Direct product via round-preserving compression

Mark Braverman[⋆1] , Anup Rao[⋆⋆2], Omri Weinstein[⋆⋆⋆1], and Amir Yehudayoff[†3]

[1] Princeton University
[2] University of Washington
[3] Technion IIT

Abstract. We obtain a strong direct product theorem for two-party bounded round communication complexity. Let $\mathsf{suc}_r(\mu, f, C)$ denote the maximum success probability of an r-round communication protocol that uses at most C bits of communication in computing $f(x, y)$ when $(x, y) \sim \mu$. Jain et al. [12] have recently showed that if $\mathsf{suc}_r(\mu, f, C) \leq \frac{2}{3}$ and $T \ll (C - \Omega(r^2)) \cdot \frac{n}{r}$, then $\mathsf{suc}_r(\mu^n, f^n, T) \leq \exp(-\Omega(n/r^2))$. Here we prove that if $\mathsf{suc}_{7r}(\mu, f, C) \leq \frac{2}{3}$ and $T \ll (C - \Omega(r \log r)) \cdot n$ then $\mathsf{suc}_r(\mu^n, f^n, T) \leq \exp(-\Omega(n))$. Up to a $\log r$ factor, our result asymptotically matches the upper bound on $\mathsf{suc}_{7r}(\mu^n, f^n, T)$ given by the trivial solution which applies the per-copy optimal protocol independently to each coordinate. The proof relies on a compression scheme that improves the tradeoff between the number of rounds and the communication complexity over known compression schemes.

1 Introduction

We study the *direct sum* and the *direct product* problem for bounded-round randomized communication complexity. The direct sum problem studies the amount of resources needed to solve n independent copies of a task in terms of the cost of solving one copy. It is the case that if one copy costs C resources, then n copies can be solved using $C_n \leq n \cdot C$ resources. Can one do better? Direct sum theorems answer this question by giving lower bounds for C_n in terms of C and n — aiming to give a tight $\Omega(n \cdot C)$ bound whenever possible. If the task is solved in a randomized model, with some error allowed, the performance of a solution for a

⋆ Department of Computer Science, Princeton University, Research supported in part by an Alfred P. Sloan Fellowship, an NSF CAREER award (CCF-1149888), and a Turing Centenary Fellowship.

⋆⋆ Computer Science and Engineering, University of Washington, Supported by the National Science Foundation under agreement CCF-1016565, an NSF Career award and by the Binational Science Foundation under agreement 2010089. Part of this work was done while the author was visiting Princeton University and the Technion.

⋆⋆⋆ Department of Computer Science, Princeton University,

† Department of Mathematics, Technion-IIT, Haifa, Israel, Horev Fellow – supported by the Taub Foundation. Supported by the Israel Science Foundation and by the Binational Science Foundation under agreement 2010089.

single copy of the task is characterized by its cost C and its success probability ρ. Clearly, with $n \cdot C$ resources a success probability of at least ρ^n is attainable, but is it optimal? A direct product theorem is stronger than a direct sum theorem in that in addition to asserting that a certain amount of resources is necessary to compute the n copies, it also shows that using a smaller amount of resources will lead to a very low (possibly exponentially small) success probability.

Direct product theorems have a long history in complexity theory, and in communication complexity in particular [19,16,21,11,12,6]. See [12] for a discussion of the various direct product theorems. In the context of communication complexity, direct product results for specific lower-bound techniques were given by a number of papers: for discrepancy in the two party case by Shaltiel [21] and Lee, Shraibman and Spalek [17], by Sherstov for generalized discrepancy [22], and by Viola and Wigderson for the multiparty case [23]. More recently, a direct product theorem was given by Jain and Yao in terms of the smooth rectangle bound [13]. Direct product results for specific communication problems such as set disjointness include [15,2]. Famous examples for direct product theorems for other models of computation include Yao's XOR lemma and Raz's parallel repetition theorem [20]. For (unbounded-round) communication complexity, the current state-of-the-art results are given by [6], which shows that n copies of a function cost $\Omega(\sqrt{n})$ times the cost of one copy, and any computation using less communication will fail except with an exponentially small probability. [13], building on [14], obtains a strong direct product theorem in terms of the smooth rectangle bound – showing that a strong direct product theorem holds for the communication complexity of a large number of commonly studied functions.

