Coding for interactive computation: progress and challenges

Mark Braverman*
Department of Computer Science
Princeton University
Email: mbraverm@cs.princeton.edu

Abstract

We highlight some recent progress and challenges in the area of interactive coding and information complexity.

1. Introduction

The purpose of this brief note is to outline some of the recent progress and open problems in the area of coding for interactive computation. The majority of the paper will focus on the two-terminal scenario, though we will briefly mention extensions to $k > 2$ terminals. The main difference between the interactive computation scenario and the standard transmission scenario, is that in the interactive setting the parties wish to perform (or compute) a general operation that depends on their inner states before the operation, and that is not necessarily a simple transmission task. For example, if before the interaction the state of the terminals if given by a distribution $(X,Y) \sim \mu$, the terminals may be required to compute a function $F(X,Y)$ that depends on both X and Y. In this language, the transmission problem from the first terminal to the second one is equivalent to computing the function $F(X,Y) = X$.

Studying the problem of interactive computing has at least two benefits. Firstly, it gives us a better picture on the ability of parties to perform complex tasks with given communication resources, and the ability of channels to support such tasks. Secondly, it allows one to extend the reach of information theory deeper into the realm of communication complexity lower bounds, and produce results in that area with the hope of obtaining new computational complexity lower bounds that would follow.

For the purpose of this presentation, we break the interactive computing problem into three main directions: interactive compression and noiseless coding, interactive worst-case noisy coding, and interactive channel capacity. Our main focus will be on compression problems and their connections to communication complexity, but we will present problems in all three directions.

*Partially supported by an Alfred P. Sloan Fellowship, an NSF CAREER award, and a Turing Centenary Fellowship.

1.1. Basic setup: communication complexity

The basic setup is that of communication complexity. There are two parties (terminals), Alice and Bob. Prior to the communication, they are given a pair of inputs (X,Y): Alice is given X and Bob is given Y. Depending on the context, the input pair (X,Y) may be drawn from a distribution μ. Alice and Bob need to perform a computational task $T(X,Y)$ on their inputs. Unless otherwise stated, we assume that they employ a noiseless binary channel. In communication complexity, the objective of Alice and Bob is to perform $T(X,Y)$ while sending as few bits as possible over the channel. Sometimes we are also interested in the number of rounds (i.e. the number of alternations between Alice and Bob speaking). A comprehensive introduction to communication complexity, and to the state-of-the-art as of 1997 can be found in [KN97].

Computation is performed by means of a protocol. A protocol is a sequence of functions which specify the next message to be sent (and the speaker). Thus a *deterministic* protocol (i.e. one that uses no randomness) is just a sequence of functions that map the transcript so far and the speaker's input to the next message. In *randomized* protocols the parties in addition are allowed to use sources of randomness. In general, it is important to distinguish between two types of randomness. *Public randomness* refers to a random string R that is shared by Alice and Bob. *Private randomness* refers to random strings R_A and R_B such that only Alice has access to R_A and only Bob has access to R_B. We assume that access to the shared randomness does not add to the communication cost. This is justified in the context of communication complexity by a generic reduction from public to private randomness [New91]. In the

communication complexity context, the opposite reduction is trivial: parties with access to public randomness can simulate private randomness. However, in some settings, particularly in the setting of *information complexity*, having access to private randomness may enable parties to solve problems more efficiently – allowing them to release their information in a more controlled manner during an interaction.

The communication complexity of a task T is the smallest number of bits that Alice and Bob need to exchange to perform T. The communication complexity usually refers to the *worst case* number of bits, although *average case communication complexity* is also sometimes considered.

Another useful way of looking at protocols is by means of a *protocol tree*. A protocol tree is a binary tree where to each node v one associates an owner $o(v) \in \{A,B\}$, and a function that maps the owner's input to the next message. Thus if $o(v) = A$ we will have a function $f_v : X \mapsto \{0,1\}$ in the deterministic setting, and a function $f_v : (X,R,R_A) \mapsto \{0,1\}$ in the randomized setting.

In the remainder of the paper, we outline some progress, and some problems that are, to the best of our knowledge, open, in three directions: (1) interactive information complexity; (2) interactive coding for adversarial noise; and (3) interactive coding for noisy channels. The unifying theme of these three directions is that we view the interactive task as one whole operation, without breaking it down into a series of one-way transmissions. While this complicates matters, it also raises opportunities for more efficient protocols as well as for a new lens through which the communication complexity of functions can be studied.

