mathbf{P}[a_i \notin \mathcal{P} \text{ for } i = 1, \ldots, n \cdot |\mathcal{U}|] = (1 - 1/|\mathcal{U}|)^{|\mathcal{U}| \cdot n} < e^{-n}.$$

□

Thus the expected length of the first two messages from Player A is $\log 1/\varepsilon + O(1)$ bits. It remains to analyze Step 4 of the protocol. We say that an element $a = (x, p)$ *survives* iteration t if $a \in 2^{t^2} \cdot \mathcal{Q}$ and it satisfies $h_j(x) = h_j(x_i)$ for all $j = 1, \ldots, s_t$ for this t.

Note that the "correct" element a_i survives iteration t if and only if $2^{t^2} \geq P(x_i)/Q(x_i)$.

Claim 3.3. *Let E_{a_i} be the event that the element selected by player A is a_i, which is the i-th element on the tape. Denote $k := \lceil i/|\mathcal{U}| \rceil$. Conditioned on E_{a_i}, the probability that a different element a_j with $j \in \{(k-1) \cdot |\mathcal{U}| + 1, \ldots, k \cdot |\mathcal{U}|\}$ survives iteration t is bounded by $\varepsilon/2^{t+1}$.*

Proof. Without loss of generality we can assume that $|\mathcal{U}| \geq 2$, since for a singleton universe our sampling protocol will succeed trivially. This implies that for any $C > 0$ and for a uniformly selected $a \in \mathcal{A}$,

$$\mathbf{P}[a \in C \cdot \mathcal{Q} |\ a \notin \mathcal{P}] \leq \mathbf{P}[a \in C \cdot \mathcal{Q}]/\mathbf{P}[a \notin \mathcal{P}] \leq 2 \cdot \mathbf{P}[a \in C \cdot \mathcal{Q}] \leq 2C/|\mathcal{U}|.$$

Denote $K := k \cdot |\mathcal{U}|$. Conditioning on E_{a_i}, the elements $a_{K-|\mathcal{U}|+1}, \ldots, a_{i-1}$ are distributed uniformly on $\mathcal{A} \setminus \mathcal{P}$, and $a_{i+1}, \ldots, a_K$ are distributed uniformly on $\mathcal{A}$. For any such $j = K - |\mathcal{U}| + 1, \ldots, i-1$, and for any $C > 0$,

$$\mathbf{P}[a_j \in C \cdot \mathcal{Q}] \leq 2C/|\mathcal{U}|.$$

For such a j, surviving round t means a_j belonging to $2^{t^2} \cdot \mathcal{Q}$ and agreeing with a_i on $s_t = 1 + \lceil \log 1/\varepsilon \rceil + (t+1)^2$ random hashes $h_1, \ldots, h_{s_t}$. The probability of this event is thus bounded by

$$\mathbf{P}[a_j \text{ survives round } t] \leq \mathbf{P}[a_j \in 2^{t^2} \cdot \mathcal{Q}] \cdot 2^{-s_t} \leq \frac{2 \cdot 2^{t^2}}{|\mathcal{U}|} \cdot 2^{-s_t} \leq$$
$$2^{t^2 - s_t - 1}/|\mathcal{U}| \leq 2^{-2t-1}\varepsilon/|\mathcal{U}| \leq \varepsilon/(|\mathcal{U}| \cdot 2^{t+1}).$$

By taking a union bound over all $j = K - |\mathcal{U}| + 1, \ldots, K$, $j \neq i$, we obtain the $\varepsilon/2^{t+1}$ bound. □

Thus for any E_{a_i}, the probability of Player B to output anything other than x_i conditioned on E_{a_i} is $< \sum_{t=0}^{\infty} \varepsilon/2^{t+1} = \varepsilon$.

