Some Conversations Simply Cannot Be Concise: Separations between Information and Communication Complexity for Protocols

Dimitris Tsipras

May 6, 2016

Abstract

We survey the recent developments in the quantitative relation between communication and information complexity of interactive tasks. We briefly present state-of-the-art compression results, and then focus on an intuitive exposition of impossibility results for compressing protocols down to their information cost. We discuss which results and assumptions have room for improvement and pose a series of open questions.

1 Introduction

In the communication complexity setting, two parties, Alice and Bob, are each given an input and are asked to carry out a computation based on both inputs. In order to share information and compute the desired output, the parties agree on a (possibly randomized) protocol π. We measure the efficiency of the protocol using its communication cost $\mathtt{CC}(\pi)$, that is the maximum (over inputs) number of bits exchanged during its execution. Naturally, one might wonder how is the necessary amount of communicated bits related to the required amount of information exchange.

In order to even define such a notion we need to assume that inputs are sampled according to some joint, publicly known distribution μ. Then, we define the internal (resp. external) information cost of a protocol $\mathtt{IC}_\mu(\pi)$ (resp. $\mathtt{IC}^o_\mu(\pi)$), to be the number of information bits revealed to an internal (resp. external) observer. For one-round, deterministic protocols, any protocol can be compressed to the amount of information carried. This is a fact know for more than half a century, from the seminal works of Shannon and Huffman, [Sha48, Huf52]. However, for interactive communication the answers are not as clear and elegant, and took a significantly longer time to be found.

A long line of work investigates how the information and communication complexity task relate to each other. Initially, a series of progressively better compression schemes were invented, providing incomparably different guarantees under incomparable assumptions. This gave hope that eventually it would be possible to construct "optimal" compression schemes, that could compress protocols all the way down to their information cost. Recently however, in a surprising line of work Ganor, Kol, and Raz [GKR14] explicitly showed the existence of a communication task with (extremely long) protocols of information cost $O(k)$ for some large k, requiring communication cost at least $O(2^k)$ to simulate with non-trivial error guarantees. The results was later proved also for boolean functions [GKR15b] and simplified by Rao and Sinha in [RS15]. Moreover, the same separation was shown for the case of external information complexity in [GKR15a].

The goal of the survey is to focus on the lower bound proofs and provide a self-contained and intuitive explanation. From there, we plan to investigate the finer aspects of the protocol that determine whether or not efficient compression is possible or not. We present insight, and pose open questions about the nature of guarantees that one could hope for, and under which restrictions these are achievable.

2 Preliminaries

In this section, we will formally define the notions outlined in the introduction. We will consider discrete random variables. We denote random variables with capital letters (e.g. A), and their respective values with lowercase (e.g. a). For succinctness, when A is distributed according to p we denote $\Pr_p[A = a]$ by $p(a)$.

2.1 Information Theory

We begin by stating some basic definitions and facts from information theory, a powerful tool for our analysis. For a complete exposition refer to [CT12].

Definition 2.1 (Entropy)**.** *The (Shannon)* entropy *of a random variable X is defined as*

$$H(X) := \sum_{x \in X} p(x) \log \left(\frac{1}{p(x)} \right) = \mathop{\mathbb{E}}_{x \in X} \left[\log \left(\frac{1}{p(x)} \right) \right],$$

while the entropy conditioned on Y as, $H(X \mid Y) := \mathbb{E}_{y \sim Y}[H(X \mid Y = y]$.

The entropy of a random variable captures the amount of uncertainty of its distribution and is used to measure the expected amount of information that a value reveals.

Definition 2.2 (Mutual Information)**.** *We define the* mutual information *between variables X and Y as*

$$I(X;Y) := H(X) - H(X \mid Y) = I(Y;X),$$

and we similarly define the mutual information of X and Y conditioned on Z as $I(X;Y \mid Z) := H(X \mid Z) - H(X \mid YZ)$.

Mutual information is exactly the amount by which the uncertainty of X decreases, when we condition on the variable Y. As a sanity check, observe that when X and Y are independent, $I(X;Y) = 0$, while $I(X;X) = H(X)$, since $H(X|X) = 0$.