2. Interactive information complexity

We start with looking into the properties of interactive information complexity. A more detailed recent discussion on the topic can be found in [Bra12b], though some of the open problems presented here are new. While communication complexity is concerned with the number of bits Alice and Bob need to exchange to perform a task T, information complexity is concerned with the amount of information they need to exchange irrespective of the actual number of bits transmitted in the process. Informally speaking, the minimum amount of information that needs to be exchanged to perform T is the *information complexity* of T. We will also be interested in minimizing the amount of information about the inputs (X,Y) that needs to be revealed to an external observer in order to perform T. This quantity is referred to as the *external information complexity of* T. In addition to the connections of these quantities to communication complexity, they are also interesting in their own right in the context of information-theoretic privacy [Kla04]: where not revealing too much information is in fact the primary objective.

For the remainder of the section, unless otherwise stated, we assume that Alice and Bob are given a pair of inputs (X,Y) distributed according to a distribution μ. The have access to public and private randomness. Let π be any protocol. An execution of the protocol on the pair of (random) inputs $(X,Y) \sim \mu$ gives rise to the transcript $\Pi = \Pi(X,Y)$ of the protocol which is itself a random variable. The *information cost* of a protocol Π is the amount of information that the protocol reveals to Alice and Bob about their input. For example, the amount revealed to Alice – who knows X – about Y is given by the conditional mutual information $I(Y;\Pi|X)$. Thus the information cost of π is given by

$$\mathsf{IC}_\mu(\pi) := I(Y;\Pi|X) + I(X;\Pi|Y). \qquad (1)$$

The task of finding the communication complexity of a task T can be reformulated as the task of finding a low-communication protocol for computing T. In the same vein, we can use (1) to define the task of finding the *information complexity* of T as the task of minimizing the information complexity of the protocol for T:

$$\mathsf{IC}_\mu(T) := \inf_{\pi \text{ protocol performing } T} \mathsf{IC}_\mu(\pi). \qquad (2)$$

Note that we must take an inf and not a min in (2), because the protocol π can be arbitrarily long (as long as it doesn't reveal too much information to the parties). Thus $\mathsf{IC}_\mu(T)$ may not be achievable by any individual protocol, but only as a limit of a sequence of protocols $\pi_1, \pi_2, \ldots$ which increase in their length and decrease in their information cost. In fact, this is exactly the case even for such a simple task T as computing the AND of two bits $T(X,Y) = X \wedge Y$ [BGPW12a].

What gives $\mathsf{IC}_\mu(T)$ operational meaning is the fact that it is equal to the scaling limit of communication complexity:

Theorem 1 *([BR11b][1], somewhat loosely re-stated.) Let T^n be the task of performing n independent copies of the task T. Suppose that T allows for some non-zero error $\varepsilon > 0$. Then we have:*

$$\mathsf{IC}_\mu(T) = \lim_{n\to\infty} \mathsf{CC}_{\mu^n}(T^n)/n. \qquad (3)$$

[1] The '$\geq$' direction of Theorem 1 was independently proved by Ma and Ishwar [MI11, MIG12] using repeated application of the Wyner-Ziv compression scheme.

In particular, Theorem 1 applies when T is the task of computing a function $F(x,y)$ with success probability $1-\varepsilon$, where $\varepsilon > 0$. Moreover, the information complexity of tasks is additive over independent tasks, and subadditive over dependent ones [Bra12b]. We have

$$\mathsf{IC}_{\mu_1\times\mu_2}((T_1,T_2)) = \mathsf{IC}_{\mu_1}(T_1) + \mathsf{IC}_{\mu_2}(T_2), \qquad (4)$$

and

$$\mathsf{IC}_{\mu}((T_1,T_2)) \leq \mathsf{IC}_{\mu|_{(X_1,Y_1)}}(T_1) + \mathsf{IC}_{\mu|_{(X_2,Y_2)}}(T_2). \qquad (5)$$

Thus the properties of $\mathsf{IC}_\mu(T)$ make it a well-behaved complexity measure. Its properties and the connection to communication complexity make the study of information complexity worthwhile. While information complexity applies to general tasks, to keep the discussion simple we will focus on tasks of evaluating $(0/1)$-valued functions from now on. Thus $F : (X,Y) \to \{0,1\}$ will be a function, which Alice and Bob would like to evaluate correctly except with error $\varepsilon \geq 0$. We denote the information complexity of this task by $\mathsf{IC}_\mu(F,\varepsilon)$.