It remains to observe that Step 4 of the protocol is guaranteed to terminate when $t^2 \geq \log P(x_i)/Q(x_i)$ since a_i belongs to $\frac{P(x_i)}{Q(x_i)} \cdot \mathcal{Q}$. Denote $T := \left\lceil \sqrt{\log P(x_i)/Q(x_i)} \right\rceil$. Thus the amount of communication in Step 4 is bounded by

$$S_T - S_0 + T = (T+1)^2 - 1 + T = T^2 + 3T < \log P(x_i)/Q(x_i) + 2 + 5\sqrt{\log P(x_i)/Q(x_i)},$$

and the expected amount of communication is bounded by

$$\mathbf{E}_{x_i \sim P}\left[\log P(x_i)/Q(x_i) + 2 + 5\sqrt{\log P(x_i)/Q(x_i)}\right] =$$
$$\mathbf{D}\left(P||Q\right) + 2 + 5 \cdot \mathbf{E}_{x_i \sim P}\sqrt{\log P(x_i)/Q(x_i)} \leq$$
$$\mathbf{D}\left(P||Q\right) + 2 + 5 \cdot \sqrt{\mathbf{E}_{x_i \sim P} \log P(x_i)/Q(x_i)} = \mathbf{D}\left(P||Q\right) + O(\mathbf{D}\left(P||Q\right)^{1/2} + 1),$$

where the inequality is by the concavity of $\sqrt{\ }$. This completes the proof. □

Remark 3.4. The sampling in the proof of Theorem 3.1 may take significantly more than one round. In fact, the expected number of rounds is $\Theta(\sqrt{\mathbf{D}\left(P||Q\right)})$. One should not hope to get rid of the dependence of the number of rounds in the simulation on the divergence since $\mathbf{D}\left(P||Q\right)$ is not known to the players ahead of time, and the only way to "discover" it (and thus to estimate the amount of communication necessary to perform the sampling task) is through interactive communication. By increasing the expected communication by a constant multiplicative factor, it is possible to decrease the expected number of rounds to $O(\log \mathbf{D}\left(P||Q\right))$.

For technical reasons we will need the following easy extension of Theorem 3.1:

Lemma 3.5. *In the setup of Theorem 3.1 there is an event E such that $\mathbf{P}[E] > 1 - \varepsilon$, and conditioned on E:*

- *both parties output the same value distributed exactly according to P;*
- *the expected communication is still bounded by $\mathbf{D}\left(P||Q\right) + \log 1/\varepsilon + O(\mathbf{D}\left(P||Q\right)^{1/2} + 1)$.*

Proof. Let E' be the event when both parties output the same value (i.e. when the protocol succeeds). Since the probability of success is $> 1 - \varepsilon$ conditioned on the value being output by Player A, there is an event $E \subset E'$ such that $\mathbf{P}[\text{Player } A \text{ outputs } x | E] = P(x)$ for all x and $\mathbf{P}[E] > 1 - \varepsilon$.

It remains to see that the communication guarantee holds. This is trivially true since assuming the protocol succeeds the estimate on the communication amount depends exclusively on the element x sampled, and the analysis in the proof of Theorem 3.1 carries conditioned on E. □

4 Correlated Pointer Jumping

Here we define the correlated pointer jumping problem, that is at the heart of several of our results. The input in this problem is a rooted tree such that

- Each non-leaf node is *owned* by P_x or by P_y.
- Each non-leaf node owned by a particular player has a set of children that are owned by the other player. Each of these children is labeled by a binary string, in such a way that this coding is prefix free: no child has a label that is a prefix of another child.
- Each node v is associated with two distributions on its children: $\mathsf{child}(v)_x$ known to P_x and $\mathsf{child}(v)_y$ known to P_y.
- The leaves of the tree are labeled by output values.

The number of rounds in the instance is the depth of the tree.

The goal of the problem is for the players to sample the leaf according to the distribution that is obtained by sampling each child according to the distribution specified by the owner of the parent. We give a way to measure the correlation between the knowledge of the two parties in the problem. Given an instance F of the correlated pointer jumping problem and a vertex from the tree, we write F_v to denote the correlated pointer jumping problem associated with the tree rooted at v.

Definition 4.1 (Divergence Cost)**.** The divergence cost of a correlated pointer jumping instance whose root is v, denoted $\mathbf{D}(F)$, is recursively defined as follows:

$$\mathbf{D}(F) = \begin{cases} 0 & \text{if the tree has depth 0} \\ \mathbf{D}(\mathsf{child}(v)_x || \mathsf{child}(v)_y) + \mathbf{E}_{w \in_{\mathrm{R}} \mathsf{child}(v)_x}[\mathbf{D}(F_w)] & \text{if v is owned by } P_x \\ \mathbf{D}(\mathsf{child}(v)_y || \mathsf{child}(v)_x) + \mathbf{E}_{w \in_{\mathrm{R}} \mathsf{child}(v)_y}[\mathbf{D}(F_w)] & \text{if v is owned by } P_y \end{cases}$$

We can use our sampling lemma to solve the correlated pointer jumping problem.

