Randomised Individual Communication Complexity

Harry Buhrman [*]
Centrum voor Wiskunde en Informatica and the University of Amsterdam.
Kruislaan 413, 1098 SJ Amsterdam The Netherlands. email: buhrman@cwi.nl

Michal Koucký [†]
Institute of Mathematics of the Academy of Sciences of the Czech Republic,
Praha, Czech Republic. e-mail: koucky@math.cas.cz

Nikolai Vereshchagin[‡]
Department of Mathematical Logic and Theory of Algorithms,
Faculty of Mechanics and Mathematics,
Moscow State University, Leninskie Gory, Moscow, Russia 119991. email: ver@mccme.ru

Abstract

In this paper we study the individual *communication complexity of the following problem. Alice receives an input string x and Bob an input string y, and Alice has to output y. For deterministic protocols it has been shown [3] that $C(y)$ many bits need to be exchanged even if the actual amount of information $C(y|x)$ is much smaller than $C(y)$. It turns out that for randomised protocols the situation is very different. We establish randomised protocols whose communication complexity is close to the information theoretical lower bound. We furthermore initiate and obtain results about the randomised* round *complexity of this problem and show trade-offs between the amount of communication and the number of rounds. In order to do this we establish a general framework for studying these types of questions.*

1 Introduction

In this paper we continue the study of *individual* communication complexity introduced in [3]. We focus our attention on the following communication problem denoted by I. Alice has a binary string x and Bob a binary string y, both of the length n. Alice has to output y. We are interested in the communication complexity of problem I: how many bits have to be transmitted between Alice and Bob in order to let Alice learn Bob's string?

It is easy to see that the worst case complexity, i.e. the maximum complexity over all inputs of the same size, of this problem for deterministic protocols is n. The same applies for randomised protocols in both public and private coin models.

Assume therefore that x and y are somehow correlated. The problem has been studied in the following frameworks of this kind:

- We assume that a probability distribution over input pairs is given and we allow the protocol to be incorrect on a small fraction of input pairs (according to the given distribution); we consider worst case communication length. In such setting the problem was studied in [13, 10].
- We assume that a probability distribution over input pairs is given (known both to Alice and Bob) and we measure the average length of communication on a random input pair picked according to that distribution. In such setting the problem was studied in [12].
- We assume that the input pair belongs to a predetermined set R of pairs (known both to Alice and Bob) and consider worst case communication length over all input pairs from R. In such setting the problem was studied in [10, 11, 8, 1] (see also Chapter 4.7 of the textbook [4]).

[*] Partially supported by BRICKS project AFM1, EU project QAP and NWO VICI grant 2004-2009.

[†] Part of this work was done while visiting CWI supported in part by NWO VICI project 2004-2009. Partially supported by grant 201/07/P276 of GA ČR, project No. 1M0021620808 of MŠMT ČR, and Institutional Research Plan No. AV0Z10190503.

[‡] Part of this work was done while visiting CWI. Supported in part by Grants 03-01-00475, 02-01-22001, 358.2003.1 from Russian Federation Basic Research Fund.

In this paper we study this problem in none of these three settings. We allow all input pairs (thus Alice and Bob have no a priori knowledge about the input pair) and we do not consider any probability distribution on pairs. We are interested in how the communication length depends on the input pair. The only paper studying the problem in such setting is [7]. There, a deterministic 1 round protocol with communication length $C(y|x)+O(\log n)$ (conditional Kolmogorov complexity of y given x) is designed that solves the problem correctly on all input pairs provided Bob gets $O(\log n)$ bits of extra information from a third party who knows both x, y. The first result of our paper can be considered as getting rid, in this protocol, of extra $O(\log n)$ bits of help information at the expense of increasing the number of rounds and allowing randomised protocols.

To introduce our results we start with a toy example. Consider the following protocol. Bob with input y sends the bit 0 if y is the string that contains only 0's, otherwise he sends a 1 followed by y. Clearly, for $y = 00\dots0$ this protocol communicates only one bit, and for all other y's it communicates $n+1$ bits.

This protocol can be easily improved by letting Bob send a program that outputs y instead of sending y itself. This corresponds then to sending $C(y)$ bits from Bob to Alice, where $C(y)$ is the Kolmogorov complexity of y. No deterministic protocol can do better [3], i.e., the individual communication complexity of this problem is essentially $C(y)$. In particular, the string x which Alice holds is of no use for any protocol.

When one considers *randomised* protocols one can make use of Alice's string x as the following protocol demonstrates. Alice and Bob first perform a randomised equality test of x and y which is well known to require only $O(1)$ bits of communication. If they conclude that $x = y$ then Alice outputs x otherwise Bob sends a description of y to Alice and she outputs y. Thus, the protocol requires only $O(1)$ bits of communication on inputs where $x = y$ and at most $C(y) + O(1)$ otherwise. The individual communication complexity of the string y has gone down significantly when Alice's input x equals y. The question we are investigating here is how Alice's input x influences the cost of communicating Bob's y.

We say that $R^A_\varepsilon(I, x, y) \leq g(x, y)$ if there is a public coin randomised protocol in which Alice outputs Bob's y with probability at least $1 - \varepsilon$ after communicating at most $g(x, y)$ bits. Since Newman (see [4, Th. 3.14]) shows that a public coin protocol can be turned into a private one using an additional $O(\log n)$ bits of communication our results can be easily adapted to such a setting as well.

We first observe that the above mentioned protocol based on equality tests can be generalised as follows. Nisan [9] has proved that the worst case randomised complexity of the predicate GT (x is lexicographically greater than y) on strings of length n is $O(\log n)$ (for any fixed positive probability of error). By the same argument one can prove that the worst case randomised complexity of computing the length of the largest common prefix $lcp(x, y)$ of x and y is also $O(\log n)$ (for completeness' sake we present the proof in Section 3.1). This implies that the individual complexity of our problem I is at most

$$R^A_\varepsilon(I, x, y) \leq n - lcp(x, y) + O(\log n)$$

for any constant positive ε. (Bob first learns $lcp(x, y)$ and then communicates to Alice the missing $n - lcp(x, y)$ bits of y.)

There are many other ways than having a common prefix in which x and y can share a common information. For example x and y can differ in the first bit and have the same suffix of length $n - 1$. Clearly, for such x, y the randomised individual communication complexity of problem I is also small. To catch all possible ways to have common information consider the conditional Kolmogorov complexity $C(y|x)$ of y given x.

In this paper we compare the randomised individual communication complexity of problem I to $C(y|x)$. Namely, we show that $R^A_\varepsilon(I, x, y) \leq C(y|x) + O(\sqrt{C(y|x)})$.

Is it possible to do substantially better? That is, is it possible to solve I with communication length less than $C(y|x)$? The answer is no. We prove that if a randomised protocol solves I with probability of error $\varepsilon < 1$ for all x, y, then for all x and i there is y such that $C(y|x) < i$ and the protocol communicates on x, y at least $i + \log(1-\varepsilon) - O(1)$ bits.

Our protocol achieving communication $C(y|x) + O(\sqrt{C(y|x)})$ runs in $O(\sqrt{C(y|x)})$ rounds. We can also obtain a protocol that solves the problem in $O(\log C(y|x))$ rounds by exchanging only $(1 + \delta)C(y|x)$ bits for an arbitrary constant $\delta > 0$. Is it possible to reduce the number of rounds in these protocols to a constant keeping the communication length close to $C(y|x)$? We don't know the answer to this question.