2.1. Computability of information complexity

The first problem that is (somewhat embarrassingly) open, is computing the information complexity from the truth table of F:

Problem 2 *Given the truth table of a function $F : (X,Y) \to \{0,1\}$, and error parameter $\varepsilon \geq 0$, and a distribution μ of (X,Y), can one give a general procedure for computing the information complexity $\mathsf{IC}_\mu(F,\varepsilon)$?*

We believe the answer to Problem 2 to be affirmative. As noted above, the problem is that there might be a sequence of protocols whose information cost decreases as protocol size increases. The $\leq$ direction of Theorem 1 gives one way to obtain a decreasing sequence that converges to $\mathsf{IC}_\mu(F,\varepsilon)$ by considering the amortized cost of n copies of F as $n \to \infty$. Unfortunately, for this procedure to compute $\mathsf{IC}_\mu(F,\varepsilon)$, we need to have an effective bound on the sequence's rate of convergence down to $\mathsf{IC}_\mu(F,\varepsilon)$.

The work of Ma and Ishwar [MI11] gives a computable characterization of $\mathsf{IC}_\mu(F,\varepsilon)$, but only when one fixes the number of rounds of interaction (back-and-forth messages) in advance. Once again, we do not know an effective rate of convergence of the round-restricted information complexity to the unrestricted value.

2.2. Compressibility of interactive computation

The next set of questions, which is somewhat related to the computability question above has to do with *compressibility of interactive computation*. If one were to view $\mathsf{IC}_\mu(F,\varepsilon)$ as the interactive analogue of Shannon's entropy, then Theorem 1 would be the analogue of the noiseless source coding theorem. The other piece of the puzzle, which is currently missing, is an analogue of Huffman coding – an efficient one-shot encoding scheme for low-information computations. We first formulate this as a question about the relationship between information and communication complexity. Note that in the non-interactive case, Huffman coding implies that the expected cost of sending one message X is bounded by $H(X)+1$. In the interactive case we ask:

Problem 3 *How close is the average case ε-error communication complexity $\mathsf{CC}^{average}_\mu(F,\varepsilon)$ of F to its information complexity $\mathsf{IC}_\mu(F,\varepsilon)$? In particular, is it true that*

$$\mathsf{CC}^{average}_\mu(F,\varepsilon) = O(\mathsf{IC}_\mu(F,\varepsilon))? \qquad (6)$$

The only general result for a bound on the communication complexity C in terms of the information complexity I that we currently have is of the form $C < 2^{O(I)}$ [Bra12b]. This is a far cry from the linear relationship we ask for in Problem 3. Our current understanding of Problem 3 is limited. In particular, while it is possible that the answer to the question is affirmative, it is also possible that the exponential bound is tight – i.e. that there are examples such that $C = 2^{\Omega(I)}$.

There is another viewpoint onto Problem 3. By Theorem 1, $\mathsf{IC}_\mu(F,\varepsilon)$ is equal to the amortized communication complexity of F as long as $\varepsilon > 0$. Thus (6) can be rephrased as

$$\mathsf{CC}^{average}_\mu(F,\varepsilon) = O(\mathsf{CC}_{\mu^n}(F^n,\varepsilon)/n)?$$

or, more familiarly, as

$$\mathsf{CC}_{\mu^n}(F^n,\varepsilon) = \Omega(n\cdot \mathsf{CC}^{average}_\mu(F,\varepsilon))? \qquad (7)$$

This latter formulation is known as the *direct sum problem* for distributional communication complexity. In general, the direct sum problem asks whether solving n copies of a problem is n times as hard as solving one copy. It is usually clear that it is at most as hard, but could there be savings in solving n copies in parallel?

In some other contexts savings indeed are attainable. Consider a fixed matrix $A \in \mathbb{R}^{n\times n}$ and the task of computing the product Av for vectors $v \in \mathbb{R}^n$. A simple counting argument shows that for most matrices this task requires a boolean circuit of size $\widetilde{\Theta}(n^2)$. At the same time, the task of computing $Av_1,\ldots,Av_n$ in parallel can be accomplished by a circuit of size $\widetilde{O}(n^\omega)$, where $\omega < 3$ is the matrix multiplication constant.