In this paper we focus on the *bounded-round*, distributional, two party communication complexity model. Bounded-round communication complexity is used extensively in streaming and sketching lower bounds (see e.g. [9,18] and references therein). We prove a tight direct sum and direct product theorem for this model. The two players are given inputs according to a distribution $(x, y) \sim \mu$ and need to compute a function $f(x, y)$. The players perform the computation using a communication protocol π. In the bounded-round model, the players are allowed a total of at most r messages in their protocol π. The *communication cost* $\|\pi\|$ of a protocol π is the (worst-case) number of bits the players send when running π. If π has r rounds then $\|\pi\| \geq r$. The *success probability* of π, denoted $\mathsf{suc}(\mu, f, \pi)$, is the probability it outputs the correct value of f (for a formal definition see Section 3.3). The probability that *any* r-round protocol of communication cost C succeeds at computing f is denoted by

$$\mathsf{suc}_r(\mu, f, C) := \max_{\pi \text{ is } r\text{-round and } \|\pi\| \leq C} \mathsf{suc}(\mu, f, \pi).$$

The *unbounded round* success probability $\mathsf{suc}(\mu, f, C)$ is defined as $\mathsf{suc}_C(\mu, f, C)$ (the trivial bound of C does not limit interaction, as $r \leq C$ by definition).

The function $f^n((x_1, \ldots, x_n), (y_1, \ldots, y_n))$ is just the concatenation of n copies of f. In other words, it outputs $(f(x_1, y_1), \ldots, f(x_n, y_n))$. Assume that $\mathsf{suc}_r(\mu, f, C) < 2/3$. Both the direct sum and the direct product question ask what can be said about the cost, and the success probability of solving f^n. A

strong *direct sum* theorem for bounded-round computation would assert that $\mathsf{suc}_{\alpha r}(\mu^n, f^n, \alpha n \cdot C) < 3/4$, for some constant $\alpha > 0$. A *direct product* theorem would further assert that $\mathsf{suc}_{\alpha r}(\mu^n, f^n, \alpha n \cdot C) < (2/3)^{\alpha n}$. Clearly, the latter statement is the best one can hope for up to constants, since trivially $\mathsf{suc}_r(\mu^n, f^n, n \cdot C) \geq \mathsf{suc}_r(\mu, f, C)^n$.

Prior to the present work, several general direct sum and direct product results for bounded-round communication complexity were given. The work [10] by Harsha, Jain, McAllester and Radhakrishnan gives a strong direct sum result for bounded-round communication, but it only works for product distributions (i.e. when μ is of the form $\mu = \mu_x \times \mu_y$). The paper [5] by Braverman and Rao gives a direct sum result for bounded-round communication of the following form: if $\mathsf{suc}(\mu, f, C) < 2/3$, then $\mathsf{suc}_r(\mu^n, f^n, n \cdot C \cdot (1 - o(1))) < 3/4$, for n sufficiently large. This result gives a tight dependence on the communication complexity, but assumes a lower bound on the communication complexity of a single copy of f *without restriction on the number of rounds*. Therefore, strictly speaking, it is not a direct sum result for *bounded-round* communication complexity. The only general *direct product* result for bounded-round communication complexity was recently given by a Jain, Pereszlenyi, and Yao [12], who showed that if $\mathsf{suc}_r(\mu, f, C) \leq \frac{2}{3}$ and $T \ll (C - \Omega(r^2)) \cdot \frac{n}{r}$, then $\mathsf{suc}_r(\mu^n, f^n, T) \leq \exp(-\Omega(n/r^2))$. This result is indeed a proper direct product theorem for bounded-round communication. Its parameters are sub-optimal in two respects: (1) there is no reason for the direct product theorem to not hold all the way to $T = \Omega(C \cdot n)$, and (2) in a tight direct product theorem the success probability $\mathsf{suc}_r(\mu^n, f^n, T)$ would be $\exp(-\Omega(n)) \ll \exp(-\Omega(n/r^2))$.

Our results. Our main result is an optimal (up to constants and a $\log r$ factor) direct product theorem for bounded-round communication complexity (see Theorem 2 below). The theorem improves over the parameters in [12], with the exception of the dependence on the number of rounds: we require a lower bound for protocols using $7r$ rounds of communication for one copy to get a lower bound for an r-round protocol for n copies. Using Yao's minimax principle [24], our result also applies to the randomized bounded-round communication complexity.

Our techniques. Our general strategy is similar to other recent direct sum and direct product results [10,1,12,6]. The first main ingredient is the notion of *information cost* of protocols. The information cost of a two-party protocol π over a distribution μ of inputs $(x, y) \sim \mu$ is defined as the amount of information the parties learn about each other's inputs from the messages of the protocol. More formally, if we define X, Y to be the random variables representing the inputs, and M to be the random variable representing the messages or transcript, then the information cost of π with respect to μ is given by

$$IC(\pi, \mu) := I(X; M|Y) + I(Y; M|X),$$

where $I(A; B|C)$ is the mutual information between A and B conditioned on C.

