Symmetry of information and bounds on nonuniform randomness extraction via Kolmogorov extractors

Marius Zimand *

Abstract

We prove a strong Symmetry of Information relation for random strings (in the sense of Kolmogorov complexity) and establish tight bounds on the amount on nonuniformity that is necessary for extracting a string with randomness rate 1 from a single source of randomness. More precisely, as instantiations of more general results, we show:

- For all n-bit random strings x and y, x is random conditioned by y if and only if y is random conditioned by x;
- While $O(1)$ amount of advice regarding the source is not enough for extracting a string with randomness rate 1 from a source string with constant random rate, $\omega(1)$ amount of advice is.

The proofs use Kolmogorov extractors as the main technical device.

Keywords: Symmetry of information, random strings, randomness extraction, Kolmogorov extractors.

1 Introduction

Kolmogorov extractors are procedures that increase the Kolmogorov complexity rate of strings and sequences. Their explicit study was initiated by Fortnow, Hitchcock, A. Pavan, Vinodchandran and Wang [FHP$^+$06] for the case of finite strings and by Reimann [Rei04] for the case of infinite strings. The recent paper [Zim10] is a survey of this field.

In this paper, we use Kolmogorov extractors as a conceptual and technical device to derive results in two directions that otherwise appear unrelated.

First, we study the symmetry of information phenomenon for random strings. We show that a random string x has essentially no information about a random string y if and only if y has essentially no amount of information about x. By "essentially no amount of information," we mean that the amount of information is bounded by an absolute constant. Up to this paper, it was only known that if x has only a constant amount information about y, then y has $O(\log n)$ information about x. Thus we replace the $O(\log n)$ term by $O(1)$.

Secondly, we investigate the amount of non-uniformity that is necessary for randomness extraction from one source. It is well-known that randomness extraction from a single source is not possible (if we exclude some trivial cases). In fact, as a consequence of a result of Vereshchagin and Vyugin [VV02], we note that obtaining a source with randomness rate 1 from a source with randomness rate, say, 0.99 is not possible even if the extractor has access to a constant amount of non-uniform information. In contrast, we show that an $\omega(1)$ amount of non-uniform information is sufficient for this task.

We continue with a more detailed discussion of the two types of results and of the technical method that we use.

*Department of Computer and Information Sciences, Towson University, Baltimore, MD.; email: mzimand@towson.edu; http://triton.towson.edu/∼mzimand. The author is supported in part by NSF grant CCF 1016158.

Symmetry of Information. Strictly speaking, a string x of length n is said to be random if its Kolmogorov complexity is at least n (i.e., $C(x \mid n) \geq n$) and is said to be random conditioned by string y if its Kolmogorov complexity conditioned by y is at least n (i.e., $C(x \mid y) \geq n$). Since the Kolmogorov complexity depends up to an additive constant on the choice of the universal machine, a relaxed definition is more robust. Such a definition is obtained if we require that $C(x \mid n)$ (respectively, $C(x \mid y)$) are within a constant from n. Formally, for a constant c, we say that x is c-random if $C(x \mid n) \geq n - c$, and x is c-random conditioned by y if $C(x \mid y) \geq n - c$.

We prove a strong symmetry-of-information relation for random strings: for any n-bit random strings x and y, x is random conditioned by y iff y is random conditioned by x. More exactly, we show the following.

Fact 1 *(Informal statement; see Theorem 3.4 for full statement.) Let x and y be two n-bit strings that are c-random and such that x is c-random conditioned by y. Then y is c'-random conditioned by x, where c' is a constant that only depends on c.*

This result is a consequence of a new form of the Symmetry of Information Theorem, which is tighter than the classical theorem of Kolmogorov and Levin [ZL70] for some types of strings, including the class of random strings. This is an important class because most strings are random and because counting arguments based on Kolmogorov complexity typically use random strings. Symmetry of information is one of the basic principles in information theory. It states that for any two random variables X and Y, the information in X about Y is *equal* to the information in Y about X, i.e., $I(X : Y) = I(Y : X)$, where $I(X : Y) = H(Y) - H(Y \mid X)$ and $H()$ is the Shannon entropy. The principle also holds in algorithmical information theory, provided we replace *equal* by *approximately equal*. More formally, for two binary strings x and y, we define $I(x : y) = C(y) - C(y \mid x)$. Then, the Symmetry of Information Theorem of Kolmogorov and Levin states that $|I(x : y) - I(y : x)| = O(\log |x| + \log |y|)$. For the general case of arbitrary strings, the logarithmical term cannot be avoided because there are n-bit strings x and y such that $|I(x : y) - I(y : x)| > \log n - O(1)$ (for example, such strings can be obtained by a simple modification of Example 2.27 in [LV93]).

Symmetry of information follows immediately from another basic result, the *Chain Rule*: $C(xy) \approx C(y) + C(x \mid y)$. While it is easy to see that $C(xy) \leq C(y) + C(x \mid y) + C^{(2)}(y)$ [1], the more difficult direction in the proof of the chain rule shows that

$$C(xy) \geq C(y) + C(x \mid y) - C^{(2)}(xy).$$

We prove a new form of the Chain Rule (which implies the alluded new form of the Symmetry of Information Theorem; here and in the rest of this paper, $I(x : y) = C(y \mid n) - C(y \mid x)$.)