The paper [7] shows that there is a deterministic 1 round protocol with communication length $C(y|x) + O(\log n)$ provided Bob gets $O(\log n)$ bits of extra information from a third party who knows both x, y.

It seems that the core of the problem lies in estimating $C(y|x)$ efficiently. In particular we show that if Alice and Bob know $C(y|x)$ and $C(x|y)$ then they can solve the problem I deterministically in a single round by communicating $2C(y|x) + 2C(x|y)$ bits. In randomised setting a single round and only $C(y|x) + C(x|y)$ bits suffice.

Thus we identify the problem of efficiently approximating $C(y|x)$ as a crucial problem. We say that a protocol Δ-approximates a function $f(x, y)$ if on input x to Alice and y to Bob the protocol outputs a value between $f(x, y)$

and $f(x,y)+\Delta$. In the case of a randomised protocol with error probability ε this should happen with probability at least $1-\varepsilon$ for each x and y. We show that a k-round randomised protocol that Δ-approximates $C(y|x)$ with probability of error $\varepsilon < 1/2$ must communicate $\Omega((n/\Delta)^{1/k})$ bits. This can be translated into a lower bound $\Omega(n^{1/k})$ on $R_\varepsilon^{A,\text{k-rounds}}(I,x,y)$ as once Alice knows x and y she can determine $C(y|x)$.

So far we have compared the cost of the protocol with the conditional Kolmogorov complexity of x and y. Since Alice and Bob are all powerful there really seems to be no good reason why Kolmogorov complexity should be the right measure to consider. For example Alice and Bob could a priori agree on some non-computable oracle and use Kolmogorov complexity relative to that oracle. It turns our that most of the results stated so far hold not only for Kolmogorov complexity but rather for any measure that we call a *normalised function*. A function $d : \{0,1\}^* \times \{0,1\}^* \to \mathbb{N} \cup \{+\infty\}$ is *normalised* if for all x, i the number of y with $d(x,y) < i$ is less than 2^i. The normalisation condition requires that the number of y that are close to x is small. Clearly $C(y|x)$ is normalised and it is essentially optimal among all lower semi-computable normalised functions. Another useful normalised function is the function $s(x,y) = n - lcp(x,y) - 1$ where n is the length of x and y; if the lengths of x and y are different, the function $s(x,y)$ takes the value $+\infty$.

Our upper bound $R_\varepsilon^A(I,x,y) \le d(x,y) + O(\sqrt{d(x,y)})$ as well as $R_\varepsilon^A(I,x,y) \le (1+\delta)d(x,y)$ hold for any normalised function $d(x,y)$. Also, if Alice and Bob know $d(x,y)$ and $d(y,x)$ for their inputs x and y, then they can in a single deterministic round solve the problem I using $2d(x,y) + 2d(y,x)$ bits of communication. If the function $d(x,y)$ satisfies the additional property that for some constant c and all x the number of y such that $d(x,y) < i$ is at least 2^{i-c}, then for any randomised protocol that solves I with probability of error $\varepsilon < 1$ for all x, y, for every x and i we can find y such that $d(x,y) < i$ and the protocol communicates on x, y at least $i - c + \log(1-\varepsilon)$ bits.

In order to show stronger and more interesting lower bounds one has to however focus on particular functions $d(x,y)$. For example in the case of $d(x,y) = |y|$, one can solve I in one round with communication $d(x,y)$.

The normalised function $s(x,y)$ plays a key role in our lower bound proof for Δ-approximating $C(y|x)$. In fact the $\Omega((n/\Delta)^{1/k})$ lower bound holds for Δ-approximating $s(x,y)$ in k rounds as well. However as opposed to $C(y|x)$ we know how to efficiently approximate $s(x,y)$: we can Δ-approximate $s(x,y)$ in k rounds using $O((n/\Delta)^{1/k})$ communication. This allows us to solve problem I in k rounds using at most $s(x,y) + O(n^{1/k})$ communication, which for $k = 1, 2$ is tight.

We conclude by three interesting open questions:

1. Is there a randomised protocol to solve problem I in constant rounds by exchanging at most $C(y|x) + o(n)$ bits?
2. What is the worst case randomised complexity of approximating $C(y|x)$?
3. What is the worst case randomised complexity of approximating $C(y|x)$ in k rounds (k is a constant)? We give only a lower bound and only for error probability less than 1/2.

2. Notation and definitions

We establish our notation here. We consider the usual model of two-party communication complexity. In this model Alice and Bob get inputs $x \in X$ and $y \in Y$, respectively, and either one or both of them want to learn the value of some function $f(x,y)$. More generally, for some relation $T \subseteq X \times Y \times Z$, they may want to learn some value z such that $(x,y,z) \in T$. (We will identify a function $f : X \times Y \to Z$ with its graph $\{(x,y,f(x,y)) \mid (x,y) \in X \times Y\}$.) They accomplish this goal by exchanging messages according to some fixed *protocol*. For each pair of inputs x and y we measure how many messages and how many bits of information were exchanged during the communication. We call the first quantity the *number of rounds* and the second quantity the *length of communication*.

To avoid any ambiguity we can define our terms formally as follows. A *protocol* P over domain $X \times Y$ with range Z is a finite rooted binary tree, whose nodes are divided into two parts, A and B, called Alice's nodes and Bob's nodes. (They indicate the turn of move.) Each internal Alice's node v is labelled by a function $a_v : X \to \{0,1\}$ and each internal Bob's node v is labelled by a function $b_v : Y \to \{0,1\}$. Each Alice's leaf v is labelled by $a_v : X \to Z$ and each Bob's leaf v is labelled by $b_v : Y \to Z$. A node *reached* by a protocol P on inputs x, y is the leaf reached by starting at the root of P and walking towards leaves where in each encountered internal node we go left if $a_v(x) = 0$ or $b_v(y) = 0$, depending on whose node v is, and we go right otherwise. We say that using P on input x and y, *Alice learns* a relation T if the leaf v reached on x and y belongs to Alice and satisfies $(x,y,a_v(x)) \in T$. Similarly, *Bob learns* T if the leaf v belongs to him and $(x,y,b_v(y)) \in T$.

The *length of communication* of the protocol P on inputs x, y is the length of the path from the root of P to the leaf reached on x, y. A protocol P runs in k rounds on inputs x, y if there are $k-1$ alternations between Alice's and Bob's nodes along the path from the root of the protocol to the leaf reached on x, y. The message sent by Alice in the l-th round of the protocol P on x, y is the concatenation of bits $a_{v_i}(x), a_{v_{i+1}}(x), \ldots, a_{v_j}(x)$ where $v_{i-1}, v_i, v_{i+1}, \ldots, v_{j+1}$ are the nodes along the path from the root the the leaf reached on x, y, and the $(l-1)$-th alternation between nodes labelled by b_u and a_u occurs between nodes v_{i-1} and v_i, and the l-th alternation occurs between v_j and v_{j+1}.