In general, direct sum and direct product proofs proceed in two steps: As a first step, it is shown that if f^n can be solved using fewer than T resources, then one copy of f can be solved using a protocol π, that while having high communication complexity (T), has low information complexity: $IC(\pi, \mu) = O(T/n)$.[4] The second step is to convert the protocol π into a protocol π' that has low communication cost, such as $O(IC(\pi, \mu)))$. This is done through *protocol compression*: the process of converting a low-information interactive protocol into a low communication protocol. If successful, this step leads to a low-communication protocol for one copy of f, which contradicts the initial lower bound assumption on one copy of f.

The process of obtaining new direct sum results in communication complexity has been tightly linked to the process of obtaining new protocol compression results. In fact, the question of whether the general (unbounded-round) direct sum for communication complexity holds is equivalent to the question of whether all protocols can be compressed [5,4]. In the case of bounded-round protocols the problem of compressing protocols reduces to the problem of compressing individual messages in the protocol. The problem of message compression can be rephrased as follows: player 1 has a distribution P of the message $M \sim P$ he wants to send to player 2. Player 2 has some prior belief Q about the distribution of M. How much communication is needed to ensure that both players jointly sample $M \sim P$? The natural information-theoretic lower bound for this problem is the KL-divergence $\mathsf{D}\left(\frac{P}{Q}\right)$. More specifically, if the element being sampled is a, we should expect player 1 to communicate at least $\log(P(a)/Q(a))$ bits to player 2.

If we start off with the assumption that it is hard to solve one copy of f using a *bounded-round* protocol, then to obtain a contradiction our compression scheme should preserve (or at least not blow-up) the number of rounds in the protocol. This means, ideally, that compression of one round should take only a constant number of rounds. The round-compression scheme of [5], in fact, manages to attain near-optimal compression in terms of communication cost. The communication cost of the problem described above is reduced to $\mathsf{D}\left(\frac{P}{Q}\right) \cdot (1 + o(1)) + O(\log 1/\epsilon)$, where ϵ is an error parameter. There is a price to be paid for such communication performance: there is no good bound on the number of rounds such compression would take. Thus the resulting compressed protocol is no longer bounded-round. Therefore, [5] only obtains a lower bound on the bounded-round communication complexity of f^n in terms of *the unbounded-round communication complexity of* f.

The recent works [11,12] devise a different compression scheme that does not increase the number of rounds at all: each message in the original protocol is compressed into one message in the compressed protocol. As a result, these works obtain direct product theorems for bounded-round communication complexity.

[4] In the case of direct product, what is shown is that π is statistically close to being a low information protocol.

These compressions, however, end up paying a high price in the communication overhead. Specifically, due to an application of Markov inequality, sending a message a, on average, takes $r \cdot \log(P(a)/Q(a))$ bits – a multiplicative loss by an r factor, which leads to a factor-r loss in the ultimate result.

Our main technical contribution is a new family of compression protocols for compressing one round of communication. These protocols are parameterized by two parameters (d, ℓ). They give a tradeoff between the communication overhead and the resulting number of rounds. Specifically:

Theorem 1. *For any $a, \ell > 0$, let $\log_\ell^+(a) = \max\{0, \log_\ell(a)\}$. Suppose that player 1 is given a distribution P (unknown to Player 2), and player 2 is given a distribution Q, both over a universe $\mathcal{U}$. Then, for every $0 < \epsilon < 1/2$, $d \geq 1$ and integer $\ell \geq 2$, there is a protocol such that at the end of the protocol:*

- *player 1 outputs an element a distributed according to P.*
- *player 2 outputs an element b s.t for each $x \in \mathcal{U}$, $\Pr[b = a | a = x] > 1 - \epsilon$.*
- *the communication is at most $(2\ell+1) \cdot \log_2^+(P(a)/Q(a)) + 2\log(1/\epsilon) + 2d + 5$.*
- *the number of rounds is at most $2\log_\ell^+[(1/d)\log_2^+(P(a)/Q(a))] + 2$.*

The second condition implies in particular that player 2 outputs an element b such that $b = a$ with probability at least $1 - \epsilon$. The protocol requires no prior knowledge or assumptions on P, Q.

One can see that setting d and ℓ to be large in Theorem 1 will result in few rounds but long communication, and vice versa. The compression scheme in Theorem 1 may be of independent interest. It is possible to view both compression schemes from [5] and from [11,12] as special cases of Theorem 1. The scheme in [5] approximately corresponds to $(d, \ell) = (2, 1)$. The scheme in [11,12] corresponds to $d = \Theta(IC(\pi, \mu))$. By carefully choosing the parameters in Theorem 1, and analyzing the resulting number of rounds and communication cost over all rounds simultaneously, we obtain a compression scheme that at the same time increases the communication cost and the number of rounds of communication by only a constant. This scheme, together with direct product reductions from [6], allows us to complete the proof of Theorem 2.