Fact 2 *(Informal statement; see Theorem 3.1 for full statement) Let x and y be n-bit strings such that $C(x \mid y) = \Omega(\log n)$ and $C(y \mid x) = \Omega(\log n)$. Then*

$$\begin{aligned} C(xy \mid n) \quad &\geq C(x \mid n) + C(y \mid x) - \log I(x : y) \\ &\quad -O(C^{(2)}(x \mid n) + C^{(2)}(y \mid n) + C^{(2)}(y \mid x)). \end{aligned}$$

Note that if x is random and y is random even conditioned by x, then $C^{(2)}(x \mid n) = O(1)$, $C^{(2)}(y \mid n) = O(1)$, $C^{(2)}(y \mid x) = O(1)$, and $I(x : y) = O(1)$. In this case we obtain $C(xy \mid n) \geq C(x \mid n) + C(y \mid x) - O(1) = 2n - O(1)$. Fact 1 follows easily from this relation.

Randomness extraction with small advice. By and large, randomness extraction is an algorithmical process that constructs a source of randomness of high quality from one or several sources of

[1] We use the following shortcut notations: $C^{(2)}(x)$ is $C(C(x))$, $C^{(2)}(x \mid y)$ is $C(C(x \mid y))$, $C^{(2)}(x \mid n)$ is $C(C(x \mid n) \mid n)$.

lower quality. If we restrict to the case when there is only one input source, one wants to design an effective transformation E from the set of n-bit strings to the set of m-bit strings such that for any source x "with randomness $\leq k$", $E(x)$ has randomness $\approx m$. It is desirable to have $m \approx k$ (i.e., to extract all, or almost all, of the randomness in the source). The problem of randomness extraction has been modeled in two ways. In the first model, a source is a probability distribution X over $\{0,1\}^n$ and its randomness is given by the min-entropy $H_\infty(X)$. In the second model, a source is a string $x \in \{0,1\}^n$ and its randomness is given by its Kolmogorov complexity $C(x)$. For our study, the second model is more convenient; moreover all the results can be translated in the first model. (For the relation between the two models, see [FHP$^+$06], [HPV09] and also the survey paper [Zim10]).

Formally (in the second model), given the parameter $k \leq n$, one would like to have a function $E : \{0,1\}^n \to \{0,1\}^m$ such that $C(E(x)) \approx m$ whenever $C(x) \geq k$. It is well known that no such computable function E exists for non-trivial parameters. Indeed for any given E, consider the string $y \in \{0,1\}^m$ with the largest number of preimages. Then $C(y \mid n) = O(1)$ and among its at least 2^{n-m} preimages there must be some x with $C(x) \geq n - m$. In other words, for any given E, there are some strings (such as the above x) on which E fails. Thus, in order for a function E to extract randomness from any source x with randomness $\geq k$, E must have some additional information α_x, which we call *advice about the source.* The question is how much such advice information should be provided.

Fortnow et al. [FHP$^+$06] have shown that a constant number of advice bits are sufficient if one settles to extracting from strings with linear randomness a string whose randomness rate is $1-\epsilon$. [2] More precisely, they show that for any positive rational numbers σ and ϵ, there exists a polynomial-time computable function E and a constant h such that for any $x \in \{0,1\}^n$ with $C(x) \geq \sigma n$, there exists a string α_x of length h such that $C(E(x, \alpha_x)) \geq (1-\epsilon)m$ and $m \geq cn$, for some constant c that depends on σ and ϵ. Note that this result implies that it is possible to construct in polynomial time a list with 2^h strings and one of them is guaranteed to have Kolmogorov complexity at least $(1-\epsilon)m$.

The shortcoming of Fortnow et al's result is that the randomness rate of the output is not 1. It would be desirable that $C(E(x, \alpha_x)) \geq m - o(m)$. We first remark that as a consequence of a result of Vereshchagin and Vyugin [VV02], randomness rate 1 cannot be obtained with a constant number of bits of advice about the input. Indeed, we show that if a computable function $E : \{0,1\}^n \times \{0,1\}^h \to \{0,1\}^m$ has the property that for all strings x with $C(x) \geq \sigma n$, it holds that there exists α_x such that $C(E(x, \alpha_x)) \geq (1-\epsilon)m$, then $\epsilon \geq \frac{1-\sigma}{2^{h+1}-1} - o(1)$ (provided that $m = \omega(\log n + h)$).

In contrast with the above impossibility result, we show that from sources with a linear amount of randomness, one can extract a string with randomness rate 1 with basically any non-constant amount of advice, such as, for example, the inverse of the Ackerman function. This is an instantiation of the following more general result.

Fact 3 *(Informal statement; see Theorem 4.6 for full statement.) For any m computable from n there exists a computable function E with the following property:*

For every n-bit string x with complexity $\geq m$, there exists a string α_x of length $\omega(\log \frac{n}{m})$ such that $C(E(x, \alpha_x)) = m - o(m)$ and the length of $C(E(x, \alpha_x))$ is m.

Note that the function E from Fact 3 is computable, but no complexity bound is claimed for it. We can obtain an extractor E computable by a polynomial-size circuit with almost all the properties from Theorem 3. Basically the weakening is that the output length m has to be $\leq cn$, for some positive constant c. Moreover the polynomial-size circuit is itself computable, in the sense that there exists an algorithm that on input n outputs the description of the circuit that computes E. We call such a circuit an *effectively constructible circuit.*

[2] The randomness rate of an n-bit string x is $C(x)/n$.