We will measure the distance between distribution in two ways:

Definition 2.3 (Statistical Distance)**.** *For two distributions p, q, their* statistical distance *is defined as*

$$|p - q| = \max_{Q} |p(Q) - q(Q)|,$$

and we say that p and q are ϵ-close, denoted by $p \overset{\epsilon}{\simeq} q$, if $|p - q| \leq \epsilon$.

Definition 2.4 (KL-Divergence)**.** *For two distributions p, q, their* KL-divergence *is defined as*

$$D(p\|q) = \sum_{a} p(a) \log\left(\frac{p(a)}{q(a)}\right).$$

Observe that KL-divergence does not have all the properties a norm distance would have (i.e. it's not symmetric), but the following fact holds

Fact 2.5. *For any p, q distributions, $D(p\|q) \geq 0$.*

Additionally, by direct calculations one can show that

Fact 2.6. *For any random variables A, B, C, we have*

$$I(A : B \mid C) = \mathop{\mathbb{E}}_{c,b}[D\left(p(A \mid b, c)\|p(A \mid c)\right)].$$

In other words, the mutual information between A and B is the expected (over B) divergence of the distribution of A, when conditioning on a particular value b. Sometimes, calculating the KL-divergence can be tricky, and the following fact can simplify the derivations.

Fact 2.7. *For any p, q*

$$\mathop{\mathbb{E}}_{p(b)}[D(p(A \mid b)\|p(A)] \leq \mathop{\mathbb{E}}_{p(b)}[D(p(A \mid b)\|q(A)].$$

Intuively, this property captures the fact that conditioning on some random variable will cause less expected divergence from the original distribution then from any other distribution. Finally, the following rule greatly facilitates the analysis of mutual information when dealing with sequential events.

Fact 2.8 (Chain Rule)**.** *For random variables A_1, A_2, B, C,*

$$I(A_1A_2 : B \mid C) = I(A_1 : B \mid C) + I(A_2 : B \mid A_1C).$$

The chain rule states that the information A_1A_2 reveal about B is equal to the information A_1 reveals about B, plus the information A_2 reveals about B, conditioning on A_1.

2.2 Information and Communication Complexity

We will now formally define that efficiency measures in terms of information and communication for different protocols.

Definition 2.9 (Protocol Communication Cost)**.** *For a protocol π we define the* communication cost *of a public coin* protocol $\mathtt{CC}(\pi)$*, to be the maximum number of bits that can be transmitted in any execution.*

For a function, its communication complexity is the communication cost of the cheapest protocol computing it.

Definition 2.10 (Function Communication Complexity)**.** *Let some function $f: \mathcal{X} \times \mathcal{Y} \to \mathbb{Z}_k$, a distribution μ supported on $\mathcal{X} \times \mathcal{Y}$, and an error parameter ϵ. We define $D^\mu_\epsilon(f)$ to be the minimum over* deterministic *protocols communication complexity required to compute f with error probability at most ϵ when the inputs are sampled from μ. Moreover we define $R_\epsilon(f)$ to be the communication cost of the cheapest public coin protocol for f with error probability at most ϵ.*

At first glance, the two measures of function complexity seem diffent. This is not the case when we focus on μ being the worst case distribution.

Theorem 2.11 (Yao's Minimax)**.** $R_\epsilon(f) = \max_\mu D^\mu_\epsilon(f)$.

Since often randomized protocols are hard to reason about and thus difficult to construct lower bounds for, Yao's theorem allows us to come up with a specific distribution, for which no protocol can perform good on. This immediately translates to a lower bound for $R_\epsilon(f)$. We will follow exactly this approach in the lower bounds that follow.

Definition 2.12 (Protocol Information Cost)**.** *Let some function $f: \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, a distribution μ over $\mathcal{X} \times \mathcal{Y}$, and a (possibly randomized) protocol π for f. Let Π the random variable for the transcript of the protocol, then the* internal information cost *of π is defined as*

$$\mathtt{IC}(\pi) := \mathbf{I}(X : \Pi \mid Y) + \mathbf{I}(Y : \Pi \mid X),$$

while the external information cost *of π is defined as*

$$\mathtt{IC}^{\mathrm{o}}(\pi) := \mathbf{I}(XY : \Pi).$$

In the case of protocol using public coin flips, we modify the definition such that mutual information is also conditioned on the string of random coins. Intuitively, we expect the external cost to be always higher than the internal cost, since an external observation knew less information about x and y to begin with, while the participants can possibly infer information about the input of each other by looking at their own inputs. Moreover, we expect that in the case of product μ the quantities are indeed equal. This can be made formal as follows