Discussion and open problems. Our work essentially closes the direct product question in the regime where the number of rounds r is small compared to C, and $\mathsf{suc}_r(\mu, f, C)$ is constant in $(0, 1)$. The general direct product problem (and even the weaker direct sum problem) remains wide open. The key compression challenge one needs to overcome is the problem of compressing protocols when $r \gg I$, that is, when the amount of information π conveys in a typical round is $o(1)$. Further discussion on this problem can be found in [4,3].

An important area of tradeoff – both in terms of direct sum/product results and in terms of compression is the relationship between error, communication complexity, and the number of rounds. When performing compression to a bounded number of rounds r, we inevitably have to abort the protocol if the rounds "quota" is exceeded. What is the effect this has on error incurred? A

very recent work by Brody, Chakrabarti, and Kondapally [8] suggests the general tradeoff may take an interesting form. Understanding these tradeoffs is crucial for getting tight parameters for bounded-round direct sum and product in the regime where $\mathsf{suc}_r(\mu, f, C)$ is very close to 1.

2 Results

Let $\mathsf{suc}_r(\mu, f, C)$ denote the maximum success probability of an r-round communication protocol that uses at most C bits of communication to compute $f(x, y)$ when $x, y \sim \mu$. Denote by $f^n(x_1, \ldots, x_n, y_1, \ldots, y_n)$ the function that maps its inputs to the tuple $(f(x_1, y_1), f(x_2, y_2), \ldots, f(x_n, y_n))$ and μ^n denote the product distribution on n pairs of inputs, where each pair is sampled independently according to μ. We prove the following direct product result.

Theorem 2 (Main Theorem). *Let f be a 2-party Boolean function. There is a universal constant $\alpha > 0$ such that if $\gamma = 1 - \mathsf{suc}_{7r}(\mu, f, C)$, $T \geq 2$, and $T < \alpha n \gamma^2 \left(C - \frac{r \log(r/2\gamma)}{\alpha\gamma} - \frac{r}{\alpha\gamma^2} \right)$, then $\mathsf{suc}_r(\mu^n, f^n, T) \leq \exp\left(-\alpha\gamma^2 n\right)$.*

When $\mathsf{suc}_r(\mu, f, C) \leq \frac{2}{3}$ and $r \log r \ll C$, Theorem 2 ensures that the success probability of any protocol attempting to compute f^n under μ^n using $\ll Cn$ communication and $r/7$ rounds must be exponentially small in n.

Our main technical contribution is showing how to compress bounded-round protocols without introducing (too many) additional rounds.

The first step is the sampling protocol described in Theorem 1, which shows how to jointly and efficiently sample from a desired distribution in an oblivious manner. Suppose player 1 knows a distribution P, player 2 knows a distribution Q, and the players wish to jointly sample from P without knowing the distribution of the other player. It is an extension of a protocol from [5]. The protocol is interactive and the requires multiple rounds. The number of rounds required for the simulation in [5] is $\Theta(\sqrt{\Delta})$, where Δ is the KL divergence between the distributions P and Q. While this suffices for the particular objective in [5], this is more than we can afford here: the compression scheme implies that an r-round protocol which reveals I bits of information can be simulated by an $O(r\sqrt{I})$-round protocol that has $I + o(I)$ communication. The resulting compressed protocol is no longer bounded-round, requiring us to assume a stronger lower bound on the hardness of one copy of f to reach a contradiction. Our new compression protocol ensures that at most $7r$ rounds of communication are used with high probability, which means that assuming that f cannot be efficiently solved by a $7r$-round protocol suffices.

The second step in the proof is showing how to use the single-message sampling protocol from Theorem 1 to simulate communication protocols, with communication comparable to the amount of information they convey, while keeping the number of rounds comparable to the original number. In fact, to prove our main result, we actually need to analyze protocols that are merely close to having low information cost. As noticed in [6], such protocols need not have low

information themselves. E.g., consider the protocol π in which player 1 sends her n-bit uniformly random input x with probability ϵ, and otherwise sends a random string. Then π is ϵ-close to a 0-information protocol, but $IC(\pi) = \epsilon n$. Nevertheless, *truncation* of protocols (as in [6]) implies that compression is possible even in this more general setting. This is formalized by the next theorem.