In the context of randomized communication complexity, the problem remains open, with the best currently known result [BBCR10] giving (for any constant

$\varepsilon > 0$)

$$\mathsf{CC}_{\mu^n}(F^n,\varepsilon) = \widetilde{\Omega}(\sqrt{n}\cdot \mathsf{CC}^{average}_{\mu}(F,\varepsilon)). \qquad (8)$$

In the special case of μ being a product distribution $\mu = \mu_X \times \mu_Y$ the statement (8) can be strengthened to

$$\mathsf{CC}_{\mu^n}(F^n,\varepsilon) = \widetilde{\Omega}(n\cdot \mathsf{CC}^{average}_{\mu}(F,\varepsilon)), \qquad (9)$$

which is tight up to polylogarithmic factors.

As mentioned above, yet another closely related way of rephrasing Problem 3 is in terms of protocol compression.

Problem 4 *Given a protocol π that has information cost $I = \mathsf{IC}_\mu(\pi)$ can π be simulated by another protocol π' whose* communication complexity $\mathsf{CC}_\mu(\pi') \approx I$*?*

It is not hard to see that such a compression scheme would imply an affirmative answer to Problem 3: we can just take a low-information complexity protocol for F and compress it into a low communication protocol. A converse statement is also somewhat true if we broaden Problem 3 to tasks, then we can define T as the task of simulating the protocol π. Completing T in low communication cost is equivalent to compressing π. A naïve attempt at performing such a compression would be to try and compress the protocol round-by-round. This doesn't work due to the 1-bit per round overhead that it introduces. This overhead turns out to be critical when π is highly interactive.

A slightly less ambitious goal is to compress interactive protocols when the compression is allowed to have a (weak) dependence on the communication complexity of the original protocol, in addition to the dependence on its information complexity. Let π be a protocol whose information cost is still $I = \mathsf{IC}_\mu(\pi)$, and whose (worst-case) communication cost is $C = \mathsf{CC}_\mu(\pi') \geq I$. For what functions $B(I,C)$ can we simulate π using a protocol π' whose communication cost is $\mathsf{CC}_\mu(\pi') = B(I,C)$? An ideal scenario would be $B(I,C) = O(I)$. Trivially, one has $B(I,C) \leq C$ (corresponding to $\pi' = \pi$). Different functions $B(I,C)$ correspond to different statements of the direct sum theorem. For example, the statement (8) follows from a compression scheme of the form

$$B(I,C) = O\left(\sqrt{I\cdot C}\,(\log C)^{O(1)}\right).$$

A stronger compression bound, e.g. one of the form $B(I,C) = \widetilde{O}\left(I^q\cdot C^{1-q}\right)$ for $q > 1/2$ would imply stronger direct sum theorems than currently known. For μ's that are product distributions, we have a stronger bound of $B(I,C) = O\left(I\cdot(\log C)^{O(1)}\right)$ [BBCR10], which implies the near-optimal direct sum theorem (9).

2.3. External information complexity

So far, we have focused on the information complexity of functions and tasks, which is the amount of information the parties need to exchange to perform the task. Another quantity of interest is the *external* information complexity of functions. External information complexity is arguably as natural a notion as information complexity, and it measures the amount of information the parties need to reveal to an external observer while performing the task. In fact, within the context of theoretical computer science it was historically considered before (internal) information complexity [CSWY01, BYJKS04]. Formally, the external information cost of a protocol π with inputs over a prior distribution μ is given by

$$\mathsf{IC}^{ext}_\mu(\pi) := I(XY;\Pi). \qquad (10)$$

It is not hard to see that the external information cost is always at least as high as the information cost of π:

$$\mathsf{IC}_\mu(\pi) \leq \mathsf{IC}^{ext}_\mu(\pi) \leq \mathsf{CC}_\mu(\pi). \qquad (11)$$

The notion of $\mathsf{IC}^{ext}_\mu(\pi)$ immediately gives rise to the notion of the external information complexity of tasks:

$$\mathsf{IC}^{ext}_\mu(T) := \inf_{\pi \text{ protocol performing } T} \mathsf{IC}^{ext}_\mu(\pi). \qquad (12)$$

We will focus of the task of computing a function F with error $\varepsilon \geq 0$, whose external information complexity is denoted by $\mathsf{IC}^{ext}_\mu(F,\varepsilon)$. We will be particularly interested in the scenario where $\varepsilon = 0$.