Proposition 2.13. *For every protocol π, $\mathtt{IC}(\pi) \leq \mathtt{IC}^{\mathrm{o}}(\pi)$. Additionally, when μ is a product distribution, $\mathtt{IC}(\pi) \leq \mathtt{IC}^{\mathrm{o}}(\pi)$.*

3 Known Compression Schemes

In order to facilitate the understanding of what is possible and what is not, we present a brief outline of the best known compression schemes for various settings. In [BBCR10], Barak, Braverman, Chen, and Rao show that for any protocol π and distribution μ, there exist compressed protocol τ equivalent to π (except with probability some small constant ϵ), with communication complexity

$$\mathtt{CC}(\tau) \leq \begin{cases} O\left(\sqrt{\mathtt{CC}(\pi)\mathtt{IC}_\mu(\pi)}\log(\mathtt{CC}(\pi)\right) \\ O\left(\mathtt{IC}^\circ_\mu(\pi)\log(\mathtt{CC}(\pi)\right). \end{cases}$$

These guarantees are incomparable, since a priori there are no concrete, quantitative bounds for the relation between $\mathtt{IC}^\circ$ and $\mathtt{IC}$. For the case of product distribution μ, we know that $\mathtt{IC}^\circ = \mathtt{IC}$ and therefore the second compression schemes is superior. The fact that the cost depends on the original communication cost only logarithmically does not say much by itself, since this cost can be e.g. quadratically exponential in the information cost.

An esssentially orthogonal result of Braverman [Bra12], shows the existence of a protocol τ such that,

$$\mathtt{CC}(\tau) \leq 2^{O(\mathtt{IC}_\mu(\pi))}.$$

Note that this compression scheme has no dependence on the communication cost of the original protocol. This implies that arbitrarily long protocols leaking a small amount of information bits can still be compressed independently of their initial length, in contrast to the guarantees from [BBCR10].

Another orthogonal results of Braverman and Rao [BR11] states that if π has r rounds we can construct a compressed protocol τ with,

$$\mathtt{CC}(\tau) \leq O(\mathtt{IC} + r).$$

While this results may seem interesting for small r, most essentially interactive protocols have a number of rounds close to the communication cost (or at most logarithmic factors away).

When focusing on restricted protocols and distributions, two more results are known. In [Kol15], Kol proves that given any protocol with information cost I, there is a way to compress it to a protocol with communication cost of at most $O(I^2\text{polylog}(I))$ bits. This is the first known protocol that relates information and communication cost polynomially. In [BMY15], Bauer, Moran, and Yehudayoff, show that any public coin protocol with information I and communication C can be simulated by a protocol with communication cost $O(I^2 \log\log C)$. We will further discuss the implications of these results towards the end of the survey.

4 Exponential Separation

In this section we will present an explicit boolean function, along with a distribution over inputs, such that a low information protocol exists, but every correct protocol needs communication exponential to that information cost. The first time such a construction appeared (albeit not for boolean functions) was in [GKR14]. Soon after, in [GKR15a], the same construction was modified to work for boolean function, while simultaneously having a simpler analysis. In [RS15], a construction with a further simplified analysis is presented, while morally staying close to the original paper. We will present the construction of [RS15], along with an overview of the analysis. We focus on providing clear intuition and motivation, and we therefore skip formal and involved proofs. We briefly present the original construction and follow with a discussion, outlining the similarities between the two.

4.1 The Problem

We will be concerced with the k-ary pointer jumping problem. Our alphabet with be $[k] = \{1, \ldots, k\}$ for some parameter k. The input of the protocol are functions $X, Y : [k]^{<n} \to [k]$ and boolean functions $F, G : [k]^n \to \{0, 1\}$. Alice is given X, F and Bob is given Y, G. By definition, there is a unique $z \in [k]^n$ such that for every $i \in \{0, \ldots, n-1\}$,

$$X(z_{\leq i}) + Y(z_{\leq i}) = z_{i+1} \pmod{k}.$$

The parties are asked to compute $F(z) + G(z) \pmod{2}$, and are allowed to err with probability at most ϵ for some constant parameter $\epsilon > 0$. The trivial protocol for this task consists of starting with z being the empty string, and at step i exchange $X(z_{<i})$ and $Y(z_{<i})$ to compute the next letter z_i. The communication cost of this protocol is $O(n \log k)$. It is straightforward to see that, for this protocol, information cost equals communication cost, and no other protocol can succeed with probability less than 1/2, for general input distributions (e.g. X, F, Y, G uniformly random). We will now define that distribution μ over $\mathcal{X} \times \mathcal{Y}$, with two goals in mind:

1. we want the output value of a correct protocol to depend on a small number of critical input bits. This will guarantee, that the parties learn only a few bits of each others inputs, achieving a communication of $O(\log k)$,

2. the only way for a protocol to perform better than the trivial one, is to figure out these critical bits, which will require at least $O(\log n)$ communication (since there are will be $O(n)$ possible values for them).

Setting $n = 2^k$, will result in a protocol of information cost $O(\log k)$ for a communucation task requiring communication cost $O(k)$. We say that some $z \in [k]^n$ is *consistent* with respect to J, X, Y if $X(z_{\leq J}) + Y(z_{\leq J}) = z_{j+1} \pmod{k}$. The distribution μ is defined according to the following procedure:

- Sample J uniformly at random from $\{0, \ldots, n-1\}$.

- Sample X uniformly at random from $[k]^{<n} \to [k]$.

- Sample Y as follows

$$Y(z) = \begin{cases} X(z) & \text{, if } |z| < J \\ \text{uar from [k]} & \text{, if } |z| = J \\ X(z) & \text{, if } |z| > J \text{ and } z \text{ is consistent} \\ \text{uar from [k]} & \text{, otherwise.} \end{cases}$$

- Sample a uniformly random bit $b \in \{0,1\}$.
- Sample F, G uniformly at random from $[k]^n \to \{0,1\}$, subject to the constraint that $F(z) + G(z) = b \pmod 2$ for every consistent z.

When inputs are sampled from this distribution, we can see that the protocols with error probability less than $1/2$ are exactly those that compute a consistent z. If the parties evaluate $F(z) + G(z) \pmod 2$ for any non-consistent z, they will compute a uniform value in $\{0,1\}$, while any consistent z will result in the same output b as the definition of the problem, thus being correct. Finally, notice that the distribution is symmetric w.r.t. X and Y.

4.2 A Low Information Protocol

Our goal is to design a protocol that utilizes the input distribution to compute a consistent z with low information cost. Observe that an protocol outputting some z is correct as long as $X(z_{\leq J}) + Y(z_{\leq J}) = z_{j+1} \pmod k$. Additionally, as long as both parties compute a consistent z, for any $i \neq J$ we know that $X(z \leq i) = Y(z \leq i)$, and therefore they know all the relevant inputs of the other party.

A first attempt: At first glance, it might seem that the trivial protocol has low information complexity. In fact, at every step, except for the J-th, Alice and Bob expect their messages to be the same (both X and Y have the same value), so they obtain no additional information. However, under more careful examination we see that with probability $1 - 1/k$ the messages at step J will be different. This implies that with probability $1 - 1/k$ they will learn J, since this is the only step where their messages can be different. By learning J, since J is distributed u.a.r. from n values, they will learn $\log n$ bits of information.

Hiding J: In order to prevent themselves from discovering the value of J, the parties will modify the protocol. Instead of sending the true function value at every step, they send a uniformly random value in $[k]$ with probability ϵ, and the real value otherwise. This implies that in expectation about ϵn messages will contain different values, and therefore the information obtained about J is roughly $\log(1/\epsilon)$ (we will make this point formal soon). Notice that for small enough ϵ we are still able to compute a consistent z with good probability. However, we are still not done. With probability ϵ, Alice and Bob will compute a non-consistent z. This implies that from this point on their messages will be different with probability $1 - 1/k$ and will reveal their inputs. In this case the information revealed will be $O(\epsilon n \log k)$, which is prohibitive for our separation.

Aborting: We will need one last fix for the protocol. Since we cannot hope the protocol to succeed when the parties compute an inconsistent z, we need to make sure that they detect this case and abort before they learn too much information. We therefore set some tolerance r, and insist that if the parties experience r consecutive messages where their inputs defer, to abort the protocol (and output some arbitrary answer). We outline the complete protocol in Protocol 1.