Theorem 4.2. *Let F be an r-round correlated pointer jumping instance. Then there is a protocol to sample a leaf such that there is an event E, with $\mathbf{P}[E] > 1 - r\varepsilon$, and conditioned on E, the sampled leaf has the correct distribution and conditioned on E, the expected communication of the protocol is $\mathbf{D}(F) + r\log(1/\varepsilon) + O(\sqrt{r\mathbf{D}(F)} + r)$.*

Proof. We prove the theorem by induction on the depth r. For $r = 0$ the statement is true, since the sampling is trivial. Suppose the statement is true for depth $(r-1)$ instances. Suppose, without loss of generality, that the root v of F is owned by P_x. Then for each $w \in \mathsf{child}(v)$ there is a protocol to sample a leaf from F_w and an event E_w with $\mathbf{P}[E_w] > 1-(r-1)\varepsilon$ such that conditioned on E_w, the expected communication of the protocol is $\mathbf{D}(F_w)+(r-1)\log(1/\varepsilon)+C\cdot(\sqrt{(r-1)(\mathbf{D}(F_w)}+(r-1))$ for a constant C. Denote $D_{rt} := \mathbf{D}(\mathsf{child}(v)_x || \mathsf{child}(v)_y)$ and $D_{ch} := \mathbf{E}_{w \in_{\mathrm{R}} \mathsf{child}(v)_x}[\mathbf{D}(F_w)]$. Then by definition, $\mathbf{D}(F) = D_{rt} + D_{ch}$.

By the sampling theorem, and specifically by Lemma 3.5, there is a protocol for sending the first message and an event E' such that the expected communication conditioned on E' is

$$D_{rt} + \log(1/\varepsilon) + C \cdot (\sqrt{D_{rt}} + 1),$$

and $\mathbf{P}[E'] > 1 - \varepsilon$. Let E be the event that E' holds, child w is sampled and E_w holds. Then clearly $\mathbf{P}[E] > 1 - r\varepsilon$, and conditioned of E holding the communication of the protocol is

$$\begin{aligned} &D_{rt} + \log(1/\varepsilon) + C \cdot (\sqrt{D_{rt}} + 1) + \\ &\qquad \mathop{\mathbf{E}}_{w \in_{\mathrm{R}} \mathsf{child}(v)_x} \left[\mathbf{D}(F_w) + (r-1)\log(1/\varepsilon) + C \cdot (\sqrt{(r-1)\mathbf{D}(F_w)} + (r-1)) \right] \leq \\ &\qquad\qquad \mathbf{D}(F) + r\log(1/\varepsilon) + C \cdot r + C \cdot (\sqrt{D_{rt}} + \sqrt{(r-1)D_{ch}}) \leq \\ &\qquad\qquad\qquad \mathbf{D}(F) + r\log(1/\varepsilon) + C \cdot r + C \cdot (\sqrt{r \cdot \mathbf{D}(F)}) \end{aligned}$$

The first inequality is by the concavity of $\sqrt{\ }$, and the second one holds by the Cauchy Schwartz inequality since $\mathbf{D}(F) = D_{rt} + D_{ch}$. This completes the proof of the inductive step. □

A key fact is that both the internal and external information cost of a protocol can be used to bound the expected divergence cost of an associated distribution on correlated pointer jumping instances. Since, in this work, we only require the connection to internal information cost, we shall restrict our attention to it.

Given a public coin protocol with inputs X, Y and public randomness R, for every fixing of x, y, r, we obtain an instance of correlated pointer jumping. The tree is the same as the protocol tree with public randomness r. If a node v at depth d is owned by P_x, let M be the random variable denoting the child of v that is picked. Then define $\mathsf{child}(v)_x$ so that it has the same distribution as

$M|\, X = x, \pi(X,Y)_{\leq d} = rv$, and $\mathsf{child}(v)_y$ so it has the same distribution as $M|\, Y = y, \pi(X,Y)_{\leq d} = rv$. We denote this instance of correlated sampling by $F_\pi(x,y,r)$. Let μ denote the distribution on X, Y. Next we relate the average divergence cost of this instance to the internal information cost of π:

Lemma 4.3. $\mathbf{E}_{X,Y,R}\left[\mathbf{D}\left(F_\pi(x,y,r)\right)\right] = \mathsf{IC}^{\mathsf{i}}{}_\mu(\pi)$

Proof. We shall prove that for every r, $\mathbf{E}_{X,Y,R}\left[\mathbf{D}\left(F_\pi(x,y,r)\right)\right] = \mathsf{IC}^{\mathsf{i}}{}_\mu(\pi_r)$. The proof can then be completed by Lemma 2.14.