Theorem 3 (Round preserving compression). *Suppose θ is an r-round protocol with inputs x, y and messages m, and q is another distribution on these variables such that $\theta(xym) \overset{\epsilon}{\approx} q(xym)$. Let $I = I_q(X; M|Y) + I_q(Y; M|X)$. Then there exists a $7r$-round protocol τ that 11ϵ-simulates θ such that*

$$\|\tau\| \leq 7\frac{I}{\epsilon^2} + 2\frac{r\log(r/\epsilon)}{\epsilon} + 30\frac{r}{\epsilon^2}.$$

The compression protocol in Theorem 3 is obtained by sequential applications of Theorem 1. However, in order to prevent a blowup in the number of simulating rounds, we cannot use the guarantees of Theorem 1 on a per-round basis. We analyze the protocol in a global manner, which yields the desirable tradeoff between the number of rounds and the communication complexity.

3 Preliminaries

3.1 Notation

Unless otherwise stated, logarithms in this text are computed in base two. Random variables are denoted by capital letters and values they attain are denoted by lower-case letters. For example, A may be a random variable and then a denotes a value A may attain and we may consider the event $A = a$. Given $a = a_1, a_2, \ldots, a_n$, we write $a_{\leq i}$ to denote $a_1, \ldots, a_i$. We define $a_{>i}$ and $a_{\leq i}$ similarly. For an event E, define $\mathbf{1}_E$ to be the indicator random variable of E.

We use the notation $p(a)$ to denote both the distribution on the variable a, and the number $\Pr_p[A = a]$. The meaning will typically be clear from context, but in cases where there may be confusion we shall be more explicit about which meaning is being used. We write $p(a|b)$ to denote either the distribution of A conditioned on the event $B = b$, or the number $\Pr[A = a|B = b]$. For an event W, we write $p(W)$ to denote the probability of W according to p. We let $\mathbb{E}_{p(a)}[g(a)]$ denote the expected value of $g(a)$ when a is distributed according to p.

For two distributions p, q, we write $|p(a) - q(a)|$ to denote the ℓ_1 distance between the distributions p and q. We write $p \overset{\epsilon}{\approx} q$ if $|p - q| \leq \epsilon$.

The *divergence* between p, q is defined to be

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

For three random variables A, B, C jointly distributed according to $p(a, b, c)$, the *mutual information* between A, B conditioned on C is defined as

$$I_p(A; B|C) = \mathop{\mathbb{E}}_{p(cb)}\left[\mathsf{D}\left(\frac{p(a|bc)}{p(a|c)}\right)\right] = \mathop{\mathbb{E}}_{p(ca)}\left[\mathsf{D}\left(\frac{p(b|ac)}{p(b|c)}\right)\right] = \sum_{a,b,c} p(abc)\log\frac{p(a|bc)}{p(a|c)}.$$

3.2 Properties of divergence

Lemma 1 (Chain Rule). *If $a = a_1, \ldots, a_s$, then*

$$\mathsf{D}\left(\frac{p(a)}{q(a)}\right) = \sum_{i=1}^{s} \mathop{\mathbb{E}}_{p(a_{<i})}\left[\mathsf{D}\left(\frac{p(a_i|a_{<i})}{q(a_i|a_{<i})}\right)\right].$$

The following lemmas describe basic properties of divergence (for proofs see [6]).

Lemma 2. *Let $S = \{a : p(a) < q(a)\}$. Then, $\sum_{a\in S} p(a) \log \frac{p(a)}{q(a)} \geq -1/(e \ln 2)$.*

Lemma 3 (Truncation Lemma [6]). *Let $p(a,b,c) \overset{\epsilon}{\approx} q(a,b,c)$ where $a = a_1, \ldots, a_s$. For every a, b, c, define k to be the minimum number j in $[s]$ such that*

$$\log \frac{p(a_{\leq j}|bc)}{p(a_{\leq j}|c)} > \beta.$$

If no such index exists, set $k = s + 1$. Then,

$$p(k < s+1) < \frac{I_q(A;B|C) + \log(s+1) + 1/(e\ln 2)}{\beta - 2} + 9\epsilon/2.$$

3.3 Communication complexity

Given a protocol π that operates on inputs X, Y drawn from a distribution μ and (possibly) using public randomness S and messages M, we write $\pi(xyms)$ to denote the joint distribution of these variables. We write $\|\pi\|$ to denote the *communication complexity* of π, namely the maximum number of bits that may be exchanged by the protocol.

A central measure in this paper is the information complexity of a communication protocol (see [1,4] and references within for a more detailed overview). The *internal information cost* of π is defined to be $IC(\pi) := I_\pi(X; M|YS) + I_\pi(Y; M|XS)$. It is well known (e.g, [4]) that for any protocol π, $IC(\pi) \leq \|\pi\|$.