A randomised protocol is a probability distribution μ on protocols. For a relation T, we say that $R^A_\varepsilon(T,x,y) \le g(x,y) + O(g'(n))$, if there exist a randomised protocol μ and a constant $c > 0$ such that for every $(x,y) \in X \times Y$, with μ-probability at least $1-\varepsilon$ for a random protocol P the length of communication on x,y is at most $g(x,y) + cg'(n) + c$ and Alice learns T on x,y (where n is the length of x,y). Similarly, $R^B_\varepsilon(T,x,y) \le g(x,y) + O(g'(n))$ for the case where Bob learns T. We say that $R^A_\varepsilon(T,x,y) \ge g(x,y) + \Omega(g'(n))$ if for every randomised protocol μ and every function $h(n)$ with $R^A_\varepsilon(T,x,y) \le g(x,y) + h(n)$ we have $h(n) = \Omega(g'(n))$.

If $f : \{0,1\}^* \times \{0,1\}^* \to \mathbb{N}$ and $\alpha : \mathbb{N} \to \mathbb{N}$ are functions, Alice or Bob may want to learn the value of $f(x,y)$ only α-approximately. This means that they want to learn the relation $T = \{\langle x,y,i\rangle \mid f(x,y) \le i \le f(x,y)+\alpha\}$. We write $R^B_\varepsilon(f,\alpha,n) \le g(n) + O(g'(n))$, if $R^B_\varepsilon(T,x,y) \le g(|x|) + O(g'(|x|))$. In the similar way we define $R^B_\varepsilon(f,\alpha,n) \ge g(n) + \Omega(g'(n))$.

When we restrict ourselves to protocol with k rounds we will use notation $R^{\cdot,k\text{-rounds}}_{\cdots}(\cdots)$ to denote the appropriate quantity.

A function $d : \{0,1\}^* \times \{0,1\}^* \to \mathbb{N} \cup \{+\infty\}$ is *normalised* if for all x,i the number of y with $d(x,y) < i$ is less than 2^i. Normalisation condition requires that the number of y that are close to x is small. The conditional Kolmogorov complexity $C(y|x)$ is *minimal* (up to an additive constant) among all normalised functions for which the set $\{\langle x,y,i\rangle \mid d(x,y) < i\}$ is computably enumerable. This means that for every such normalised function d there is a constant c such that $C(y|x) \le d(x,y) + c$ for all x,y. This property of $C(x|y)$ can be considered as its definition since it defines it uniquely up to an additive constant.

We will use the following probability distribution on $\{0,1\}^n \times \{0,1\}^n$. To generate a random pair (x,y) choose first a random number $i \in \{0,1,2,\dots,n-1\}$, and then choose uniformly at random a pair (x,y) with $lcp(x,y) = i$. That is, the probability of a pair (x,y) is equal to $2^{lcp(x,y)-2n+1}/n$. We call this distribution *quasi-uniform.*

3. Non-constant number of rounds

In this section we consider protocols using non-constant number of rounds. We start with the following theorem.

THEOREM 1. *For all normalised functions d, $R^A_\varepsilon(I,x,y) \le d(x,y) + O(\sqrt{d(x,y)})$.*

Before proving the theorem we start with some remarks explaining why the problem to communicate y to Alice by exchanging about $d(x,y)$ bits is hard.

If Bob knew x then he could just send to Alice the index of y in the set $\{y' \mid d(x,y') = d(x,y)\}$. By normalisation requirement the binary length of the index is at most $d(x,y)+1$.

Assume now that Bob knows $d(x,y)$ but does not know x. One can prove that, for $d(x,y) = C(y|x)$, it is impossible to transmit y to Alice in about $d(x,y)$ bits deterministically (see [3]). Using randomness, we employ the technique of fingerprints and proceed as follows: Bob applies a random linear function $L : \{0,1\}^n \to \{0,1\}^m$ over $GF(2)$ to his input y (called the fingerprint of y in the sequel) and sends it to Alice. The coefficients of L are read from the shared random source. Since we assume that Bob knows $d(x,y)$ he can choose $m = \log(1/\varepsilon) + d(x,y)$. Alice compares $L(y)$ with fingerprints $L(y')$ for all y' with $d(x,y') = d(x,y)$. She outputs the first such y' with $L(y') = L(y)$. By union bound the probability of error ($y' \neq y$ but $L(y') = L(y)$) in this protocol is at most $2^{d(x,y)}2^{-m} = \varepsilon$. We now turn to the actual proof.

Proof of Theorem 1. Since Bob knows only y, Bob will try to guess an upper bound for $d(x,y)$ probing the numbers $1,3,6,\dots,j(j+1)/2,\dots$ successively. At the beginning of the protocol he prepares fingerprints $L_1(y), L_2(y),\dots,$ where each $L_j : \{0,1\}^n \to \{0,1\}^{m_j}$ is a linear function chosen independently at random using the shared random source, and m_j will be specified later. Then our protocol runs in rounds numbered $1,2,\dots$. In round $2j-1$ Bob sends to Alice the fingerprint $L_j(y)$. Then Alice in round $2j$ looks for a string y' with

$$d(x,y') \le j(j+1)/2$$

such that $L_i(y') = L_i(y)$ for all $i = 1,2,\dots,j$. If she succeeds to find such y', she outputs it, informs Bob that she has succeeded and stops the protocol. Otherwise she informs Bob that she has not succeeded and the protocol continues.

Clearly, the protocol stops after at most $2j$ rounds where j is minimal such that $d(x,y) \le j(j+1)/2$. As $j = O(\sqrt{d(x,y)})$ the protocol works in these many rounds and exchanges at most $j + m_1 + \cdots + m_j$ bits.

Now we will adjust m_j so that the probability of error is less than ε. An error may occur only if there are $y' \neq y$ and $k \le j$ such that $d(x,y') \le k(k+1)/2$ and $L_i(y') = L_i(y)$ for all $i = 1,2,\dots,k$. For fixed k and y' the probability that the k fingerprints of y and y' coincide is $2^{-m_1-\cdots-m_k}$. Since $d(x,y')$ is normalised, by union bound over y' the probability of error is less than $2^{k(k+1)/2-m_1-\cdots-m_k}$. Thus if we let $m_1 = 2 + \log 1/\varepsilon$ and $m_j = j+1$ for $j = 2,\dots,$ then for fixed k the probability of error is at most $\varepsilon 2^{-k}$. By union bound over k the total probability of error is less than

ε. The number of communicated bits is

$$\begin{aligned}&j + m_1 + \cdots + m_j \\ &= 2j + j(j+1)/2 + \log(1/\varepsilon) \\ &= d(x,y) + O(\sqrt{d(x,y)}) + \log(1/\varepsilon). \quad \square\end{aligned}$$

In the previous proof we have chosen $m_1, m_2, \ldots$ so as to balance the number of rounds of the protocol and the quality of the estimate on $d(x,y)$. If for some $\delta > 0$ we chose $m_1 = 3 + \log 1/\varepsilon$ and $m_j = (1+\delta)^j$ for $j = 2, \ldots$ in the previous proof and probed as upper bounds for $d(x,y)$ the values of the form $(1+\delta)^j$, we would obtain a protocol that runs in $O(\log d(x,y)/\log(1+\delta))$ rounds and exchanges $(1+\delta)d(x,y) + \log_{1-\delta} d(x,y) + \log 1/\varepsilon$ bits of information.

The following lemma provides a lower bound on the amount of communication needed for solving I and it is applicable to all $d(x,y)$ that satisfy certain natural property. That property holds for both $C(y|x)$ and $s(x,y)$.