All the compression questions from the previous section can be restated for external information complexity. In light of (11) the following question is strictly easier than Problem 4:

Problem 5 *Given a protocol π that has information cost $I_{ext} = \mathsf{IC}^{ext}_\mu(\pi)$ can π be simulated by another protocol π' whose* communication complexity $\mathsf{CC}_\mu(\pi') \approx I_{ext}$*?*

We do not know the answer to Problem 5, although it may well be negative. However the gap is narrower than in the information complexity case (Problem 4). In particular, we know that a protocol with external information cost I_{ext} and communication cost C can be simulated using communication $O\left(I_{ext}\cdot(\log C)^{O(1)}\right)$ [BBCR10][2].

[2]For product distributions $\mu = \mu_X \times \mu_Y$ the notions of information cost and external information cost coincide. Thus results about external information cost translate into results about information cost for product distributions.

In addition to the compression questions, another set of unanswered questions stems from trying to associate an operational meaning to $\mathsf{IC}^{ext}_{\mu}(F,\varepsilon)$. We do not have a general characterization similar to Theorem 1, and it is not clear that one exists. However, we can offer the following tantalizing conjecture. Recall that the characterization of information complexity as the amortized communication complexity of F only worked for non-zero error $\varepsilon > 0$. We conjecture that external information complexity gives amortized communication complexity in the zero-error regime:

Problem 6 *Is the following conjecture true:*

$$\lim_{n\to\infty} \mathsf{CC}_{\mu^n}(F^n,0)/n = \mathsf{IC}^{ext}_{\mu}(F,0)? \qquad (13)$$

We can prove the '$\leq$' direction of the conjecture [BGPW12b] using a combination of techniques from [BR11b] and from [HJMR07]. However, we currently do not have a proof for the converse direction, and cannot be sure that it is in fact true. The evidence for it is mostly numerical in the form of "coincidences". We list some of these below.

The message transmission case. In the more familiar setting where Alice wants to transmit a message to Bob, i.e. when $F(X,Y) = X$, the amortized cost of transmission with negligible but non-zero error is $H(X|Y)$ by the Slepian-Wolf theorem [SW73]. It is not hard to see that

$$\lim_{\varepsilon\to 0} \mathsf{IC}_{\mu}(F,\varepsilon) = H(X|Y),$$

which is consistent with Theorem 1. On the other hand, zero-error transmission requires $H(X)$ communication [Orl90, Orl91] – which is the external information complexity of the transmission problem.

The equality function. Suppose the two parties are given two binary strings of length n and want to determine whether they are equal or not. It turns out that one can view this question as n copies of the bit equality $EQ : \{0,1\}\times\{0,1\} \to \{0,1\}$ function where the prior μ is very skewed towards the bits being equal (i.e. μ places almost all of its weight on the $(0,0)$ and $(1,1)$ entries). For full reasoning on this connection see [BGPW12a]. It can be shown that the internal zero-error information complexity of EQ with respect to such distribution is vanishing, since even if the parties just send each other their inputs, the amount of information learned will be negligible, as the parties already know each other's inputs with high degree of confidence. At the same time, the external information complexity of such equality is close to 1 bit in the worst case. This distinction corresponds to the fact that the number of bits two parties need to exchange to determine whether their n-bit strings are equal except with tiny error is $o(n)$ (through hashing), while the number of bits that need to be exchanged to determine equality with *zero* error is $\geq n+1 = 1\cdot n + o(n)$ [KN97].

The set intersection problem. The third example to consider is that of computing the intersection of two subsets $X,Y \subset \{1,2,\ldots,n\}$. This problem can be thought of as computing n two-bit AND functions in parallel. The information complexity of AND is ≈ 1.4922 bits [BGPW12a][3]. Thus the communication complexity of set intersection with small but non-zero error is $\approx 1.4922n + o(n)$. The *external* information complexity of the two-bit AND function is $\log_2 3 \approx 1.5850$. This is consistent with the communication complexity of set intersection with *zero error* being $(\log_2 3)n$ [AC94] – providing further evidence for conjecture (13).