Let $q(x, y, a)$ be an arbitrary distribution. We say that π *δ-simulates* q, if there is a function g and a function h such that $\pi(x, y, g(x, s, m), h(y, s, m)) \overset{\delta}{\approx} q(x, y, a, a)$, where $q(x, y, a, a)$ is the distribution on 4-tuples (x, y, a, a) where (x, y, a) are distributed according to q. Thus if π δ-simulates q, the protocol allows the parties to sample a according to $q(a|xy)$. If in addition $g(x, s, m)$ does not depend on x, we say that π *strongly* δ-simulates q. Thus if π strongly simulates q, then the outcome of the simulation is apparent even to an observer that does not know x or y.

If λ is a protocol with inputs x, y, public randomness s' and messages m', we say that π δ-simulates λ if π δ-simulates $\lambda(x, y, (s', m'))$. Similarly, we say that π strongly δ-simulates λ if π strongly δ-simulates $\lambda(x, y, (s', m'))$. We say that π computes f with success probability $1-\delta$, if π strongly δ-simulates $\pi(x, y, f(x, y))$. We denote this by $\mathsf{suc}(\mu, f, \pi) = 1 - \delta$.

The next proposition is straightforward. A formal proof can be found in the full version of this paper [7].

Proposition 1. *Let $f : \mathcal{X} \times \mathcal{Y} \longrightarrow \mathcal{Z}$ and let π be such that $\mathsf{suc}(\mu, f, \pi) = 1 - \delta$. Then if λ is a protocol that ϵ-simulates π, there is a protocol τ such that $\mathsf{suc}(\mu, f, \tau) \geq 1 - (\delta + \epsilon)$ and $\|\tau\| = \|\lambda\| + \log|\mathcal{Z}|$. The number of rounds in τ is the same as in π.*

4 Proof of Theorem 1

Proof (of Theorem 1). Due to space constraints, here we only present the sampling protocol. A full analysis of the protocol together with the proof of Theorem 1 appears in the full version of this paper [7].

We start by describing the content of the shared random tape. Both parties interpret part of the shared random tape as a sequence of independent uniformly selected elements $\{e_i\}_{i=1}^{\infty} = \{(x_i, p_i)\}_{i=1}^{\infty}$ from the set $\mathcal{E} := \mathcal{U} \times [0, 1]$. There is also a part of the shared random tape that contains random independent hash functions $\{h_i\}_{i=1}^{\infty}$, that is, for every i, the function $h_i : \mathcal{U} \to \{0, 1\}$ is so that $\Pr[h_i(x) = h_i(y)] = 1/2$ for every $x \neq y$ in $\mathcal{U}$.

The players use the following definitions: Define

$$\mathcal{E}_P := \{(x, p) \in \mathcal{E} : P(x) > p\},$$

the set of points under the histogram of P. Similarly, define

$$\mathcal{E}_Q := \{(y, q) \in \mathcal{E} : Q(y) > q\}.$$

For a constant $C \geq 1$, define the C-multiple of $\mathcal{E}_Q$ as

$$C \cdot \mathcal{E}_Q := \{(y, q) \in \mathcal{E} : (y, q/C) \in \mathcal{E}_Q\}.$$

For a non-negative integer t, set

$$C_t := 2^{d\ell^t} \quad \text{and} \quad s_t := 2d\ell^t + \lceil \log(1/\epsilon) \rceil + 1.$$

The protocol. The protocol runs as follows:

1. Player 1 selects the first index i such that $e_i = (x_i, p_i) \in \mathcal{E}_P$, and outputs x_i.
2. Player 1 uses $1 + \lceil \log\log(1/\epsilon) \rceil$ bits to send player 2 the binary encoding of
$$k := \lceil i/|\mathcal{U}| \rceil.$$
If $k > 2^{\log\log(1/\epsilon)}$, player 1 sends the all-zero string and the players abort.
3. Repeat, until player 2 produces an output, starting with $t = 0$:
   (a) Player 1 sends the values of all hash functions $h_j(x_i)$ for $1 \leq j \leq s_t$, that have not been previously sent.
   (b) If there is an $a_r = (y_r, q_r)$ with $r \in \{(k-1) \cdot |\mathcal{U}| + 1, \ldots, k \cdot |\mathcal{U}|\}$ in $C_t \cdot \mathcal{E}_Q$ such that $h_j(y_r) = h_j(x_i)$ for some $1 \leq j \leq s_t$, then player 2 says "success" and outputs y_r (if there is more than one such a_r, player 2 selects the first one).
   (c) Otherwise, player 2 responds "failure" and the parties increment t to $t + 1$ and repeat.

5 Round preserving compression - Proof of Theorem 3

Proof (of Theorem 3). Our simulating protocol for θ is the protocol σ described in Figure 1 (The final protocol τ will be defined as a truncation of σ). Once again, due to space constraints, here we only present the protocol. For a complete analysis and the rest of the proof of Theorem 3, we refer the reader to the full version of this paper [7].