LEMMA 1. *Assume that for some constant c and all x the number of y such that $d(x,y) < i$ is at least 2^{i-c}. If a randomised protocol solves I with probability of error $\varepsilon < 1$ for all x, y, then for all x and i there is y such that $d(x,y) < i$ but the protocol communicates on x, y at least $i - c + \log(1-\varepsilon)$ bits.*

Proof. The proof is based on the relation between randomised complexity and average case complexity. Fix x and i and assume that there is a protocol such that for all y with $d(x,y) < i$ with probability at least $1-\varepsilon$ Alice outputs y after receiving at most l bits from Bob. Consider the uniform probability distribution on y's such that $d(x,y) < i$. By standard arguments one can show that there is a deterministic protocol allowing Alice for a fraction at least $1-\varepsilon$ of y's to learn y after receiving at most l bits from Bob.

On the other hand, Alice's output depends only on the concatenation b of Bob's messages. Thus the cardinality of the set A of strings which Alice can output after receiving at most l bits from Bob is 2^l or less. As the number of y's with $d(x,y) < i$ is at least 2^{i-c}, $2^l \geq 2^{i-c}(1-\varepsilon)$. $\square$

For the sake of completeness we finish this section by presenting a protocol computing $lcp(x,y)$ with $O(\log n)$ communication and thus enabling Bob to transmit his input to Alice with communication length $s(x,y) + O(\log n)$. The proof is the same as that of Nisan [9].

3.1. Computing the largest common prefix by exchanging $O(\log n)$ bits.

To find $lcp(x,y)$ with error probability at most ε Alice and Bob use binary search of the leftmost different bit in x and y. That is, they first apply the randomised equality test (RET) to x, y. If they find out that $x \neq y$, they apply RET to the first halves of x and y, and so on, $\log n$ times. At each time they have a pair of numbers l, r such that the length-r prefixes x_r, y_r of x, y are different and the length-l prefixes of x, y are supposedly equal. They recurse by applying RET to length-$(l+r)/2$ prefixes of x, y and replace either l, or r by $(l+r)/2$ depending of whether the outcome of the test is positive or negative. They use fingerprints of length $\log(\log n/\varepsilon)$. That implies that the probability of failure in each test is at most $\varepsilon/\log n$. By union bound the overall probability of error is at most ε.

This protocol communicates $O(\log n \log\log n)$ bits. To get rid of the factor $\log\log n$ we use the following trick. Instead of using long fingerprints we will use constant length, say length-3, fingerprints. To compensate the increase in error probability we will sometimes double check the equation $x_l = y_l$. More specifically, at each step we will have a pair of numbers l, r corresponding to a node in the search tree ($l = 0, r = n$ correspond to the root of the tree, and $l, (l+r)/2$ and $(l+r)/2, r$ to children of the node l, r). At the start we are in the root. We then repeat $5\log n$ times the following loop.

(1) Check whether $x_l = y_l$, using RET. If the outcome is negative, backtrack, that is, return to the parent of the current node and skip (2).
(2) Otherwise move to one of the children of the node l, r, as in the normal binary search (unless you are in the leaf, in which case skip this step).

The average number of failed tests is at most $5\log n/8$. By Chernoff bound with probability at least $1 - O(1/n)$ the number of failed tests is less than $\log n$. Let us prove that in this case we reach at some moment the target node in the search tree (it is easy to see, that if we get into the target node, we stay there forever).

By way of contradiction assume that this is not the case. It is easy to verify that then the following value

(the number of performed tests)
$+$ 2(the distance in the tree from the current node to the target node)
$-$ 4(the number of failed performed tests)

does not increase after each repetition of the loop. Indeed, if at least one test errs, then the last term decreases by at least four, which compensates the increase of the first two terms, as we make at most one move in the wrong direction and make at most 2 tests. If no tests err then we make a move in the right direction, thus the second term decreases by 2 while the first term increases by at most 2 (the last term does not change).

At the the start the distance to the destination is $\log n$ thus the value in question is always at most $\log n$. However at the end it is at least $5\log n + 2 - 4\log n > \log n$.

4. Constant number of rounds

In this section we consider the feasibility of reducing the number of rounds in the protocol of Theorem 1 to a constant while keeping the communication length close to $d(x, y)$. Whether that is possible of course might depend on the function d in question. For example, if $d(x, y) = |y|$ then the problem can be solved in 1 round and communication length $d(x, y)$. However we are interested primarily in the more natural functions like $C(y|x)$ and $s(x, y) = n - lcp(x, y) - 1$.

We do not know the answer in the case of $C(y|x)$. For $s(x, y)$ the answer is positive: we can solve the problem I in two rounds and communication length $s(x, y) + O(\sqrt{n})$, which is only slightly worse than the bound of Theorem 1. Moreover, in k rounds we can solve I with communication length $s(x, y) + O(n^{1/k})$:

THEOREM 2. *For all k and $\varepsilon > 0$ we have $R_\varepsilon^{A,k\text{-rounds}}(I, x, y) \le s(x, y) + O(n^{1/k})$. The constant in O-notation depends on k and ε.*

To prove this bound we first show that the randomised worst case complexity of approximating $lcp(x, y)$ by a k-round protocol is $O(n^{1/k})$.[1] Once Bob has learned the largest common prefix of x and y, he sends to Alice the last $s(x, y)$ bits of y. The described protocol works in $k+1$ rounds instead of the claimed k rounds. To obtain a k-round protocol, we modify the protocol so that Bob first approximates $lcp(x, y)$ in $k - 1$ rounds with precision $n^{1/k}$. That is, he finds a number p such that $p \le lcp(x, y) < p + n^{1/k}$. Then he sends Alice the last $n - p$ bits of y and Alice learns y.

Proof of Theorem 2. Let us show that

$$R_\varepsilon^{A,k\text{-rounds}}(lcp, \alpha, n) = O((n/\alpha)^{1/k}).$$

Instead of binary search Alice and Bob perform m-ary search, where $m = (n/\alpha)^{1/k}$. Divide the interval $\{1, 2, \dots, n\}$ into m segments of equal length. For every sub-segment Alice finds a fingerprint of length $l = \log(k/\varepsilon)$ of the corresponding sub-string of x. Then Alice sends all fingerprints to Bob and Bob compares them with the corresponding fingerprints of his sub-strings. He finds the leftmost segment where the fingerprints differ. In the second round Bob sends to Alice the number of that segment (if $k = 1$ then he just outputs that number times α). Then Bob divides it again in m sub-segments and for each sub-segment sends to Alice the fingerprint of length l of the corresponding sub-string of y. And so on for k rounds.

Let us estimate first the probability of an error. The length of the last segment is at most $n/m^k \le \alpha$. Thus an error can occur only if $lcp(x, y)$ does not belong to that segment. If this happens then there is a round in which the fingerprints of the leftmost different sub-strings of x and y coincide. By union bound the probability of this is at most $k2^{-l} = k2^{-\log(k/\varepsilon)} = \varepsilon$.

In each round except the last one they send $ml + \log m$ bits and in the last one ml bits. Thus the total length of the communication is $kml + (k - 1)\log m = O(n^{1/k})$.

To finish the proof of the theorem, first apply $k-1$ round protocol allowing Bob to $n^{1/k}$-approximate $lcp(x, y)$ with communication length $O((n^{1-1/k})^{1/(k-1)}) = O(n^{1/k})$. Bob then learns a number p such that $p \le lcp(x, y) < p + n^{1/k}$. Then he sends to Alice the last $n - p < n - lcp(x, y) + n^{1/k}$ bits of y and Alice learns y. □

4.1. Tightness results for Theorem 2

In the case of one and two rounds we are able to show that the bound in Theorem 2 is tight.