2.4. Beyond two terminals

It is a natural and very interesting goal to generalize the discussion above to more than two terminals. There are various models for multi-terminal interactive computation. The main complication comes from the fact that the prior distributions may be rather sophisticated. One popular model of multi-party computation is that of number-on-forehead (NOF). In the NOF model each party gets to see all inputs but its own and the goal is to compute a function $F(X_1,\ldots,X_k)$ of the inputs [KN97, CP10]. Lower bounds in this model would have profound implications in complexity theory [BT91].

There are numerous complications in extending notions of information complexity to multi-terminal settings. Apart from sheer technical difficulties, a major obstacle is finding the "right" analogue of public and private randomness. Note that even with three parties we have seven different types of randomness (one "private" for each party, one "public", and three shared between two of the three parties but not the third). Allowing all the different types of randomness leads to another impasse, as in this regime there are information-theoretically secure protocols for multi-party computation [BGKW88] which would bring the information complexity of all problems close to 0.

It is possible that answering Problem 6 would be a useful first step toward the multi-terminal case.

[3] The AND function, analyzed in [BGPW12a] is, to the best of our knowledge, the first example where it can be shown that information complexity is an infimum over an infinite sequence of protocols: no finite round protocol quite achieves $\mathsf{IC}_{\mu}(AND,0)$ for almost all μ.

3. Interactive error correction

The discussion so far focused on coding for interactive computing in a noiseless binary channel. In the next two sections we will focus on error-correction problems when the channel contains random or adversarial noise. The first regime we would like to consider is that of adversarial noise. In this regime Alice and Bob are trying to perform a task T over a channel in which an adversary is allowed to corrupt a constant fraction of the messages. Both the regime of a binary channel and that of a channel with constant-size alphabet Σ are interesting.

If the task T is just a simple transmission task, then the theory of (worst-case) error-correcting codes [MS77, Sud01] applies. While there are many open problems in coding theory, the overall picture is fairly well understood. In particular constructions of "good" (i.e. positive-rate, constant-distance) codes exist, and there are efficient encoding and decoding constructions. In the interactive case, the task may include many back-and-forth messages. As a generic task, it is convenient to think about alternating binary pointer jumping (BPJ_d). In this problem the parties are looking at a depth-d binary tree. Alice is given a subset T_A of edges on the odd layers of the tree, such that exactly one edge coming out of each vertex on odd layers. Similarly, Bob is given a subset T_B of edges on the even layers of the tree. Their goal is to find the unique leaf that is connected to the root by edges from $T_A \cup T_B$. There is an obvious d-bit protocol for finding the leaf, where Alice and Bob alternate. In a sense BPJ_d is the generic interactive task, as any interactive protocol can be recast as an instance of BPJ_d.

Now suppose an adversary is allowed to corrupt a δ-fraction of the symbols exchanged by Alice and Bob, for some $\delta > 0$. Can they still compute BPJ_d? One obvious solution is to use ordinary error-correcting codes, using which Alice can send her input T_A to Bob, who can then compute the leaf. This solution would work, but would cause an exponential blow-up in communication, since T_A takes $\sim 2^d$ bits to describe. It is not at all clear that a constant-rate error correcting code is possible. Note that no round-by-round solution could possibly succeed since the adversary is free to use her budget to corrupt some of the early rounds and thus to completely derail the computation. Surprisingly, constant-rate error-correcting codes for interactive computing do exist. The first such code was demonstrated in a breakthrough work by Schulman [Sch96], who showed a constant-rate code against an adversary who is allowed to corrupt a constant δ-fraction of the symbols on the channel. Schuman introduced a concept of a *tree code* that has been used in all constructions since. Schulman's parameters are far from optimal. In particular δ is limited to be below $1/240$. In addition, the construction is not efficient in that it requires time exponential in d to compute the encoding/decoding (even though the communication itself is $O(d)$ symbols).