Protocol 1 A Low Information Protocol for k-ary Pointer Jumping

```
$z \leftarrow [\,]$
for $i \leftarrow \{1, \ldots n\}$ do
    $a_i = \begin{cases} X(z_{<i}) & \text{, w.p. } 1-\epsilon \\ \text{u.a.r. from } [k] & \text{, w.p. } \epsilon \end{cases}$
    $b_i = \begin{cases} Y(z_{<i}) & \text{, w.p. } 1-\epsilon \\ \text{u.a.r. from } [k] & \text{, w.p. } \epsilon \end{cases}$
    Exchange $m_i = a_i, b_i$;
    Append $((a_i + b_i) \mod k)$ to $z$;
    if $a \neq b$ during the last $r$ rounds then
        Abort Protocol;
Alice sends $f(z)$;
Bob outputs $f(z) + g(z) \pmod 2$
```

We first compute the probability of success for our protocol.

Lemma 4.1. *Protocol 1 succeeds with probability at least* $1 - (2\epsilon + n(2\epsilon)^r)$.

Proof. The consistency of z is determined only by step J. The probability of either party sending random values at step J is (by union bound) at most 2ϵ. Conditioning on the event that on step J both parties send their correct values, we will bound the probability of aborting before $|z| = n$. Since all the values of X and Y are the same it suffices to bound the number of consecutive rounds in which random bits were sent. For any fixed round k, the probability of sending random bits in all the previous r rounds up to k is at most $(2\epsilon)^r$. By union bounding over all the rounds, we get that the probability of aborting in any of them is at most $n(2\epsilon)^r$. Union bounding over these two bad event concludes the proof. □

By setting $r = \left\lceil \frac{\log n}{\log(1/2\epsilon)} + 1 \right\rceil$, the previous lemma implies that the error probability is at most 4ϵ. It remains to bound the information cost of the protocol.

Lemma 4.2. *Let M be the random variable for the messages of Protocol 1. Then,*

$$\left.\begin{array}{l} I(M : XF \mid YG) \\ I(M : YG \mid XF) \end{array}\right\} \leq 2\log(k/\epsilon)\left(1 + 2\epsilon \cdot \log n \cdot 2^{\frac{2\log n}{k\log(1/2\epsilon)}}\right)$$

Proof. We will prove the bound for Alice and the argument for Bob follows by symmetry. By applying the chain rule, we can write the total information cost as the marginal contribution of every message,

$$\begin{aligned} I(M : XF \mid YG) &= \sum_i I(M_i : XF \mid YGM_{<i}) \\ &= \sum_i \mathop{\mathbb{E}}_{xfygm} [D(p(M_i \mid xfygm_{<i}) \| p(M_i \mid ygm_{<i})]. \end{aligned}$$

By using Fact 2.7, we can consider the distribution $p(M_i \mid ygm_{<i}(x_i = y_i))$ instead of $p(M_i \mid ygm_{<i})$ in the display above, and still get an upper bound for the information. It turns out this distribution is easier to understand and reason about. Therefore we get,

$$I(M_i : XF \mid YGM_{<i}) \leq \mathop{\mathbb{E}}_{xfygm} [D(p(M_i \mid xfygm_{<i}) \| p(M_i \mid ygm_{<i}(x_i = y_i)].$$

The divergence can be simply bound in the following way:

- If $x_i = y_i$, the distributions are equal and therefore $I(M_i : XF \mid YGM_{<i}) = 0$,
- If $i = n+1$, $I(M_i : XF \mid YGM_{<i}) = 1$, since players unconditionally learn the values of $F(z), G(z)$ for each other,
- If $x_i \neq y_i$, Alice expects to receive x_i w.p. $(1 - \epsilon + \epsilon/k)$ and anything else w.p. ϵ/k, while the actual message will be y_i w.p. $(1 - \epsilon + \epsilon/k)$ and anything else w.p. ϵ/k. By direct calculations.

$$\begin{aligned} I(M_i : XF \mid YGM_{<i}) &= (1 - \epsilon + \epsilon/k) \log \frac{1 - \epsilon + \epsilon/k}{\epsilon/k} + (\epsilon/k) \log \frac{\epsilon/k}{1 - \epsilon + \epsilon/k} \\ &\leq \log(k/\epsilon). \end{aligned}$$