THEOREM 3. *For $k = 1, 2$ and all $\varepsilon < 1$ we have $R_\varepsilon^{A,k\text{-rounds}}(I, x, y) \ge s(x, y) + \Omega(n^{1/k})$. This means that there is a positive δ depending on ε such that for all large enough n and all k-round protocols there are x, y of length n such that the communication length of the protocol on x, y is at least $s(x, y) + \delta n^{1/k}$. As $C(y|x) \le s(x, y) + O(1)$, the same lower bound holds for $C(y|x)$ in place of $s(x, y)$.*

Fist we prove the theorem for $k = 1$.

Proof. We use again relation between randomised and average case complexity.

Fix $\varepsilon < 1$ and n and assume that there is a randomised 1-round protocol such that for all x, y with probability $1 - \varepsilon$ the communication length is at most $s(x, y) + l$ and Alice learns y. Then there is a deterministic 1-round protocol such that for at least $1 - \varepsilon$ fraction of the input pairs (x, y) (with respect to the quasi-uniform distribution) the communication length is at most $s(x, y) + l$ and Alice learns y.

Depending on his input y Bob chooses a message b and sends it to Alice. Based on b and her input x Alice outputs a string y'. We need to upper bound the fraction of pairs x, y such that $y' = y$ and $|b| \le s(x, y) + l$ (call such pairs *successful*). To this end fix a message b and let Y_b denote the set of all Bob's inputs y for which he sends b. Note that for every x there is at most one successful pair x, y with $y \in Y_b$.

Let p_b denote the total probability of all successful pairs (x, y) such that $y \in Y_b$.

LEMMA 2. $p_b \le \lambda(Y_b) \cdot \frac{l+2-\log\lambda(Y_b)-|b|}{n}$ *where λ stands for the uniform probability distribution on $\{0, 1\}^n$.*

[1] The same upper bound holds for GT, which is a slight improvement of the known bound $R_\varepsilon^{k\text{-rounds}}(GT, n) = O(n^{1/k}\log n)$ (stated in [6] without a proof).

Proof. By definition p_b is the sum of probabilities of successful pairs x, y with $y \in Y_b$. Fix i and upper bound the contribution to this sum of all pairs (x, y) with $s(x, y) = i$. By the definition of the quasi-uniform probability, the contribution of each such pair is $2^{-i-n}/n$.

We distinguish three cases: (a) $i < |b| - l$, (b) $|b| - l \leq i < n - k$ where k is the smallest integer number such that $\lambda(Y_b) \leq 2^{k-n}$, and (c) $i \geq n - k$.

Case (a). The contribution of all such pairs is zero, as for every successful pair $s(x, y) \geq |b| - l$.

Case (b). We upper-bound the quasi-uniform measure of *all* pairs (successful or not) such that $|b| - l \leq s(x, y) < n - k$.

For every fixed y there are 2^i pairs with $s(x, y) = i$ and the contribution of each such pair is $2^{-i-n}/n$. Thus the contribution of all pairs (x, y) with $s(x, y) = i$ is at most $|Y_b|2^{-n}/n = \lambda(Y_b)/n$.

Case (c). The contribution of every pair with $s(x, y) \geq n - k$ is at most $2^{k-2n}/n$. For every x there is at most one successful pair x, y with $y \in Y_b$, hence the contribution of all such successful pairs is at most $2^{k-n}/n$.

Thus we obtain

$$\begin{aligned} p_b &\leq \frac{\lambda(Y_b)[(n-k) - (|b| - l)] + 2^{k-n}}{n} \\ &\leq \frac{\lambda(Y_b)(-\log \lambda(Y_b) - |b| + l)) + 2\lambda(Y_b)}{n}. \quad \square \end{aligned}$$

The lemma implies that the total probability of all successful pairs (for all b) is at most

$$\begin{aligned} &\frac{\sum_b \lambda(Y_b)(l + 2 - \log \lambda(Y_b) - |b|)}{n} \\ &= \frac{l + 2 + \sum_b \lambda(Y_b)(-\log \lambda(Y_b)) - \sum_b \lambda(Y_b)|b|}{n}. \end{aligned}$$

Consider the mapping $Y_b \mapsto b$ as a prefix code of the source consisting of all different Y_b's, where probability of the source letter Y_b is $\lambda(Y_b)$. By Shannon's Noiseless Coding Theorem we have $\sum_b \lambda(Y_b)|b| \geq \sum_b \lambda(Y_b)(-\log \lambda(Y_b))$ and thus the total probability of all successful pairs is at most $(l + 2)/n$. Hence $l \geq (1 - \varepsilon)n - 2$. □

4.1.1 Proof of Theorem 3 for $k = 2$

Fix $\varepsilon < 1$ and n and assume that there is a randomised 2-round protocol such that for all x, y with probability $1 - \varepsilon$ Alice learns y after exchanging at most $s(x, y) + l$ bits with Bob (Alice sends the first message and Bob the second one). Then there is a deterministic 1-round protocol such that at least $1 - \varepsilon$ fraction of the input pairs (x, y) (with respect to the quasi-uniform distribution) is successful, that is, Alice learns y after exchanging at most $s(x, y) + l$ bits with Bob.

Fix such deterministic protocol and fix any positive $\delta < 1 - \varepsilon$. We first show that this protocol allows Bob in 1 round to $l - \log \delta$-approximate $lcp(x, y)$ on a fraction $1 - \varepsilon - \delta$ of all input pairs. Then we show that this is possible only when $l = \Omega(\sqrt{n})$.

LEMMA 3. *For at least $1 - \varepsilon - \delta$ of input pairs picked according to the quasi-uniform distribution the length of Bob's message is between $s(x, y) + \log \delta$ and $s(x, y) + l$ and the length of Alice message is at most $l - \log \delta$.*

Proof. Estimate the fraction of successful inputs pairs x, y such that Bob communicates to Alice $i + \log \delta$ bits and Alice's output is correct. Fix x and i. The string output by Alice depends only on x and Bob's message. Thus the number of different y's printed after receiving less than $i + \log \delta$ bits from Bob is less than $2^{i+\log \delta} = \delta 2^i$. On the other hand, there are 2^i different y's for which $s(x, y) = i$. Hence for every fixed x, i the fraction of y's such that (x, y) is a successful pair and Bob sends less than $i + \log \delta$ bits among all y's with $s(x, y) = i$ is less than δ. Averaging over x, i shows that the fraction of all successful input pairs on which Bob sends to Alice at most $s(x, y) + \log \delta$ bits is at most δ.

Thus for at least $1 - \varepsilon - \delta$ input pairs Bob's message has at least $s(x, y) + \log \delta$ bits and Alice's message has at most $l - \log \delta$ bits. □

We have shown that for at least $1 - \varepsilon - \delta$ input pairs the length of Bob's message $l - \log \delta$-approximates $s(x, y)$. Thus there is a 1 round protocol with communication length $l - \log \delta$ to $l - \log \delta$-approximate $lcp(x, y)$ on $1 - \varepsilon - \delta$ input pairs.

LEMMA 4. *If a deterministic 1-round protocol on at least $1 - \varepsilon$ pairs x, y (with respect to the quasi-uniform probability distribution) α-approximates $s(x, y)$ transmitting at most l bits then $l = \Omega(n/\alpha)$.*

Proof. The assumption implies that there is a deterministic 1-round protocol such that on a fraction of at least $(1-\varepsilon)/\alpha$ pairs Bob computes $lcp(x, y)$ exactly.