So without loss of generality, assume that π is a private coin protocol, and write $F(x,y)$ to denote the corresponding divergence cost. We shall prove this by induction on the depth of the protocol tree of π. If the depth is 0, then both quantities are 0. For the inductive step, without loss of generality, assume that P_x owns the root node of the protocol. Let M denote the child of the root that is sampled during the protocol, and let $F(x,y)_m$ denote the divergence cost of the subtrees rooted at m. Then

$$\mathop{\mathbf{E}}_{X,Y}\left[\mathbf{D}\left(F(x,y)\right)\right] = \mathop{\mathbf{E}}_{X,Y}\left[\mathbf{D}\left(\mathsf{child}(v)_X || \mathsf{child}(v)_Y\right)\right] + \mathop{\mathbf{E}}_{X,Y}\left[\mathop{\mathbf{E}}_{M\in_{\mathrm{R}}\mathsf{child}(v)_X}\left[\mathbf{D}\left(F(X,Y)_M\right)\right]\right] \tag{1}$$

Since for every x, y, $M|xy$ has the same distribution as $M|x$, Proposition 2.8 gives that the first term in Equation 1 is exactly equal to $I(X;M|Y) = I(X;M|Y) + I(Y;M|X)$. We can rewrite the second term $\mathbf{E}_M\left[\mathbf{E}_{X,Y}\left[\mathbf{D}\left(F(X,Y)_M\right)\right]\right]$. For each fixing of $M = m$, we can use the inductive hypothesis to show that the inner expectation is equal to $I(X;\pi(X,Y)|Ym) + I(Y;\pi(X,Y)|Xm)$. Together, these two bounds imply that

$$\begin{aligned}
&\mathop{\mathbf{E}}_{X,Y}\left[\mathbf{D}\left(F(x,y)\right)\right] \\
&= I(X;M|Y) + I(Y;M|X) + I(X;\pi(X,Y)|YM) + I(Y;\pi(X,Y)|XM) \\
&= \mathsf{IC}^{\mathsf{i}}{}_\mu(\pi)
\end{aligned}$$

□

5 Applications

In this section, we use Theorem 4.2 to prove a few results about compression and direct sums.

5.1 Compression and Direct sum for bounded-round protocols

We start by proving our result about compressing bounded round protocols:

Proof of Corollary 1.2. The proof follows by applying our sampling procedure to the correlated pointer jumping instance $F_\pi(x,y,r)$. For each fixing of x, y, r, define the event $G_{x,y,r}$ to be the event E from Theorem 4.2. Then we have that $\mathbf{P}[G] > 1 - r\varepsilon$. Conditioned on G, we sample from

exactly the right distribution, and the expected communication of the protocol is

$$\mathop{\mathbf{E}}_{X,Y,R}\left[\mathbf{D}\left(F_\pi(X,Y,R)\right) + r\log(1/\varepsilon) + O(\sqrt{r\mathbf{D}\left(F_\pi(X,Y,R)\right)} + r)\right]$$

$$\leq \mathop{\mathbf{E}}_{X,Y,R}\left[\mathbf{D}\left(F_\pi(X,Y,R)\right)\right] + r\log(1/\varepsilon) + O\left(\sqrt{\mathop{\mathbf{E}}_{X,Y,R}\left[r\mathbf{D}\left(F_\pi(X,Y,R)\right)\right]} + r\right),$$

where the inequality follows from the concavity of the square root function. By Lemma 4.3, this proves that the expected communication conditioned on G is $\mathsf{IC}^{\mathsf{i}}{}_\mu(\pi) + r\log(1/\varepsilon) + O\left(\sqrt{r\mathsf{IC}^{\mathsf{i}}{}_\mu(\pi)} + r\right)$. □

5.2 Information = amortized communication

In this section we will show that Theorem 4.2 reveals a tight connection between the amount of information that has to be revealed by a protocol computing a function f and the amortized communication complexity of computing many copies of f. Recall that $\mathsf{IC}^{\mathsf{i}}{}_\mu(f,\rho)$ denotes the smallest possible internal information cost of any protocol computing f with probability of failure at most ρ when the inputs are drawn from the distribution μ. Observe that $\mathsf{IC}^{\mathsf{i}}{}_\mu(f,\rho)$ is an infimum over all possible protocols and may not be achievable by any individual protocol. It is also clear that $\mathsf{IC}^{\mathsf{i}}{}_\mu(f,\rho)$ may only increase as ρ decreases.