The proof is almost complete. If we define the random variable Q_i to be 1 iff $X(Z_{<i}) \neq Y(Z_{<i})$ and zero otherwise, invoking the case analysis above we conclude that

$$I(M : XF \mid YG) \leq 1 + \log(k/\epsilon) \cdot \mathbb{E}\left[\sum_i Q_i\right].$$

Recall that by definition $Q_i = 0$ for $i < J$. Moreover, when the protocol samples a consistent z, $Q_i = 0$ for $i \neq J$, and thus $\sum_i Q_i \leq 1$. Finally, the probability of sampling an inconsistent z is at most 2ϵ, and in this case we need to bound the number of steps after the J-th before the protocol termination. The probability of stopping after r steps is $(1 - 1/k)^r$. Thus the expected number of steps until stopping is at most $\frac{r}{(1-1/k)^r}$. Replacing the value of r concludes the proof. □

4.3 A Lower Bound to the Communication Cost

As mentioned earlier, intuitively, any protocol with error probability less than $1/2$, has to output a consistent string z with good probability, otherwise $F(z) + G(z) \pmod 2$ is uniformly random in $\{0, 1\}$. Recall that by Yao's theorem it suffices to consider only deterministic protocols, since we are considering inputs sampled from a distribution. We will prove that any protocol with communication significantly less than $O(\log n)$, learns very little information about which string are consistent. In order to conclude the lower bound to the communication complexity of k-ary pointer jumping, one needs to leverage this fact to show that such low communication protocols produce a distribution over messages such that

$$p(m \mid b = 0) \overset{\epsilon}{\simeq} p(m) \overset{\epsilon}{\simeq} p(m \mid b = 1),$$

where b is the correct answer, as described in the distribution over inputs (Section 4.1). This part of the proof is a rather involved series of technical claims, so we are not going to go over it (the full proof can be found in Section 4 of [RS15]). Instead, we will suffice to the following lemma, showing that parties learn little about the set of consistent string when restricting to low communication cost.

Lemma 4.3. *Let any protocol with communication cost at most ℓ, and let S the (random) set of consistent strings. Then,*

$$\left.\begin{array}{l} \mathbf{I}(M : S \mid Y_{\leq J} J) \leq \mathbf{I}(M : X_J \mid Y_{\leq J} J) \\ \mathbf{I}(M : S \mid X_{\leq J} J) \leq \mathbf{I}(M : Y_J \mid X_{\leq J} J) \end{array}\right\} \leq \frac{2^\ell}{n}$$

Proof. We will prove only the first set of inequalities, as the second follows by symmetry. We know that fixing $Y_{\leq J}$ and J, S depends only on X_J, implying the $\mathbf{I}(M : S \mid Y_{\leq J} J) \leq \mathbf{I}(M : X_J \mid Y_{\leq J} J)$. Moreover, by the definition of μ, $X_{<J} = Y_{<J}$ and thus, $\mathbf{I}(M : X_J \mid Y_{\leq J} J) = \mathbf{I}(M : X_J \mid Y_J X_{<J} J)$. For every possible transcript m of the protocol, there exists a set $S_m \times T_m$ of inputs such that $M = m$ iff the input of Alice is from the set S_m and the input of Bob from T_m. Moreover after fixing $x_{<j} j$, X_j is independent of Y and thus,

$$\begin{aligned} \mathbf{I}(M : X_J \mid Y_J X_{<J} J) &= \sum_m p(m) \mathop{\mathbb{E}}_{yxj \mid m} [D(p(X_j \mid m y_j x_{<j} j) \| p(X_j \mid y_j x_{<j} j))] \\ &\leq \sum_m p(S_m) \mathop{\mathbb{E}}_{xj \mid S_m} [D(p(X_j \mid S_m x_{<j} j) \| p(X_j \mid x_{<j} j))] \end{aligned}$$

Finally, since J is independent of X (and thus S_m) we can apply chain rule to rewrite the last display as

$$\frac{1}{n} \sum_m p(S_m) \mathop{\mathbb{E}}_{S_m} [D(p(X_j \mid S_m) \| p(X_j))]$$

which can be further bound by

$$\frac{1}{n} \sum_m p(S_m) \log \frac{1}{p(S_m)} \leq \frac{2^\ell}{n},$$

since there are at most 2^ℓ rectangles.