Fix a message a sent by Alice and let $X = X(a)$ be the set of all x such that Alice sends the message a on input x. After receiving a Bob knows that Alice's string is in X. Depending on his string y, Bob chooses a number $b(y)$. The goal of Bob is to maximise the quasi-uniform probability of the set $Z = \{(x, y) \in X \times \{0, 1\}^n \mid b(y) = lcp(x, y)\}$. We need to prove that whatever function $b(y)$ Bob chooses, the quasi-uniform probability of the set Z is small.

For $X \subseteq \{0, 1\}^n$, let $\lambda(X) = |X|/2^n$ denote the density of X.

LEMMA 5. *For every function b mapping strings of length n to integers $\{0, 1, \ldots, n - 1\}$ and every set X of strings of length n the quasi-uniform probability of the set Z is at most $2\lambda(X)(1 + \ln \lambda(X)^{-1})/n$.*

Proof. Fix X. Then the set Z depends only on the function $y \mapsto b(y)$. We use a game interpretation of the quasi-uniform probability of Z. Consider the following game played by one player against a "random" adversary. The adversary chooses at random a string y of length n. We choose its non-empty prefix z and flip the last bit of z. Then the adversary chooses a random continuation x of length n of the resulting string z'. We gain 1 if x gets into X and 0 otherwise.

Our strategy is essentially to choose z of length $b(y)+1$. Clearly, our gain in the game is n-times the quasi-uniform measure of Z.

Let us modify slightly this game: in the modified game we do not flip the last bit of z and we are allowed to choose the empty prefix. The maximal gain in the modified game is at least half of the maximal gain in the original game. Thus it suffices to prove that the maximal gain in the second game is at most $\lambda(X)(1+\ln\lambda(X)^{-1})$.

We will prove this claim by induction on n. To prove the induction step, we have to generalise the (second) game as follows.

Let r by a real number in $[0,1]$. Consider the third game: the adversary chooses at random a string y of length n. Then we either stop the game, in which case we gain r. Or we continue as in the second game.

Let $w^r(X)$ denote the maximal possible gain in the last game. Note that the second game is a special case of the last game ($r=0$). Thus it suffices to prove the following

Claim 1. *For all r, X we have*

$$w^r(X) \le f^r(\lambda(X))$$

where

$$f^r(u) = \begin{cases} r + u\ln(1/r), & \text{if } u \le r,\\ u + u\ln(1/u), & \text{if } u \ge r.\end{cases}$$

Note that $f^0(u) = u + u\ln(1/u)$. We prove the statement by induction on n. The tricky definition of $f^r(u)$ is explained as follows. To make the induction step we need $f^r(u)$ to be concave in u and satisfy the inequality $f^u(u) \le f^r(u)$ for $r \le u$. The function $f^r(u)$ is indeed concave: both functions $r + u\ln(1/r)$, $u + u\ln(1/u)$ are concave and in the point $u = r$ they coincide and have the same derivative. For the base of induction we need the inequalities: $r \le f^r(0)$ and $1 \le f^r(1)$. One can verify that $f^r(u)$ is the minimal function satisfying these requirements.

Base of induction: $n = 0$. In this case we need to verify the inequality for $\lambda X = 0, 1$ as these are the only densities: $r \le f^r(0)$ and $1 \le f^r(1)$. Both inequalities are obvious.

Induction step. Let X_0 denote the set of all x of length $n-1$ such that $0x \in X$. Let X_1 be defined in a similar way. We claim that

$$w^r(X) \le \frac{w^{\max\{r,\lambda(X)\}}(X_0) + w^{\max\{r,\lambda(X)\}}(X_1)}{2}.$$

The first term in the numerator corresponds to y starting with 0 and the second one to the remaining y. If the adversary chooses y starting with 0 we have three options: either gain r, or gain $\lambda(X)$ (by letting let $b(y) = 0$), or let $b(y) > 0$. The last option means that we play the second game for X_0 in place of X. Thus, our average gain for y starting with 0 is at most $w^{\max\{r,\lambda(X)\}}(X_0)$. A similar bound holds for y starting with 1.

By induction hypothesis we have

$$w^{\max\{r,\lambda(X)\}}(X_0) \le f^{\max\{r,\lambda X\}}(\lambda X_0),$$
$$w^{\max\{r,\lambda(X)\}}(X_1) \le f^{\max\{r,\lambda X\}}(\lambda X_1).$$

Thus $w^r(X)$ does not exceed

$$\frac{f^{\max\{r,\lambda X\}}(\lambda X_0) + f^{\max\{r,\lambda X\}}(\lambda X_1)}{2}.$$

By concavity this does not exceed

$$f^{\max\{r,\lambda X\}}((\lambda X_0 + \lambda X_1)/2)$$
$$= f^{\max\{r,\lambda X\}}(\lambda X) = f^r(\lambda X). \quad \square$$

Let us finish the proof of Lemma 4. By Lemma 5 we have $\sum_a \lambda(X(a))(1+\ln\lambda(X(a))^{-1}) \ge (1-\varepsilon)n/2\alpha$. The left hand side is at most 1 plus the Shannon entropy of the message sent by Alice. The Shannon entropy of every random value having at most 2^l outcomes does not exceed $l\ln 2$ and we obtain the inequality $1 + l\ln 2 \ge (1-\varepsilon)n/2\alpha$. $\square$

By Lemmas 3 and 4 if $R_\varepsilon^{2-\text{rounds}}(I,x,y) \le s(x,y)+l(n)$ then $l(n) - \log\delta = \Omega(n/(l(n)-\log\delta))$ for some positive constant δ. This implies that $l(n) = \Omega(\sqrt{n})$. Theorem 3 is proved.

Remark. We could handle the case $k = 1$ similar to the case $k = 2$: with probability $1-\varepsilon-\delta$ the length of Bob's message $l - \log\delta$-approximates $lcp(x,y)$. Without any information of x the maximal probability of success in such approximation is $(l-\log\delta)/n$, thus $l \ge (1-\varepsilon-\delta)n + \log\delta$. The choice of δ that maximises this lower bound is $\delta = \log e/n$. In this way we obtain the bound $l \ge (1-\varepsilon)n - \log n - O(1)$, which is slightly worse than the bound $l \ge (1-\varepsilon)n - O(1)$ from the first proof.

4.2. Approximating $C(y|x)$

We have already noticed that the problem of transmitting Bob's input to Alice in k-rounds by exchanging about $C(y|x)$ bits reduces to approximating $C(y|x)$ in $k-1$ rounds by exchanging few bits. For the latter problem we are able to prove the following lower bound.

Theorem 4. *For all $k \ge 0$ and all $\varepsilon < 1/2$ we have $R_\varepsilon^{k\text{-rounds}}(C(y|x),\alpha,n) = \Omega((n/\alpha)^{1/k})$. The constant in Ω-notation depends on k and ε.*

Note that the bound of Theorem 4 does not imply similar bound for all $\varepsilon < 1$. We do not know if this is the case.