We first make the following simple observation.

Claim 5.1. *For each f, ρ and μ,*

$$\lim_{\alpha\to\rho}\mathsf{IC}^{\mathsf{i}}{}_\mu(f,\alpha) = \mathsf{IC}^{\mathsf{i}}{}_\mu(f,\rho)$$

Proof. The idea is that if we have any protocol with internal information cost I, error δ and input length ℓ, for every ε we can decrease the error to $(1-\varepsilon)\delta$ at the cost of increasing the information by at most $\varepsilon\cdot\ell$ just by using public randomness to run the original protocol with probability $1-\varepsilon$, and with probability ε, run the trivial protocol where the players simply exchange their inputs. Thus as α tends to ρ, the information cost of the best protocols must tend to each other. □

Next we define the amortized communication complexity of f. We define it to be the cost of computing n copies of f with error ρ in each coordinate, divided by n. Note that computing n copies of f with error ρ in each coordinate is in general an easier task than computing n copies of f with probability of success $1-\rho$ for all copies. We use the notation $D_\rho^{\mu,n}(f)$ to denote the communication complexity for this task, when the inputs for each coordinate are sampled according to μ. $D_\rho^{\mu,n}(f)$ was formally defined in Definition 2.16.

It is trivial to see in this case that $D_\rho^{\mu,n}(f) \leq n\cdot D_\rho^\mu(f)$. The amortized communication complexity of f with respect to μ is the limit

$$\mathsf{AC}(f_\rho^\mu) := \lim_{n\to\infty} D_\rho^{\mu,n}(f)/n,$$

when the limit exists. We prove an exact equality between amortized communication complexity and the information cost:

Theorem 5.2.

$$\mathsf{AC}(f_\rho^\mu) = \mathsf{IC}^{\mathsf{i}}{}_\mu(f,\rho).$$

Proof. There are two directions in the proof:

$\underline{\mathsf{AC}(f^{\mu}_{\rho}) \geq \mathsf{IC}^{\mathsf{i}}{}_{\mu}(f,\rho).}$ This is a direct consequence of Theorem 2.17.

$\underline{\mathsf{AC}(f^{\mu}_{\rho}) \leq \mathsf{IC}^{\mathsf{i}}{}_{\mu}(f,\rho).}$ Let $\delta > 0$. We will show that $D^{\mu,n}_{\rho}(f)/n < \mathsf{IC}^{\mathsf{i}}{}_{\mu}(f,\rho) + \delta$ for all sufficiently large n.

By Claim 5.1 there is an $\alpha < \rho$ such that $\mathsf{IC}^{\mathsf{i}}{}_{\mu}(f,\alpha) < \mathsf{IC}^{\mathsf{i}}{}_{\mu}(f,\rho) + \delta/4$. Thus there is a protocol π that computes f with error $< \alpha$ with respect to μ and that has an internal information cost bounded by $I := \mathsf{IC}^{\mathsf{i}}{}_{\mu}(f,\rho) + \delta/2$. Denote by C the communication complexity of π. C can be very large compared to I. For every n, the protocol π^n that is comprised of n independent copies of π that are executed in parallel, computes n copies of f as per Definition 2.16 with error bounded by α.

The internal information cost of π^n is $n \cdot I$, and by Theorem 4.2 we can simulate π^n with a total error $\varepsilon < \rho - \alpha$ using

$$C_n := n \cdot I + C \cdot \log 1/\varepsilon + O(\sqrt{C \cdot I \cdot n} + C)$$

bits of communication. The total additional error is ε and hence the new protocol makes at most an error $\alpha + \varepsilon < \rho$ on each copy of f. Hence $D^{\mu,n}_{\rho}(f) \leq C_n$. By letting n be large enough (with respect to C and $1/\varepsilon$) we see that we can make $D^{\mu,n}_{\rho}(f) \leq C_n < n \cdot I + n\delta/2$, thus completing the proof. □

5.3 A complete problem for direct sum

Let f^n denote the function mapping n inputs to n outputs according to f. We will show that the promise version of the correlated pointer jumping problem is complete for direct sum. In other words, if near-optimal protocols for correlated pointer jumping exist, then direct sum holds for all promise problems. On the other hand, if there are no near-optimal protocols for correlated pointer jumping, then direct sum fails to hold, with the problem itself as the counterexample. Thus any proof of direct sum for randomized communication complexity must give (or at least demonstrate existence) of near-optimal protocols for the problem.