A recent line of work on interactive error-correction [BR11a, GMS11, Bra12a, BK12] improves on Schulman's original work in several directions. [BR11a] improves error tolerance to any $\delta < 1/4$ for constant symbol space size $|\Sigma|$, and $\delta < 1/8$ for binary channels $\Sigma = \{0,1\}$. This error-tolerance is arguably optimal, at least for a large class of protocols. Note that an error rate of $\delta = 1/4$ means that the adversary may corrupt $1/2$ or Alice's messages – as long as she speaks at most half of the time. Thus unique decoding is not possible for protocols where the order of messages is fixed in advance, or more generally for encoded protocols where both parties speak equally often. There is still a theoretical possibility that the $1/4$ barrier can be broken if the parties can adaptively allocate the turn to the party whose messages are being corrupted more often. Other works aim at making the constructions more computationally efficient. In particular, [BK12] succeeds at making the construction poly-time efficient, while tolerating error rates of up to $\delta < 1/16$ ($\delta < 1/32$ over the binary alphabet). Finding an efficient construction with the best error rate remains open:

Problem 7 *Give an polynomial-time efficient error-correcting scheme for interactive error correction with error rate of up to $\delta < 1/4$ ($\delta < 1/8$ over a binary channel).*

None of the works so far have considered the question of rate seriously – beyond ensuring that it is non-zero. This is mainly due to insufficient tools to tackle this question. Hence the question of rate remains wide open:

Problem 8 *Given an error rate $\delta > 0$ what is the best rate $\rho(\delta)$ which we can achieve while protecting interactive communication against a δ-fraction of adversarial errors?*

Obvious upper bounds are the corresponding values for the non-interactive case, but understanding the additional "interactivity penalty" one incurs while trying to protect interactive computation may require the development of new tools.

Finally, we turn our attention to list-decodable codes [Sud00]. It is not immediately clear how to extend interactive error correction to the list-decodable scenario. In the context of the BPJ_d problem, by the end of the computation, the parties should have a small

(constant) number of leaves which are consistent with the messages received by the parties. There are many questions one can ask about list-decodable interactive coding. In particular, we conclude with the following problem:

Problem 9 *Is there a list-decodable encoding for the BPJ_d problem that for all $\delta < 1/2$, assuming the adversary corrupts at most a δ-fraction of the symbols in the transmissions, outputs a constant number of leaves one of which is the correct leaf? If the answer is 'no', is such list decoding possible for any value $\delta \geq 1/4$?*

4. Interactive channel capacity

Finally, we turn our attention to questions of interactive channel capacity. This can be viewed as the random (rather than adversarial) analogue of the coding problems discussed in previous section. To the best of our knowledge, not much is known in this regime.

A natural definition of interactive channel capacity is in terms of *the ability of the channel to sustain interactive communication, and the rate at which this communication can be sustained.* For simplicity, let us focus on a binary channel. The ability to sustain interactive computation can be measured in terms of the ability to solve the binary pointer jumping (*BPJ*) problem that has been discussed above. Thus the capacity of the channel is c if it takes d/c one-bit messages over the channel to solve the depth-d BPJ_d problem with high probability.

Definition 10 *The interactive channel capacity c of a channel $\mathscr{C}$ is the smallest ratio such that for all $c' < c$ and for all d large enough, BPJ_d can be solved using d/c' transmissions over the channel, with a vanishing error probability.*

Definition 10 gives rise to a large number of questions about the capacities of individual channels. Clearly that standard Shannon's channel capacity is an upper bound on the interactive channel capacity. On the other hand, results about interactive error correction that were discussed in the previous section imply that channels that have non-zero capacity will also have non-zero interactive capacity.

Among the possible channels, we highlight two interesting special cases. The first one is that of binary erasure channels.

Problem 11 *What is the interactive capacity c_{p_e} of a binary erasure channel with erasure probability p_e?*

The second one is for a binary symmetric channel:

Problem 12 *What is the interactive capacity c_{p_s} of a binary symmetric channel with crossover probability p_s?*

In particular, understanding how c_{p_e} and c_{p_s} behave as $p_e \to 0$ and $p_s \to 0$ are interesting questions.

5. Conclusion

In this note we have discussed some basic notions, recent progress, and open problems surrounding interactive coding theory. Far from being a broad historical or bibliographic survey, the brief note focuses on recent developments and directions with the hope of further engaging the information theory community in interactive information theory.

Acknowledgments

I would like to thank Ankit Garg, Sreeram Kannan, Denis Pankratov, Anup Rao, Pramod Viswanath, and Omri Weinstein for enlightening discussions. I would like to thank Omri Weinstein for his comments on earlier drafts of this paper.

References

[AC94] R. Ahlswede and N. Cai. On communication complexity of vector-valued functions. *Information Theory, IEEE Transactions on*, 40(6):2062–2067, 1994.