Fact 4 *(Informal statement; see Theorem 4.8 for full statement.) There exists a constant c such that for any m computable in polynomial time from n and $m \leq cn$ there exists a function E, which is computable by a polynomial-size effectively constructible circuit with the following property:*

For every n-bit string x with complexity $\geq m$, there exists a string α_x of length $\omega(\log \frac{n}{m})$ such that $C(E(x, \alpha_x)) = m - o(m)$ and the length of $E(x, \alpha_x)$ is m.

Discussion of technical aspects. We present the main ideas in the proofs of Fact 2, Fact 3, and Fact 4. As mentioned, the proofs rely on Kolmogorov extractors. A Kolmogorov extractor is a computable ensemble of functions $E : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \to \{0,1\}^m$ such that for all $x \in \{0,1\}^{n_1}$ and $y \in \{0,1\}^{n_2}$ that have Kolmogorov complexity above a certain threshold value and that are sufficiently independent (which roughly means that $C(y \mid x) \approx C(y)$), it holds that $C(E(x, y)) \approx m$.

The general idea for the Chain Rule in Fact 2 is to construct a Kolmogorov extractor $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$ that extracts almost all the randomness from its inputs, i.e., $C(E(x, y) \mid n) \geq C(x \mid n) + C(y \mid x) - (\text{small term})$. If E would be a computable function, then $C(E(x, y) \mid n) \leq C(xy \mid n) + O(1)$ (*), and we would obtain the difficult direction in the Chain Rule, $C(xy \mid n) \geq C(x \mid n) + C(y \mid x)$, modulo the small additive term. However, in order to have that term small enough for the desired level of tightness in the Chain Rule, the randomness extraction needs to be finely tuned (this is the most technical part of the proof) depending on some attributes of x and y. Thus, E will not be fully computable, as required by the standard definition of a Kolmogorov extractor. Instead, E needs a few bits of information about its inputs, and even if this weakens the inequality (*), it still allows us to achieve the desired level of tightness in the Chain Rule.

To obtain the extractors E that require a small amount of non-uniform information about the source (see Fact 3 and Fact 4), the use of Kolmogorov extractors is quite direct. The goal here is to show that for each x, it is enough to have a short string α_x such that $E(x, \alpha_x)$ has randomness rate 1 and contains almost all the randomness of x. The solution is based on the fact that from x and a short string that is random even conditioned by x, one can extract almost all the randomness of x. This is similar to the well-studied case of seeded extractors, with the remark that we can have shorter seeds because requiring that the output has randomness rate equal to 1 is weaker than requiring that the output is statistically close to the uniform distribution (as stipulated in the definition of seeded extractors). Then we take α_x to be such a short seed. The above fact is obtained via an elementary use of the probabilistic method. We first identify a combinatorial object, called a *balanced table*, that characterizes a Kolmogorov extractor, in the sense that the table of a Kolmogorov extractor must satisfy the combinatorial constraints of a balanced table. We show (with the probabilistic method) that such an object exists with a seed of length $\omega(\log(n/m))$, where n is the length of x and m is the Kolmogorov complexity of x. This establishes Fact 3. Since the function E from Fact 3 is obtained via the probabilistic method, we cannot claim any complexity bound for it. To obtain the Kolmogorov extractor in Fact 4, which is computed by polynomial-size circuits, we derandomize the construction from Fact 3, using a method of Musatov [Mus10]. The key observation is that the combinatorial constraints of a balanced table can be checked by constant-depth circuits of relatively small size. The argument goes as follows: (a) these constraints require that in all sufficiently large rectangles of the table no element appears too many times; (b) thus one needs to count the occurrence of each element in every sufficiently large rectangle of the table; (c) by a well-known result of Ajtai [Ajt93], this operation can be done with sufficient accuracy by constant-depth circuits with relatively small size. Therefore, we can use the Nisan-Wigderson ([NW94]) pseudo-random generator NW-gen that fools bounded-size constant-depth circuits and has seeds of size polylogaritmic in the size of the output. Since balanced tables with the required parameters are abundant, we infer that there exists a seed s so that NW-gen(s) is a balanced table with the required parameters. A balanced table is an object of size exponential in n, which implies that the seed s has size polynomial in n. Moreover, the Nisan-Wigderson pseudo-random generator has the property that each bit of the output can be calculated separately in time polynomial in the length of the seed. This implies that the Kolmogorov extractor whose

table is NW-gen(s) can be computed by a polynomial-sized circuit that has s hard-wired in its circuitry.