Protocol σ for simulating θ

Player 1 repeatedly computes a message $m' = m'_1, \ldots, m'_r$ and player 2 repeatedly computes a message
$m'' = m''_1, \ldots, m''_r$ as follows.

- For odd j, player 1 sets $P = \theta(m_j | m'_{<j} x)$ and player 2 sets $Q = \theta(m_j | m''_{<j} y)$.
- For even j, player 1 sets $Q = \theta(m_j | m'_{<j} x)$ and player 2 sets $P = \theta(m_j | m''_{<j} y)$.
- In each round j, the players run the protocol from Theorem 1 with error parameter ϵ/r, with $\ell = 2$,
  and with $d = \frac{\beta}{r\epsilon} + \frac{1}{\epsilon}$ where

$$\beta = \frac{I + 1/(e \ln 2) + \log(r+1)}{\epsilon} + 2.$$

  This leaves player 1 with m'_j and player 2 with m''_j.

Fig. 1. A round preserving compression of the protocol θ.

6 Direct product for bounded round protocols

Let π be a (deterministic) r-round protocol for computing f^n with inputs $\overline{x} = x_1, \ldots, x_n$ and $\overline{y} = y_1, \ldots, y_n$ drawn from μ^n. To prove Theorem 2, we follow the approach of [6] which itself resembles the proof of the parallel repetition theorem [20]. Let W be the event that π correctly computes f^n. For $i \in [n]$, let W_i denote the event that the protocol π correctly computes the i'th copy $f(x_i, y_i)$. Let $\pi(W)$ denote the probability of W, and $\pi(W_i|W)$ denote the conditional probability of the event W_i given W (clearly, $\pi(W_i|W) = 1$). We shall prove that if $\pi(W)$ is not very small and $\|\pi\| \ll Cn$, then $(1/n)\sum_{i=1}^{n} \pi(W_i|W) < 1$, which is a contradiction. In fact, the proof holds for an arbitrary event W, as long as it occurs with large enough probability:

Lemma 4 (Main Lemma). *Let f be a 2-party Boolean function. There is a universal constant $\alpha > 0$ so that the following holds. For every $\gamma > 0$, and event W such that $\pi(W) \geq 2^{-\gamma^2 n}$, if $\|\pi\| \geq 2$, and $\|\pi\| < \alpha n \gamma^2 \left(C - \frac{r \log(r/2\gamma)}{\alpha\gamma} - \frac{r}{\alpha\gamma^2}\right)$, then $\frac{1}{n}\sum_{i\in[n]} \pi(W_i|W) \leq \mathsf{suc}_{7r}(\mu, f, C) + \gamma/\alpha$.*

First let us see how Lemma 4 implies Theorem 2. As outlined above, let W denote the event that π computes f correctly in all n coordinates. So, $(1/n)\sum_{i\in[n]}\pi(W_i|W)=1$. Set $\gamma=\alpha(1-\mathsf{suc}_{7r}(\mu,f,C))/2$ so that $\mathsf{suc}_{7r}(\mu,f,C)+\gamma/\alpha<1$. Then by Lemma 4, either $\|\pi\|<2$, $\|\pi\|\geq\alpha n\gamma^2\left(C-\frac{r\log(r/2\gamma)}{\alpha\gamma}-\frac{r}{\alpha\gamma^2}\right)$, or $\pi(W)<2^{-\gamma^2 n}$. It therefore remains to prove Lemma 4.

The overall idea is to use π to produce a $7r$-round protocol with communication complexity $<C$ that computes f correctly with probability at least $(1/n)\sum_{i=1}^{n}\pi(W_i|W)-O(\gamma)$. This would imply that $(1/n)\sum_{i\in[n]}\pi(W_i|W)\leq\mathsf{suc}_{7r}(\mu,f,C)+O(\gamma)$, as desired. The first step is to show that there exists a good simulating protocol for a random coordinate of $\pi|W$, whose average information cost is low (roughly $\|\pi\|/n$) and still uses only r rounds. The existence of such protocol was proven in [6], except their protocol is not guaranteed to actually have low information cost, but to merely be statistically close to a low-information protocol. This will suffices for our purpose:

Lemma 5 (Claims 26 and 27 from [6], restated). *There is a protocol σ taking inputs $x,y\sim\mu$ so that the following holds:*

- *σ publicly chooses a uniform $i\in[n]$ independent of x,y, and S_i which is part of the input to π.*
- $\mathbb{E}_{x,y,m,i,s_i}|\sigma(xys_im)-\pi(x_iy_is_im|W)|\leq 2\gamma$.
- $Rounds(\sigma)=Rounds(\pi)$.
- $\mathbb{E}_i\left[I_\pi(X_i;M|Y_iS_iiW)+I_\pi(Y_i;M|X_iS_iiW)\right]\leq 4\|\pi\|/n$.