To prove Theorem 4 we use the round elimination techniques from [6]. Previously this techniques was used to obtain the bound $R_\varepsilon^{k\text{-rounds}}(GT, n) = \Omega(n^{1/k})$ for the randomised worst case complexity of GT predicate on n-length strings (x is lexicographically greater than y). It is based on the Round elimination lemma that implies the following: for every positive ε there is a positive δ such that if $R_\delta^{k \text{ rounds}}(GT, n) \le l$ then $R_\varepsilon^{k-1 \text{ rounds}}(GT, n/l) \le l$. We need an average case version of this lemma for the problem of approximating $lcp(x, y)$ in place of GT. An analysis of $lcp(x, y)$ on quasi-uniform distribution will provide a bound for the complexity of approximation of $C(y|x)$.

Recall the quasi-uniform distribution: To generate a random pair $(x, y) \in \{0, 1\}^n \times \{0, 1\}^n$ choose first a random number $i \in \{1, 2, \dots, n\}$, and then choose uniformly at random a pair (x, y) with $lcp(x, y) = i$. We need the following lemma: for every positive ε there is a positive δ such that every deterministic protocol that computes in k-rounds $lcp(x, y)$ on strings of length n with error probability δ (with respect to the quasi-uniform distribution on input pairs) exchanging at most l bits can be transformed into a deterministic protocol with the same communication length that computes lcp in $k - 1$-rounds on strings of length n/l with error probability ε. Fortunately, [6] proves first the average case version of Round elimination lemma in a form very close to what we need, and only then derives its version for randomised protocol.

Applying this lemma k times we can prove that α-approximating $lcp(x, y)$ in k rounds on many input pairs requires communication length $\Omega((n/\alpha)^{1/k})$. How this bound is related to approximating $C(y|x)$? It is because, relative to the quasi-uniform distribution, with high probability $C(y|x)$ is close to $lcp(x, y)$.

We first prove a lemma on average case k-round communication complexity of $lcp(x, y)$. To this end we need a version of Round elimination lemma from [6].

Assume that there we are given a communication problem, where Alice has a $x \in X$, Bob has a $y \in Y$ and Bob wants to find a $z \in Z$ such that the triple $\langle x, y, z\rangle$ is in a given relation T on $X \times Y \times Z$. Furthermore, assume that we are given a probability distribution ν over $X \times Y$ and they want to find a right z with high ν-probability.

For any natural m associate with T a new communication problem denoted by T_m: Alice has a tuple $\langle x_1, \dots, x_m\rangle \in X^m$, and Bob has a $y \in Y$ and the tuple $\langle x_1, \dots, x_{i-1}\rangle$ for some $i \le m$ (thus Bob knows a part of Alice's input). They want to find a $z \in Z$ such that the triple $\langle x_i, y, z\rangle$ is in T.

Consider the following probability distribution ν_m on inputs to T_m. We choose $i \le m$ at random, then choose independently m pairs $\langle x_1, y_1\rangle, \dots, \langle x_m, y_m\rangle$ with respect to distribution ν. Then we set $y = y_i$ (and throw away all other y_j's). Assume that the first message in communication protocol for T_m is sent by Alice. Intuitively, that message is not very helpful, as she does not know i, and thus there should be a deterministic protocol with less rounds and with the same communication length that computes relation T with small probability of error. The Round elimination lemma clarifies this intuition.

LEMMA 6. *Assume that there is a k-round deterministic protocol with communication length l that solves T_m with ν_m-probability of error at most ε'. Assume that the first message in that protocol is sent by Alice. If $\varepsilon' \le \frac{\varepsilon^2}{100 \ln(8/\varepsilon)}$ and $m \ge 20(l \ln 2 + \ln 5)/\varepsilon$ then there is a $k - 1$-round deterministic protocol with communication length l solving T with ν-probability of error at most ε.*

Actually this lemma is stated in [6] for functional relations only (for every x, y there is a unique z with $\langle x, y, z\rangle \in T$) and for worst case complexity of randomised protocols instead of average case complexity of deterministic protocols. However its proof from [6] works as well in our case.

LEMMA 7. *For every k there is a positive ε such that the following holds. For every n, α if there is a deterministic protocol α-approximating $lcp(x, y)$ after communicating l bits on at least $1 - \varepsilon$ fraction of pairs x, y according to the quasi-uniform distribution, then $l = \Omega((n/\alpha)^{1/k})$. (The constant in Ω-notation depends on k.)*

Proof. We prove the lemma by induction. The base of induction: for $k = 1$ let $\varepsilon' = \frac{\varepsilon^2}{100 \ln(8/\varepsilon)}$ where $\varepsilon = 1/2$. Assume that there is a 1-round communication protocol with probability of error at most ε' and communication length l α-approximating $lcp(x, y)$ for strings of length n. Let $m = 20(l \ln 2 + \ln 5)/\varepsilon$.

Let T be the problem of α-approximating $lcp(x, y)$ for strings of length n/m. Clearly, the problem T_m reduces to α-approximating $lcp(x, y)$ for strings of length n and the reduction preserves the quasi-uniform probability distribution. Thus there is a 1-round communication protocol with communication length l to solve T_m. By Round elimination lemma there is a 0-round protocol to α-approximate $lcp(x, y)$ for x, y of length n/m with probability of error 1/2. That is, with probability 1/2 Bob can approximate $lcp(x, y)$ without any knowledge of x. Obviously the largest probability of success for Bob is $\alpha/(n/m)$. Thus we have $\alpha/(n/m) \ge 1/2$, and $2m\alpha \ge n$. Recalling that $m = 20(l \ln 2 + \ln 5)/\varepsilon$ we obtain that $(al + b)\alpha \ge n$ for some constants a, b.

Induction step: let $\varepsilon' = \frac{\varepsilon^2}{100 \ln(8/\varepsilon)}$ where ε is the positive constant existing by induction hypothesis for k-round protocols. Assume that there is a $k + 1$-round communication protocol with probability of error at most ε' and communication length l to α-approximate $lcp(x, y)$ on strings of length n. Let $m = 20(l \ln 2 + \ln 5)/\varepsilon$. Then there is a $k+1$-round communication protocol with communication length l to solve T_m where T is the problem of α-approximating $lcp(x, y)$ for x, y of length n/m. By Round elimination lemma there is a k-round protocol to α-approximate $lcp(x, y)$ on strings of length n/m with probability of error ε. By induction hypothesis $(al + b)^k \alpha \geq n/m$ for some constants a, b. Recalling that $m = 20(l \ln 2 + \ln 5)/\varepsilon$ we see that $(a'l + b')^{k+1} \alpha \geq n$ for some other constants a', b'. □

LEMMA 8. *If $R_\varepsilon^{B,k\text{-round}}(C(y|x), \alpha, n) \leq l(n)$ then for any $0 < \delta < 1 - \varepsilon$ one can deterministically α'-approximate $lcp(x, y)$ by a k round protocol using $l(n)$ bits of communication on $1 - \varepsilon - \delta$ fraction of pairs (x, y) with respect to the quasi-uniform distribution, where $\alpha' = \alpha + \log \delta^{-1} + c$.*

Proof. Assume that $R_\varepsilon^{B,k\text{-round}}(C(y|x), \alpha, n) \leq l(n)$. Then there is a deterministic k-round protocol such that with probability of error at most ε over pairs (x, y) distributed quasi-uniformly Bob α-approximates $C(y|x)$ after at most $l(n)$ bits of communication.