2 Preliminaries

2.1 Notation and basic facts on Kolmogorov complexity

The Kolmogorov complexity of a string x is the length of the shortest effective description of x. There are several versions of this notion. We use here the *plain complexity*, denoted $C(x)$, and also the *conditional plain complexity* of a string x given a string y, denoted $C(x \mid y)$, which is the length of the shortest effective description of x given y. The formal definitions are as follows. We work over the binary alphabet $\{0,1\}$. A string is an element of $\{0,1\}^*$. If x is a string, $|x|$ denotes its length. Let M be a Turing machine that takes two input strings and outputs one string. For any strings x and y, define the *Kolmogorov complexity* of x conditioned by y with respect to M, as $C_M(x \mid y) = \min\{|p| \mid M(p,y) = x\}$. There is a universal Turing machine U with the following property: For every machine M there is a constant c_M such that for all x, $C_U(x \mid y) \leq C_M(x \mid y) + c_M$. We fix such a universal machine U and dropping the subscript, we write $C(x \mid y)$ instead of $C_U(x \mid y)$. We also write $C(x)$ instead of $C(x \mid \lambda)$ (where λ is the empty string). The *randomness rate* of a string x is defined as $\mathrm{rate}(x) = \frac{C(x)}{|x|}$. If n is a natural number, $C(n)$ denotes the Kolmogorov complexity of the binary representation of n. For two n-bit strings x and y, the information in x about y is denoted $I(x:y)$ and is defined as $I(x:y) = C(y \mid n) - C(y \mid x)$.

In this paper, the constant hidden in the $O(\cdot)$ notation only depends on the universal Turing machine.

For all n and $k \leq n$,
$2^{k-O(1)} < |\{x \in \{0,1\}^n \mid C(x \mid\ n) < k\}| < 2^k$.

Strings $x_1, x_2, \ldots, x_k$ can be encoded in a self-delimiting way (i.e., an encoding from which each string can be retrieved) using $|x_1| + |x_2| + \ldots + |x_k| + 2\log|x_2| + \ldots + 2\log|x_k| + O(k)$ bits. For example, x_1 and x_2 can be encoded as $\overline{(bin(|x_2|)}01x_1x_2$, where $bin(n)$ is the binary encoding of the natural number n and, for a string $u = u_1 \ldots u_m$, $\overline{u}$ is the string $u_1u_1 \ldots u_mu_m$ (i.e., the string u with its bits doubled).

All the Kolmogorov extractors in this paper are ensembles of functions $f = (f_n)_{n\in\mathbb{N}}$ of type $f_n : \{0,1\}^n \times \{0,1\}^{k(n)} \to \{0,1\}^{m(n)}$. For readability, we usually drop the subscript and the expression "ensemble $f : \{0,1\}^n \times \{0,1\}^k \to \{0,1\}^m$" is a substitute for "ensemble $f = (f_n)_{n\in\mathbb{N}}$, where for every n, $f_n : \{0,1\}^n \times \{0,1\}^{k(n)} \to \{0,1\}^{m(n)}$."

For any $n \in \mathbb{N}$, $[n]$ denotes the set $\{1, 2, \ldots, n\}$.

2.2 Approximate counting via polynomial-size constant-depth circuits

In the derandomization argument used in the proof of Theorem 4.8, we need to count with constant-depth polynomial-size circuits. Ajtai [Ajt93] has shown that this can be done with sufficient precision.

Theorem 2.1 *(Ajtai's approximate counting with polynomial size constant-depth circuits.) There exists a uniform family of circuits $\{G_n\}_{n\in\mathbb{N}}$, of polynomial size and constant depth, such that for every n, for every $x \in \{0,1\}^n$, for every $a \in \{0, \ldots, n-1\}$, and for every $\epsilon > 0$,*

- *If the number of 1's in x is $\leq (1-\epsilon)a$, then $G_n(x, a, 1/\epsilon) = 1$,*
- *If the number of 1's in x is $\geq (1+\epsilon)a$, then $G_n(x, a, 1/\epsilon) = 0$.*

We do not need the full strength (namely, the uniformity of G_n) of this theorem; the required level of accuracy (just $\epsilon > 0$) can be achieved by non-uniform polynomial-size circuits of depth $d = 3$ (with a much easier proof, see [Vio10]).

2.3 Pseudo-random generator fooling bounded-size constant-depth circuits

The derandomization in the proof of Theorem 4.8 is done using the Nisan-Wigderson pseudo-random generator that "fools" constant-depth circuits [NW94]. Typically, it is required that the circuit to be fooled has polynomial size, but the proof works for circuits of size 2^{n^α} for some small constant $\alpha > 0$.

Theorem 2.2 *(Nisan-Wigderson pseudo random generator.) For every constant d there exists a constant $\alpha > 0$ with the following property. There exists a function NW-gen$: \{0,1\}^{O(\log^{2d+6} n)} \to \{0,1\}^n$ such that for any circuit G of size 2^{n^α} and depth d,*

$$|\mathrm{Prob}_{s\in\{0,1\}^{O(\log^{2d+6} n)}}[G(\textit{NW-gen}(s)) = 1] - \mathrm{Prob}_{z\in\{0,1\}^n}[G(z) = 1]| < 1/100.$$

Moreover, there is a procedure that on inputs (n, i, s) produces the i-th bit of NW-gen(s) in time $\mathrm{poly}(\log n)$.