The second step of the proof of Lemma 4 is to compress the simulating protocol σ so that it actually has low communication, without introducing many additional rounds in the compression process. Since the second and fourth propositions of Lemma 5 imply that σ is 2γ-close to a low-information distribution $q=\pi(x_iy_is_im|W)$, this is precisely the setting of Theorem 3. A formal proof of Lemma 4 can be found in the full version of this paper [7].

References

1. Barak, B., Braverman, M., Chen, X., and Rao, A. How to compress interactive communication. In *Proceedings of the 2010 ACM International Symposium on Theory of Computing* (2010), pp. 67–76.
2. Ben-Aroya, A., Regev, O., and de Wolf, R. A hypercontractive inequality for matrix-valued functions with applications to quantum computing. In *FOCS* (2008), pp. 477–486.
3. Braverman, M. Coding for interactive computation: progress and challenges. In *50th Annual Allerton Conference on Communication, Control, and Computing* (2012).
4. Braverman, M. Interactive information complexity. In *Proceedings of the 44th symposium on Theory of Computing* (New York, NY, USA, 2012), STOC '12, ACM, pp. 505–524.
5. Braverman, M., and Rao, A. Information equals amortized communication. In *FOCS* (2011), R. Ostrovsky, Ed., IEEE, pp. 748–757.

6. Braverman, M., Rao, A., Weinstein, O., and Yehudayoff, A. Direct products in communication complexity. *Electronic Colloquium on Computational Complexity (ECCC) 19* (2012), 143.
7. Braverman, M., Rao, A., Weinstein, O., and Yehudayoff, A. Direct product via round-preserving compression. *Electronic Colloquium on Computational Complexity (ECCC) 20* (2013), 35.
8. Brody, J., Chakrabarti, A., and Kondapally, R. Certifying equality with limited interaction.
9. Clarkson, K. L., and Woodruff, D. P. Numerical linear algebra in the streaming model. In *Proceedings of the 41st annual ACM symposium on Theory of computing* (2009), ACM, pp. 205–214.
10. Harsha, P., Jain, R., McAllester, D. A., and Radhakrishnan, J. The communication complexity of correlation. In *IEEE Conference on Computational Complexity* (2007), IEEE Computer Society, pp. 10–23.
11. Jain, R. New strong direct product results in communication complexity.
12. Jain, R., Pereszlenyi, A., and Yao, P. A direct product theorem for the two-party bounded-round public-coin communication complexity. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on* (2012), IEEE, pp. 167–176.
13. Jain, R., and Yao, P. A strong direct product theorem in terms of the smooth rectangle bound. *CoRR abs/1209.0263* (2012).
14. Kerenidis, I., Laplante, S., Lerays, V., Roland, J., and Xiao, D. Lower bounds on information complexity via zero-communication protocols and applications. *Electronic Colloquium on Computational Complexity (ECCC) 19* (2012), 38.
15. Klauck, H. A strong direct product theorem for disjointness. In *STOC* (2010), pp. 77–86.
16. Lee, T., Shraibman, A., and Spalek, R. A direct product theorem for discrepancy. In *CCC* (2008), pp. 71–80.
17. Lee, T., Shraibman, A., and Spalek, R. A direct product theorem for discrepancy. In *Computational Complexity, 2008. CCC'08. 23rd Annual IEEE Conference on* (2008), IEEE, pp. 71–80.
18. Molinaro, M., Woodruff, D., and Yaroslavtsev, G. Beating the direct sum theorem in communication complexity with implications for sketching. In *SODA* (2013), p. to appear.
19. Parnafes, I., Raz, R., and Wigderson, A. Direct product results and the GCD problem, in old and new communication models. In *Proceedings of the 29th Annual ACM Symposium on the Theory of Computing (STOC '97)* (New York, May 1997), Association for Computing Machinery, pp. 363–372.
20. Raz, R. A parallel repetition theorem. *SIAM Journal on Computing 27*, 3 (June 1998), 763–803. Prelim version in STOC '95.
21. Shaltiel, R. Towards proving strong direct product theorems. *Computational Complexity 12*, 1-2 (2003), 1–22. Prelim version CCC 2001.
22. Sherstov, A. A. Strong direct product theorems for quantum communication and query complexity. *SIAM Journal on Computing 41*, 5 (2012), 1122–1165.
23. Viola, E., and Wigderson, A. Norms, xor lemmas, and lower bounds for polynomials and protocols. *Theory of Computing 4*, 1 (2008), 137–168.
24. Yao, A. C.-C. Probabilistic computations: Toward a unified measure of complexity. In *18th Annual Symposium on Foundations of Computer Science* (1977), IEEE, pp. 222–227.