We define the $\mathsf{CPJ}(C, I)$ promise problem as follows.

Definition 5.3. The $\mathsf{CPJ}(C, I)$ is a promise problem, where the players are provided with a *binary* instance[4] F of a C-round pointer jumping problem, i.e. player P_x is provided with the distributions $\mathsf{child}(v)_x$ and P_y is provided with the distributions $\mathsf{child}(v)_y$ for each v, with the following additional guarantees:

- the divergence cost $\mathbf{D}(F) \leq I$;
- let μ_F be the correct distribution on the leafs of F; each leaf z of F are labeled with $\ell(z) \in \{0,1\}$ so that there is a value $g = g(F)$ such that $\mathbf{P}_{z \in_{\mathrm{R}} \mu_F}[\ell(z) = g(F)] > 1 - \varepsilon$, for some small ε. The goal of the players is to output $g(F)$ with probability $> 1 - 2\varepsilon$.

Note that players who know how to sample from F can easily solve the CPJ problem. It follows from [BBCR10] that:

Theorem 5.4. *If $\mathsf{CPJ}(C, I)$ has a randomized protocol that uses $T(C, I) := \mathsf{CC}(\mathsf{CPJ}(C, I))$ communication, so that $T(C, C/n) < C/k(n)$, then for each f,*

$$\mathsf{CC}(f^n) = \Omega(k(n) \cdot \mathsf{CC}(f)).$$

[4] Each vertex has degree 2.

In [BBCR10] a bound of $T(C,I) = \tilde{O}(\sqrt{C \cdot I})$ is shown, which implies $\mathsf{CC}(f^n) = \tilde{\Omega}(\sqrt{n} \cdot \mathsf{CC}(f))$ for any f. Using Theorem 4.2 we are able to prove the converse direction.

Theorem 5.5. *For any $C > I > 0$, set $n := \lfloor C/I \rfloor$, then*

$$\mathsf{CC}(\mathsf{CPJ}(C,I)^n) = O(C \log(nC/\varepsilon)).$$

Thus, if there are parameters C and n such that $\mathsf{CPJ}(C, C/n)$ cannot be solved using $I = C/n$ communication, i.e. $T(C, C/n) > C/k(n) \gg C/n$, then $\mathsf{CPJ}(C, C/n)$ is a counterexample to direct sum, i.e.

$$\mathsf{CC}(\mathsf{CPJ}(C,I)^n) = O(C \log nC/\varepsilon) = \tilde{O}(C) = \tilde{O}(k(n)\mathsf{CC}(\mathsf{CPJ}(C, C/n))) = o(n \cdot \mathsf{CC}(\mathsf{CPJ}(C, C/n))).$$

Proof. (of Theorem 5.5) We solve $\mathsf{CPJ}(C,I)^n$ by taking $m := n \log n$ copies of the $\mathsf{CPJ}(C,I)$ problem representing $\log n$ copies of each of the n instances. The players will compute all the copies in parallel with error $< 2\varepsilon$, and then take a majority of the $\log n$ copies for each instance. For a sufficiently large n this guarantees the correct answer for all n instances except with probability $< \varepsilon$. Thus our goal is to simulate m copies of $\mathsf{CPJ}(C,I)$. We view $\mathsf{CPJ}(C,I)^m$ as a degree-2^m, C-round correlated pointer jumping problem in the natural way. Each node represents a vector $V = (v_1, \ldots, v_m)$ of m nodes in the m copies of $\mathsf{CPJ}(C,I)$. The children of V are the 2^m possible combinations of children of $\{v_1, \ldots, v_m\}$. The distribution on the children is the product distribution induced by the distributions in $v_1, \ldots, v_m$. We claim that

$$\mathbf{D}\left(\mathsf{CPJ}(C,I)^n_{v_1,\ldots,v_m}\right) = \sum_{i=1}^{m} \mathbf{D}\left(\mathsf{CPJ}(C,I)_{v_i}\right). \tag{2}$$