3 Symmetry of information for random strings

We start by proving the new form of the Chain Rule in Fact 2. The formal statement is as follows:

Theorem 3.1 *For all strings $x \in \{0,1\}^n$ and $y \in \{0,1\}^n$ with $C(x \mid n) \geq 13\log n + I(x : y) + O(1)$ and $C(y \mid n) \geq 7\log n + I(x : y) + O(1)$,*

$$\begin{aligned} C(xy \mid n) \quad &\geq C(x \mid n) + C(y \mid x) - \log I(x : y) \\ &\quad -O(C^{(2)}(x \mid n) + C^{(2)}(y \mid n) + C^{(2)}(y \mid x)). \end{aligned}$$

Proof. Let x and y be as in the statement of the theorem. We introduce some notation:

- $t_x = C(x \mid n)$,
- $t_y = C(y \mid n)$,
- $t_{y,x} = C(y \mid x)$,
- $d = 2(C(t_x \mid n) + C(t_y \mid n) + C(t_{y,x} \mid n)) + I(x : \ y) + O(1)$,
  (note that $d = O(C^{(2)}(x \mid n) + C^{(2)}(y \mid n) + C^{(2)}(y \mid \ x)) + I(x : y)$),
- $k_x = t_x - 2(C(t_x \mid n) + C(t_y \mid n) + C(t_{y,x} \mid n)) - O(1)$,
- $k_y = t_y - O(1)$,
- $m = k_x + k_y - \log d - O(1)$,

where the constant $O(1)$ only depends on the universal machine and will be chosen later. We also denote $K_x = 2^{k_x}$, $K_y = 2^{k_y}$, $N = 2^n$, $D = 2^d$ and $M = 2^m$. We state two claims, which we prove later, that immediately establish the theorem. The first claim shows the existence of a function $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$ that satisfies certain combinatorial constraints similar to those of a balanced table, whose parameters are tailored for what we need. We view E as an $[N]$-by-$[N]$ table colored with colors in $[M]$. A rectangle $B_1 \times B_2$, where $B_1 \subseteq [N]$ and $B_2 \subseteq [N]$, is the part of the table formed by the rows in B_1 and the columns in B_2. For $A \subseteq [M]$, we say that a cell (u, v) of the table is an A-cell, if $E(u, v) \in A$.

Claim 3.2 *There exists $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$ such that for every rectangle $B_1 \times B_2$, with $|B_1| \geq K_x$, $|B_2| \geq K_y$, and for every $A \subseteq \{0,1\}^m$ with $|A| \geq \frac{M}{D}$, the number of A-cells in $B_1 \times B_2$ is $\leq 2 \cdot \frac{|A|}{M} \cdot |B_1| \cdot |B_2|$.*

Note that given $n, t_x, t_y, t_{y,x}$, $C(t_x \mid n), C(t_y \mid n)$, $C(t_{y,x} \mid n)$, one can effectively enumerate the tables that satisfy the claim. Let E be the first table that appears in this enumeration. We denote the tuple $(t_x, t_y, t_{y,x}, C(t_x \mid n), C(t_y \mid n), C(t_{y,x} \mid n))$ by Λ. Given n, the tuple Λ can be encoded in a self-delimiting way using $C(t_x \mid n) + C(t_y \mid n) + C(t_{y,x} \mid n) + C(C(t_x \mid n) + C(C(t_y \mid n)) + C(C(t_{y,x} \mid n)) + 2\log C(t_x \mid n) + 2\log C(t_y \mid n) + 2\log C(t_{y,x} \mid n) + 2\log C(C(t_x \mid n) + 2\log C(C(t_y \mid n)) + 2\log C(C(t_{y,x} \mid n)) + O(1)$ bits, a value which we denote by λ. Using $C(C(t_x \mid n)) \leq \log C(t_x \mid n)$ and the other similar inequalities, we see that for an appropriate choice of the constants, $d > \lambda + I(x : y)$.

The second claim shows that E extracts almost all the randomness in x and y.

Claim 3.3 *$C(E(x, y) \mid n, m) \geq m - d$.*

The theorem follows easily from the two claims. Note that the calculation of $E(x, y)$ requires a description of x, y and λ bits for the self-delimited description of E. Then

$$\begin{aligned}
C(xy \mid n) &\geq C(E(x,y) \mid n, m) - \lambda - O(1) \\
&\geq m - d - \lambda - O(1) \\
&= k_x + k_y - \log d - d - \lambda - O(1) \\
&= t_x + t_y - I(x : y) \\
&\quad -6(C^{(2)}(x \mid n) + C^{(2)}(y \mid n) + C^{(2)}(y \mid x)) \\
&\quad - \log(2(C^{(2)}(x \mid n) + C^{(2)}(y \mid n) + C^{(2)}(y \mid x)) + I(x : y)) - O(1) \\
&\geq t_x + t_y - I(x : y) - \log I(x : y) \\
&\quad -O(C^{(2)}(x \mid n) + C^{(2)}(y \mid n) + C^{(2)}(y \mid x)),
\end{aligned}$$

where in the last line we have used the fact that

$$\begin{aligned}
&\log(2(C^{(2)}(x \mid n) + C^{(2)}(y \mid n) + C^{(2)}(y \mid x)) + I(x : y)) \\
&\leq \log(2(C^{(2)}(x \mid n) + C^{(2)}(y \mid n) + C^{(2)}(y \mid x))) + \log I(x : y).
\end{aligned}$$

Since $t_x = C(x \mid n)$ and $t_y - I(x : y) = C(y \mid n) - (C(y \mid n) - C(y \mid x)) = C(y \mid x)$, the conclusion follows.

It remains to prove the two claims.

Proof of Claim 3.2. We use the probabilistic method. The details are presented in the Appendix.