Now we reduce approximating $lcp(x, y)$ to approximating Kolmogorov complexity $C(y|x)$. For some constant c we have $C(y|x) \leq s(x, y) + c$ for all x, y. On the other hand, if for fixed i and x the string y is chosen uniformly at random among strings with $s(x, y) = i$ then with probability at least $1 - 2^{-m}$ we have $C(y|x) \geq s(x, y) - m$. Set $2^{-m} = \delta$. Therefore if a deterministic protocol α-approximates $C(y|x)$ on $1 - \varepsilon$ fraction of pairs x, y it also α'-approximates $lcp(x, y)$ on $1 - \varepsilon - \delta$ fraction of pairs. □

The previous two lemmas prove that Theorem 4 holds for sufficiently small positive ε. To show that it holds for any $\varepsilon < 1/2$ we can use amplification. There is a small technical problem though: assume that we repeat the protocol $2m - 1$ times in parallel, using in each trial fresh random coins. We obtain $2m - 1$ contiguous segments $D_1, \dots, D_{2m-1}$ of length α such that with large probability more than half of them contain $lcp(x, y)$. How to approximate $lcp(x, y)$ given $D_1, \dots, D_{2m-1}$? Pick any m segments of $D_1, \dots, D_{2m-1}$ that have a common point. The union of those segments is a length-2α segment containing $lcp(x, y)$. (Indeed, assume that $lcp(x, y)$ does not belong to that union. Then there is a set of m segments containing $lcp(x, y)$ and each of those segments is different from any of the picked segments, since it contains $lcp(x, y)$.)

Remark. We could use Lemma 7 in the proof of Theorem 3 for $k = 2$ instead of Lemma 4 However in this way we would prove the statement for $\varepsilon < 1/2$ only. Besides, the overall proof would be more complicated and would yield a smaller constant in $\Omega(\sqrt{n})$.

4.3. Protocols using an a priori knowledge of $d(x, y)$

It turns out that some non-trivial upper bounds hold even for deterministic 1-round protocols to solve problem I. Let d be a normalised function. It appears that the main obstacle for our protocol to communicate exactly $d(x, y)$ bits in few rounds is our inability to efficiently and accurately estimate $d(x, y)$. If Alice and Bob would know in advance $d(x, y)$ and $d(y, x)$ they would need only to communicate about $2(d(x, y) + d(x, y))$ bits in a single round. This is exhibited by the protocol in the following theorem.

THEOREM 5. *For every normalised function d there is a deterministic protocol such that on inputs x, y, Alice learns y after communicating at most $2(d(x, y) + d(y, x)) + O(\log n)$ bits with Bob in a single round, provided that they are given $d(x, y)$ and $d(y, x)$ for free. There is a deterministic protocol such that on inputs x, y, Alice learns y after communicating of at most $4(n - I(x : y)) + O(\log n)$ bits with Bob in a single round, provided that they are given $I(x : y)$ for free. Here, $I(x : y) = C(y) - C(y|x)$ is the information in x about y.*

For a pair of Kolmogorov random strings with lots of information in common this theorem presents a substantial improvement over previous deterministic protocols.

Proof. Assume that Alice and Bob know $d(x, y)$ and $d(y, x)$. Consider the set

$$\begin{aligned} S &= S(y) \\ &= \{y' \mid \exists x' : \ d(x', y) = d(x, y), \ d(y', x') = d(y, x)\}. \end{aligned}$$

By normalisation property we have $|S| \leq 2^{d(x,y)+d(y,x)}$. The paper [2] shows that for every k, n there is a family $f_1, \dots, f_{2^{k+O(\log n)}}$ of functions mapping string of length n to strings of length $k + O(\log n)$ with the following property: For every 2^k-element set S of n-length binary strings and every element z in S there is j such that $f_j(z) \neq f_j(z')$ for all $z' \neq z$ in S.

Alice and Bob choose such a family (before starting the communication) for $k = d(x, y) + d(y, x)$. Then Bob finds a function f_j distinguishing his string y from all other strings in S and sends both j and $f_j(y)$ to Alice. Alice finds a string y' with $d(y', x) = d(y, x)$ and $f_j(y') = f_j(y)$ and outputs it. As y' belongs to S, Alice cannot err.

Assume that Alice and Bob know $I(x : y)$. Alice finds $C(x)$ and sends it to Bob. Then Bob finds $C(y)$ and sends it to Alice. Thus they both know $C(y|x) = C(y) - I(x : y)$. By the Symmetry of information (see [5]), we have $I(x : y) = I(y : x) + O(\log n)$. Therefore they both can find $C(x|y)$ with $O(\log n)$ accuracy and apply the previous protocol. The communication length of this protocol is

$$\begin{aligned} &2(C(y|x) + C(x|y)) + O(\log n) \\ &= 2(C(x) + C(y) - 2I(x:y)) + O(\log n) \end{aligned}$$

but it runs in 2 rounds. If, instead of the precise value of $C(x)$ they use upper bound $n + O(1)$ for it, we obtain a protocol sending at most

$$\begin{aligned} 2(n + C(y) - 2I(x:y)) + O(\log n) \\ \leq 4(n - I(x:y)) + O(\log n) \end{aligned}$$

bits in one round. □

References

[1] N. Alon and A. Orlitsky. Repeated communication and Ramsey graphs. *IEEE Transactions on Information Theory*, 41:5 (September 1995), pp. 1276-1289.

[2] H. Buhrman, L. Fortnow, and S. Laplante. Resource bounded Kolmogorov complexity revisited. *SIAM Journal on Computing*, 31(3):887-905, 2005.

[3] H. Buhrman, H. Klauck, N. K. Vereshchagin, P. M. B. Vitányi. Individual Communication Complexity. In *Proceedings of 21st Annual Symposium on Theoretical Aspects of Computer Science, Montpellier, France, March 25-27, 2004.* Lecture Notes in Computer Science 2996. Springer 2004. P. 19–30.

[4] E. Kushilevitz, N. Nisan. *Communication Complexity.* Cambridge UP, 1997.

[5] M. Li and P. M. B. Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer-Verlag, 1997. 2nd Edition.

[6] P. Bro Miltersen, N. Nisan, S. Safra, and A. Wigderson. On data Structures and Asymmetric Communication Complexity. *Journal of Computer and Systems Sciences*, 57 (1998) 37–49.

[7] An. Muchnik, Conditional complexity and codes, *Theoretical Computer Science*, v. 271, no. 1–2, p. 97–109 (2002).

[8] M. Naor, A. Orlitsky and P. Shor. Three results on interactive communication. *IEEE Transactions on Information Theory*, 39:5 (September 1993), pp. 1608–1615.

[9] N. Nisan. The communication complexity of threshold gates. *Combinatorics, Paul Erdös is eighty (Vol. 1)*, D. Miklós, V.T. Sós, and T. Szönyi, Eds., Janos Bolyai Math. Society: Budapest, Hungary, 1993, 310–315.

[10] A. Orlitsky. Worst-case interactive communication I: Two messages are almost optimal. *IEEE Transactions on Information Theory*, 36:5 (September 1990), pp. 1111–1126.

[11] A. Orlitsky. Worst-case interactive communication II: Two messages are not optimal. *IEEE Transactions on Information Theory*, 37:4 (July 1991), pp. 995–1005.

[12] A. Orlitsky. Average-case interactive communication. *IEEE Transactions on Information Theory*, 38:4 (July 1992), pp. 1534–1547.

[13] D. Slepian and J. K. Wolf. Noiseless Coding of Correlated Information Sources. *IEEE Transactions on Information Theory*, vol. IT-19 (July 1973), pp. 471–480.