□

4.4 The original construction

We will briefly discuss the construction of [GKR14, GKR15a] that originally proved the separation result. The goal is to construct a function that can be computed with low information cost but cannot have low communication cost.

We define the *bursting noise* function, defined on a binary tree of depth 2^{4^k} for some parameter k. Each player will receive a bit $\in \{0,1\}$ (x_v for Alice and y_v for Bob) for every node v in the tree. We interpet the inputs of the players as alternatingly defining a path on the tree, i.e. starting from the root we move to the left child if $x_v = 0$ and right otherwise, then we move left if $y_v = 0$ and so on and so forth. This procedure uniquely defines a leaf ℓ of the tree. The parties are required to output $x_\ell + y_\ell \pmod 2$. We say that Alice (resp. Bob) *owns* vertices at odd (resp. even) depth.

In order to define the distribution over input, we group together tree layers in groups of $O(k)$ and call these *multi-layers*.We define an edge from a vertex v to be *correct* if it leads to the child indicated by the input of the party owning v. We say that a vertex is noisy if x_v, y_v are sampled independently, uniformly at random from $\{0,1\}$, and noiseless if they are sampled uniformly from $\{0,1\}$ subject to the constraint that $x_v = y_v$. The procedure for sampling an input distribution is as follows.

- sample a random multilayer ℓ^*, we call ℓ^* the noisy layer
- for vertices v in layers above ℓ^* we sample v noiselessly,
- for vertices v of layer ℓ^* we sample v noisily,
- for a node in the final layer of the multilayer ℓ^*, we call it *typical* if at least 80% of the edges – on the path to the root – belonging to the noisy layer, are correct. For every successor v of a typical node we sample v noiselessly and we sample successors of non-typical vertices noisily.
- for a leaf v, we call v typical if it the successor of a typical vertex. We sample b uniformly at random from $\{0,1\}$ and set $y_v \leftarrow y_v + b \pmod 2$ for all the typical leaves v

We observe that when inputs are sampled from this distribution, a protocol is correct with probability more than 1/2, if it outputs a typical leaf with good probability. This implies that any correct protocol, cannot communicate much less than $\log(2^{4^k}) = 4^k$, otherwise it will have insufficient information about ℓ^*. On the other hand, a protocol exchanging inputs for every layer of the tree, will only exchange different bits on the noisy multilayer, which only has $O(k)$ layers. Similarly to the construction in Section 4.1, we modify this protocol such that parties don't learn much about ℓ^*. That is, at every step, with probability 10% they send a random value instead of the real one. By Chernoff bounds, the protocol still succeeds with high probability. Finally, the parties abort if they experience two many different bits, such that they avoid leaking too much information in the case of sampling a non-typical leaf.

5 Discussion and Open Questions

In the previous section we gave an explicit example of a function and a distribution over input for which communication and information are exponentially separated. This proves that indeed for general distributions and protocols, one cannot compress information to communication. It is however interesting to consider which specific distributions and protocols fall under this theorem. We discuss two very natural special cases for which this theorem provably fails and present some natural open questions.

5.1 Product Distributions

When we defined our distribution over inputs in Section 4.1, we correlated the inputs of the two parties in a very specific way. For the vast majority of relevant inputs, we setup the distribution such that Alice is almost certain about the message she will receive from Bob and vice versa. By giving the two parties almost identical (for consistent strings) functions X and Y, we ensure that only a small amount of information is leaked during the execution of the protocol.

One might wonder if this still holds when the inputs of Alice and Bob are sampled independently. This is not the case, as shown in [Kol15]. Given any protocol with information cost I, there is a way to compress it to a protocol with communication cost of at most $O(I^2\text{polylog}(I))$ bits. This implies that we cannot hope to separate information and communication exponentially.

> **Open Question:** Is there a communication task with information complexity I, for which no protocol with communication cost $O(I\text{polylog}(I))$ exists?

In a strong sense, the result of [Kol15] implies that the strong correlation between inputs for problems in Sections 4.1,4.4 was necessary to achieve the separation results.