Proof of Claim 3.3. Suppose that $C(E(x, y) \mid n, m) < m - d$. Let $A = \{w \in \{0,1\}^m \mid C(w \mid n, m) \leq m - d + O(1)\}$, where the constant $O(1)$ (depending only on the universal machine) is chosen so that $|A| \geq 2^{m-d} = \frac{M}{D}$. It also holds that $|A| \leq 2^{m-d+O(1)}$.

We define $B_y = \{v \in \{0,1\}^n \mid C(v \mid n) \leq t_y\}$. For a convenient choice of the constant appearing in the definition of k_y, it holds that $|B_y| \geq K_y$. Also note that $|B_y| \leq 2^{t_y+1}$.

We say that a row $u \in \{0,1\}^n$ is *bad*, if the number of A-cells in the $\{u\} \times B_y$ rectangle of E is $> 2 \cdot \frac{|A|}{M} \cdot |B_y|$.

The number of bad rows is bounded by K_x (otherwise, E would not satisfy the requirement in Claim 3.2 for the rectangle formed by the bad rows and B_y). Note that, given $n, t_x, t_y, t_{y,x}, C(t_x \mid n), C(t_y \mid n), C(t_{y,x} \mid n)$, one can enumerate the set of bad rows. Therefore a bad row u can be described by its rank in an enumeration of the set of bad rows, and by the information Λ required to run this enumeration. Therefore if u is a bad row, then

$$\begin{aligned}
C(u \mid n) &\leq k_x + \lambda \\
&< t_x.
\end{aligned}$$

Since $C(x \mid n) = t_x$, x is a good row. Therefore the number of A-cells in $\{x\} \times B_y$ is

$$\begin{aligned}
&< 2 \cdot \tfrac{|A|}{M} \cdot |B_y| \\
&< 2 \cdot \tfrac{2^{m-d+O(1)}}{2^m} \cdot 2^{t_y+1} \\
&= 2^{t_y - d + O(1)}.
\end{aligned}$$

By our assumption, (x, y) is an A-cell, and, obviously, it is in the $\{x\} \times B_y$ rectangle. Given x, y can be described by the rank of (x, y) in an enumeration of A-cells in $\{x\} \times B_y$ and by the information Λ required to run this enumeration.

Thus,

$$\begin{aligned} C(y \mid x) &\leq t_y - d + O(1) + \lambda \\ &< t_y - I(x : y) \\ &= C(y \mid n) - (C(y \mid n) - C(y \mid x)) \\ &= C(y \mid x), \end{aligned}$$

which is a contradiction. ∎

From Theorem 3.1, it is easy to derive the strong Symmetry of Information relation for random strings stated in Fact 1. The formal statement is as follows.

Theorem 3.4 *For every constant $c \geq 0$, there exists a constant $c' \geq 0$ with the following property: For every $n \in \mathbb{N}$, for every c-random string $x \in \{0,1\}^n$ and every c-random string $y \in \{0,1\}^n$, if y is c-random conditioned by x, then x is c'-random conditioned by y.*

Proof. Note that if $C(x \mid n) \geq n - c$ and $C(y \mid x) \geq n - c$, then $C^{(2)}(x \mid n) = O(1)$, $C^{(2)}(y \mid n) = O(1)$ and $C^{(2)}(x \mid n) = O(1)$. Then $I(x : y) = O(1)$.

Therefore, from Theorem 3.1, we obtain $C(xy \mid n) \geq 2n - O(1)$. We also have $C(xy \mid n) \leq C(y \mid n) + C(x \mid y) + 2C(C(y \mid n)) + O(1) = C(y \mid n) + C(x \mid y) + O(1)$. Thus, $C(x \mid y) \geq 2n - C(y \mid n) - O(1) = n - O(1)$. ∎

4 Randomness extraction with small advice

In this section we study the amount of non-uniformity that is necessary for randomness extraction from a single source of randomness.

Vereshchagin and Vyugin [VV02] show the limitations of what can be extracted with a bounded quantity of advice. To state their result, let us fix n = length of the source, h = number of bits of advice that is allowed, and m = the number of extracted bits. Let $H = 2^{h+1} - 1$.

Theorem 4.1 ([VV02]) *There exists a string $x \in \{0,1\}^n$ with $C(x) > n - H \log(2^m + 1) \approx n - Hm$ such that any string $z \in \{0,1\}^m$ with $C(z \mid x) \leq h$ has complexity $C(z) < h + \log n + \log m + O(\log \log n, \log \log m)$.*

The next theorem, a consequence of Theorem 4.1, shows that no Kolmogorov extractor for sources with randomness rate σ and that uses h bits of advice about the source can output strings with randomness rate larger than $1 - (1 - \sigma)/H$.

Theorem 4.2 *Assume that the parameters m, h, σ are computable from n and satisfy the following relations: $0 < \sigma < 1, h > 0, 0 < m < n$, $m = \omega(\log n + h)$.*

Let $f : \{0,1\}^n \times \{0,1\}^h \to \{0,1\}^m$ be a computable ensemble of functions such that for every $x \in \{0,1\}^n$ with $C(x) \geq \sigma \cdot n$, there exists a string α_x such that $C(f(x, \alpha_x)) \geq (1-\epsilon) \cdot m$. Then $\epsilon \geq \frac{1-\sigma}{H} - o(1)$.