This follows easily by induction on the tree, since the distribution on each node is a product distribution, and for each independent pairs $(P_1, Q_1), \ldots, (P_m, Q_m)$ we have

$$\mathbf{D}\left(P_1 \times P_2 \times \ldots \times P_m || Q_1 \times Q_2 \times \ldots \times Q_m\right) = \mathbf{D}\left(P_1 || Q_1\right) + \ldots + \mathbf{D}\left(P_m || Q_m\right),$$

by Lemma 2.9. By applying (2) to the root of the tree we see that $\mathbf{D}\left(\mathsf{CPJ}(C,I)^m\right) \leq m \cdot I \leq C \log n$. Thus Theorem 4.2 implies that $\mathsf{CPJ}(C,I)^n$ can be solved with an additional error of $\varepsilon/2$ using an expected

$$C \log n + C \log C/\varepsilon + o(C \log n)$$

bits of communication. ∎

6 Acknowledgments

We thank Boaz Barak and Xi Chen for useful discussions.

References

[BBCR10] B. Barak, M. Braverman, X. Chen, and A. Rao. How to compress interactive communication. In *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing*, 2010.

[BYJKS04] Z. Bar-Yossef, T. S. Jayram, R. Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *Journal of Computer and System Sciences*, 68(4):702–732, 2004.

[CSWY01] A. Chakrabarti, Y. Shi, A. Wirth, and A. Yao. Informational complexity and the direct sum problem for simultaneous message complexity. In B. Werner, editor, *Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science*, pages 270–278, Los Alamitos, CA, Oct. 14–17 2001. IEEE Computer Society.

[CT91] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley series in telecommunications. J. Wiley and Sons, New York, 1991.

[FKNN91] T. Feder, E. Kushilevitz, M. Naor, and N. Nisan. Amortized communication complexity. *SIAM Journal on Computing*, 24(4):736–750, 1995. Prelim version by Feder, Kushilevitz, Naor FOCS 1991.

[GF81] G. Galbiati and M. Fischer. On the complexity of 2-output boolean networks. *Theor. Comput. Sci.*, 16:177–185, 1981.

[HJMR07] P. Harsha, R. Jain, D. A. McAllester, and J. Radhakrishnan. The communication complexity of correlation. In *IEEE Conference on Computational Complexity*, pages 10–23. IEEE Computer Society, 2007.

[JRS03] R. Jain, J. Radhakrishnan, and P. Sen. A direct sum theorem in communication complexity via message compression. In J. C. M. Baeten, J. K. Lenstra, J. Parrow, and G. J. Woeginger, editors, *ICALP*, volume 2719 of *Lecture Notes in Computer Science*, pages 300–315. Springer, 2003.

[JRS05] R. Jain, J. Radhakrishnan, and P. Sen. Prior entanglement, message compression and privacy in quantum communication. In *IEEE Conference on Computational Complexity*, pages 285–296. IEEE Computer Society, 2005.

[KN97] E. Kushilevitz and N. Nisan. *Communication complexity*. Cambridge University Press, Cambridge, 1997.

[KRW91] M. Karchmer, R. Raz, and A. Wigderson. Super-logarithmic depth lower bounds via the direct sum in communication complexity. *Computational Complexity*, 5(3/4):191–204, 1995. Prelim version CCC 1991.

[Pau76] W. Paul. Realizing boolean functions on disjoint sets of variables. *Theor. Comput. Sci.*, 2:383–396, 1976.

[Raz95] R. Raz. A parallel repetition theorem. *SIAM Journal on Computing*, 27(3):763–803, June 1998. Prelim version in STOC '95.

[Sha48] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27, 1948. Monograph B-1598.

[Sha03] R. Shaltiel. Towards proving strong direct product theorems. *Computational Complexity*, 12(1-2):1–22, 2003.

[Uhl74] D. Uhlig. On the synthesis of self-correcting schemes from functional elements with a small number of reliable elements. *Matematicheskie Zametki*, 15(6):937–944, 1974.

[Yao82] A. C.-C. Yao. Theory and applications of trapdoor functions (extended abstract). In *FOCS*, pages 80–91. IEEE, 1982.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de