[BBCR10] Boaz Barak, Mark Braverman, Xi Chen, and Anup Rao. How to compress interactive communication. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 67–76, New York, NY, USA, 2010. ACM.

[BGKW88] Michael Ben-Or, Shafi Goldwasser, Joe Kilian, and Avi Wigderson. Multi-prover interactive proofs: How to remove intractability assumptions. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing*, pages 113–131, 1988.

[BGPW12a] Mark Braverman, Ankit Garg, Denis Pankratov, and Omri Weinstein. Information and exact communication complexity. *forthcoming*, 2012.

[BGPW12b] Mark Braverman, Ankit Garg, Denis Pankratov, and Omri Weinstein. On the external information complexity. *in progress*, 2012.

[BK12] Z. Brakerski and Y.T. Kalai. Efficient interactive coding against adversarial noise. In *Electronic Colloquium on Computational Complexity (ECCC)*, 2012.

[BR11a] M. Braverman and A. Rao. Towards coding for maximum errors in interactive communication.

In *Proceedings of the 43rd annual ACM symposium on Theory of computing*, pages 159–166. ACM, 2011.

[BR11b] Mark Braverman and Anup Rao. Information equals amortized communication. In *FOCS*, pages 748–757, 2011.

[Bra12a] M. Braverman. Towards deterministic tree code constructions. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 161–167. ACM, 2012.

[Bra12b] Mark Braverman. Interactive information complexity. In *Proceedings of the 44th symposium on Theory of Computing*, STOC '12, pages 505–524, New York, NY, USA, 2012. ACM.

[BT91] R. Beigel and J. Tarui. On acc [circuit complexity]. In *Foundations of Computer Science, 1991. Proceedings., 32nd Annual Symposium on*, pages 783–792. IEEE, 1991.

[BYJKS04] Ziv Bar-Yossef, T. S. Jayram, Ravi 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.

[CP10] A. Chattopadhyay and T. Pitassi. The story of set disjointness. *ACM SIGACT News*, 41(3):59–85, 2010.

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

[GMS11] R. Gelles, A. Moitra, and A. Sahai. Efficient and explicit coding for interactive communication. In *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*, pages 768–777. IEEE, 2011.

[HJMR07] Prahladh Harsha, Rahul Jain, David A. McAllester, and Jaikumar Radhakrishnan. The communication complexity of correlation. In *IEEE Conference on Computational Complexity*, pages 10–23. IEEE Computer Society, 2007.

[Kla04] Hartmut Klauck. Quantum and approximate privacy. *Theory Comput. Syst.*, 37(1):221–246, 2004.

[KN97] Eyal Kushilevitz and Noam Nisan. *Communication complexity*. Cambridge University Press, Cambridge, 1997.

[MI11] N. Ma and P. Ishwar. Some results on distributed source coding for interactive function computation. *Information Theory, IEEE Transactions on*, 57(9):6180–6195, 2011.

[MIG12] N. Ma, P. Ishwar, and P. Gupta. Interactive source coding for function computation in collocated networks. *Information Theory, IEEE Transactions on*, 58(7):4289–4305, 2012.

[MS77] F. J. MacWilliams and N. J. A. Sloane. *The theory of error correcting codes*. North–Holland, New York, 1977.

[New91] Ilan Newman. Private vs. common random bits in communication complexity. *Information Processing Letters*, 39(2):67–71, 31 July 1991.

[Orl90] A. Orlitsky. Worst-case interactive communication. i. two messages are almost optimal. *Information Theory, IEEE Transactions on*, 36(5):1111–1126, 1990.

[Orl91] A. Orlitsky. Worst-case interactive communication. ii. two messages are not optimal. *Information Theory, IEEE Transactions on*, 37(4):995–1005, 1991.

[Sch96] Leonard J. Schulman. Coding for interactive communication. *IEEE Transactions on Information Theory*, 42(6):1745–1756, 1996.

[Sud00] M. Sudan. List decoding: Algorithms and applications. *Theoretical Computer Science: Exploring New Frontiers of Theoretical Informatics*, pages 25–41, 2000.

[Sud01] M. Sudan. Algorithmic introduction to coding theory – course notes, 2001. http://people.csail.mit.edu/madhu/FT01/course.html.

[SW73] David Slepian and Jack K. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, 19(4):471–480, July 1973.