Proof. Let $m' = \min(\lfloor \frac{1-\sigma}{H} \cdot n \rfloor, m)$. Note that $m' \geq \lfloor \frac{1-\sigma}{H} \cdot m \rfloor$. Let x be the string guaranteed by the Vereshchagin-Vyugin Theorem 4.1 for the parameters $n, h + c, m'$, where c is a constant that will be specified later. Note that $C(x) > n - H \cdot m' \geq \sigma \cdot n$. By assumption there is a string α_x such that $C(f(x, \alpha_x)) \geq (1 - \epsilon)m$. Let z be the prefix of length m' of $f(x, \alpha_x)$. Note that $C(f(x, \alpha_x)) \leq C(z) + (m - m') + 2 \log m + O(1)$, which implies that $C(z) \geq (1 - \epsilon)m - m + m' - 2 \log m - O(1) \geq \frac{(1-\sigma)m}{H} - \epsilon m - 2 \log m - O(1)$.

We also have $C(z \mid x) \leq |\alpha_x| + c = h + c$, for some constant c. It follows from Theorem 4.1 that $C(z) < h + \log n + \log m' + O(\log\log n, \log\log m')$. So, $\frac{(1-\sigma)}{H}m - \epsilon m - 2\log m - O(1) \leq h + \log n + O(\log\log n, \log\log m')$, which implies that $\epsilon \geq \frac{1-\sigma}{H} - \frac{h+O(\log n)}{m} = \frac{1-\sigma}{H} - o(1)$. ∎

We move to showing the positive results in Fact 3 and Fact 4 regarding randomness extraction with small advice that complement the negative result in Theorem 4.2. The constructions use the parameters n, n_1, m, k, δ and d. We denote $N = 2^n, N_1 = 2^{n_1}, M = 2^m, \Delta = 2^\delta$ and $D = 2^d$. We identify in the natural way a function $E : \{0,1\}^n \times \{0,1\}^{n_1} \to \{0,1\}^m$ with an $[N] \times [N_1]$ table colored with colors from $[M]$. For $A \subseteq [M]$, we say that an (u, v) cell of the table is an A-cell if $E(u, v) \in A$. The reader might find helpful to consult the proof plan presented in the Introduction. As explained there the notion of a *balanced table* plays an important role.

Definition 4.3 *A table $E : [N] \times [N_1] \to [M]$ is (K, D, Δ)-balanced if for any $B \subseteq [N]$ with $|B| \geq K$, for any $A \subseteq [M]$ with $\frac{|A|}{M} \geq \frac{1}{D}$, it holds that*

$$\frac{|A\text{-cells in } B \times [N_1]|}{|B| \times N_1} \leq \Delta \cdot \frac{|A|}{M}.$$

The following lemma shows that a balanced table is a good Kolmogorov extractor.

Lemma 4.4 *Let $E : [N] \times [N_1] \to [M]$ be a (K, D, Δ)-balanced table and $d = \delta + O(1)$. Suppose $C(E \mid n) = O(1)$ and n_1, k, d, and δ are computable from n. Let $(x, y) \in [N] \times [N_1]$ be such that $C(x \mid n) \geq k + O(1)$ and $C(y \mid x) \geq n_1$. Let $z = E(x, y)$. Then $C(z \mid m) > m - d$.*

(Note: $O(1)$ means that there exist constants, depending only on the universal machine, for which the statements hold.)

Proof. The proof is similar to the proof of Claim 3.3. We sketch the argument. Suppose $C(z \mid m) \leq m - d$.

Let $A = \{w \in \{0,1\}^m \mid C(w \mid m) \leq m - d + O(1)\}$, where the constant $O(1)$ is chosen so that $|A| \geq 2^{m-d}$. Also note that $|A| \leq 2^{m-d+O(1)}$.

We say that a row v is *bad* if the number A-cells in the $\{v\} \times [N_1]$ rectangle of E is $> \Delta \cdot \frac{|A|}{M} \cdot N_1$. The number of bad rows is at most K, because the table E is (K, D, Δ)-balanced. Therefore a bad row v is described by the information needed to enumerate the bad rows (and this information is derivable from n) and from its rank in the enumeration of bad rows. So, if v is bad, $C(v \mid n) < k + O(1)$.

Since $C(x \mid n) > k + O(1)$, it follows that x is good. Therefore, the number of A-cells in the $\{x\} \times [N_1]$ rectangle of E is $\leq \Delta \cdot \frac{|A|}{M} \cdot N_1 = 2^{\delta - d + n_1 + O(1)}$.

Note that, by our assumption, the cell (x, y) is an A-cell. Then the string y, given x, can be described by the rank of (x, y) among the A-cells in the $\{x\} \times [N_1]$ rectangle of E.

So, $C(y \mid x) \leq \delta - d + n_1 + O(1)$ and the right hand side is less than n_1 for an appropriate choice of the constant $O(1)$ in the relation between d and δ. We obtain that $C(y \mid x) < n_1$, contradiction. ∎

The next lemma establishes the parameters for which balanced tables exist.