5.2 Public Randomness

When one wants to construct a low communication protocol, using public coin flips can reduce communication and is at least as good as using private coin flips. Additionally, Newman's Theorem [New91] states that private coin protocols can be converted to public coin protocols with a mild (logarithmic in input size) increase in communication cost.

When trying to construct low information protocols, private random coins can help conceal information and are at least as good than public coin flips, since they can be published with zero information cost. In the separations of

Sections 4.1,4.4 private randomness was crucially utilized to prevent the parties from discovering the critical part of the function. Moving these coin flips to be public would immediately break the proof. One might hope however that a slightly different scheme can work. In [BMY15], it is shown that any public coin protocol with information I and communication C can be simulated by a protocol with communication cost $O(I^2 \log\log C)$. This implies that in order to achieve the same (exponential) separation for the k-ary pointer jumping problem using a *public coin* protocol of information cost $O(k)$, we need its communication to be at least triply exponential in k, while the input size is only exponential in k. However, while this shows that the existing constructions fail when randomness is made public, it leaves open the possibility that a different construction works.

> **Open Question:** Is there a communication task with information complexity I, for which no public coin protocol with communication cost $O(2^I)$ succeeds with arbitrarily high probability?

5.3 Simultaneous Information and Communication Guarantees

All the approaches that we presented so far, attempt to answer the question of how small can the communication complexity of a task be, compared to its information complexity. One might wonder whether one can simultaneously achieve low information and low communication cost. For the case of external information complexity current theorems can rule out this possibility to some extent. Consider a function with information complexity I. By the result of [Bra12], there exists a protocol π with communication cost $2^{O(I)}$. If this protocol has information complexity $C = O(I)$, by the compression scheme of [BBCR10], there exists a protocol with communication cost $O(I\text{polylog}(C)) = O(\text{poly}(I))$. Since there exists an exponential separation between external information and communication cost, there are tasks with protocols of information cost $O(I)$, that the compression scheme of [Bra12] compresses to protocols with information cost $2^{O(I)}$. This argument will not work for internal information complexity using theorems currently known.

> **Open Question:** Is there a scheme compressing protocols of information cost I to protocols of communication cost $2^{O(I)}$ and information cost $2^{\omega(I)}$?

References

[BBCR10] Boaz Barak, Mark Braverman, Xi Chen, and Anup Rao. How to compress interactive communication. In Leonard J. Schulman, editor, *STOC*, pages 67–76. ACM, 2010.

[BMY15] Balthazar Bauer, Shay Moran, and Amir Yehudayoff. Internal Compression of Protocols to Entropy. In Naveen Garg, Klaus Jansen, Anup Rao, and José D. P. Rolim, editors, *APPROX-RANDOM*, volume 40 of *LIPIcs*, pages 481–496. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.

[BR11] Mark Braverman and Anup Rao. Information Equals Amortized Communication. In Rafail Ostrovsky, editor, *FOCS*, pages 748–757. IEEE Computer Society, 2011.

[Bra12] Mark Braverman. Interactive information complexity. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 505–524, 2012.

[CT12] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[GKR14] Anat Ganor, Gillat Kol, and Ran Raz. Exponential Separation of Information and Communication. In *FOCS*, pages 176–185. IEEE Computer Society, 2014.

[GKR15a] Anat Ganor, Gillat Kol, and Ran Raz. Exponential Separation of Communication and External Information. *Electronic Colloquium on Computational Complexity (ECCC)*, 22:88, 2015.

[GKR15b] Anat Ganor, Gillat Kol, and Ran Raz. Exponential Separation of Information and Communication for Boolean Functions. In *STOC*, pages 557–566. ACM, 2015.

[Huf52] David A Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101, 1952.

[Kol15] Gillat Kol. Interactive Compression for Product Distributions. *Electronic Colloquium on Computational Complexity (ECCC)*, 22:168, 2015.

[New91] Ilan Newman. Private vs. common random bits in communication complexity. *Information processing letters*, 39(2):67–71, 1991.

[RS15] Anup Rao and Makrand Sinha. Simplified Separation of Information and Communication. *Electronic Colloquium on Computational Complexity (ECCC)*, 22:57, 2015.

[Sha48] Claude E Shannon. A mathematical theory of communication. *Bell System Tech. J*, 27:623, 1948.