Lemma 4.5 *Suppose the parameters satisfy the following relations: $D = O(\Delta), n/\delta = o(N_1)$, and $M = o(\delta \cdot K \cdot N_1)$. Then there exists a table $E : [N] \times [N_1] \to [M]$ that is (K, D, Δ)-balanced.*

Proof. The proof is by the probabilistic method and is presented in the Appendix. ∎

We can now prove Fact 3. The formal statement is as follows.

Theorem 4.6 *Parameters: Let $m(n)$ and $h(n)$ be computable functions such that $m(n) < n$ for all n and $h(n) = \omega(\log \frac{n}{m(n)})$.*

There exists a computable function $E : \{0,1\}^n \times \{0,1\}^{h(n)} \to \{0,1\}^{m(n)}$, such that for every $x \in \{0,1\}^n$ with $C(x \mid n) \geq m(n)$, there exists $\alpha_x \in \{0,1\}^{h(n)}$ such that $C(E(x, \alpha_x) | m) \geq m(n) - o(m(n))$.

Proof. We take $\delta = \frac{n}{2^{0.5h(n)}}$, $d = \delta + c$, where c is the constant from Lemma 4.4, $n_1 = h(n)$.

By Lemma 4.5, there exists a table $E : [N] \times [N_1] \to [M]$ that is (K, D, Δ)-balanced, and by brute force one can build such a table from n. Thus we obtain such a table E with $C(E \mid n) = O(1)$. We take α_x to be a string in $\{0,1\}^{h(n)}$ such that $C(\alpha_x \mid x) \geq h(n)$. Using Lemma 4.4, we obtain that $C(E(x, \alpha_x) \mid m) \geq m(n) - d = m(n) - \frac{n}{2^{0.5h(n)}} - c = m - o(m)$. ∎

Our next goal is to derandomize the construction in Theorem 4.6. As explained in the Introduction the key observation is that checking if a table is balanced can be done, in an approximate sense, by constant-depth circuits with relatively small size.

Lemma 4.7 *The parameters n_1, m, k, d, and δ are positive integers computable from n in polynomial time. We assume $k \leq n, m \leq k, d \leq n$.*

There exists a circuit G of size $\mathrm{poly}(N^K)$ *and constant depth such that for any table $E : [N] \times [N_1] \to [M]$,*

(a) if $G(E) = 1$, then E is $(K, D, 1.03\Delta)$-balanced,

(b) if E is (K, D, Δ)-balanced, then $G(E) = 1$.

Proof. Let $a = (1/0.99)\Delta \cdot 1/D \cdot K \cdot N_1$. Let us fix for the moment a set of rows $B \subseteq [N]$ of size $|B| = K$ and a set of colors $A \subseteq [M]$ of size $|A| = M/D$. Let $x_{B,A}$ be a binary string indicating which cells in the $B \times [N_1]$ rectangle of E are A-colored. Formally, $x_{B,A}$ is the string of length $K \cdot N_1$, whose $\langle i, j \rangle$-th bit is 1 if the cell (i, j) in the rectangle $B \times [N_1]$ of E is an A-cell and 0 if it is not.

By Ajtai's Theorem 2.1, there exists a polynomial-size constant-depth circuit G' (which does not depend on B and A) with a hardwired and such that

- $G'(x_{B,A}) = 1$ if the number of A-cells in $B \times [N_1]$ is at most $(1 - 0.01) \cdot a$, and
- $G'(x_{B,A}) = 0$ if the number of A-cells in $B \times [N_1]$ is at least $(1 + 0.01) \cdot a$.

Now we describe the circuit G.

The circuit G on input an encoding of the table E (having length $N \cdot N_1 \cdot m$) computes in constant depth a string $x_{B,A}$ for every $B \subseteq [N]$ with $|B| = K$ and for every $A \subseteq [M]$ with $A = M/D$. There are $\binom{N}{K}\binom{M}{M/D} = \mathrm{poly}(N^K)$ such strings $X_{B,A}$. Each such string $x_{B,A}$ is the input of a copy of G'. The output gates of all the copies of G' are connected to an AND gate, which is the output gate.

If $G(E) = 1$, then $G'(x_{B,A}) = 1$ for all B's and A's as above. This implies that for all $B \subseteq [N]$ with $|B| \geq K$ and all $A \subseteq [M]$ of size $\geq M/D$, the number of A-cells in the $B \times [N_1]$ rectangle of E is at most $(1 + 0.01)a \leq (1.03) \cdot \Delta \cdot (1/D) \cdot K \cdot N_1$, i.e., E is $(K, D, 1.03\Delta)$-balanced.

In the other direction, if E is (K, D, Δ)-balanced then for all $B \subseteq [N]$ with $|B| = K$ and for all $A \subseteq [M]$ with $A = M/D$, the number of A-cells in $B \times [N_1]$ is at most $\Delta \cdot (1/D) \cdot K \cdot N_1 = (1 - 0.01)a$, which implies that $G(E) = 1$. ∎

We next prove Fact 4. The formal statement is as follows.

Theorem 4.8 *Parameters: Let $m(n)$ and $h(n)$ be polynomial-computable functions such that $m(n) \leq 0.99 \cdot \alpha_{NW} \cdot n$ and $h(n) = \omega(\log(\frac{n}{m(n)}))$.*

There exists a function $E : \{0,1\}^n \times \{0,1\}^{h(n)} \to \{0,1\}^{m(n)}$, computable by an effectively constructible circuit having polynomial size and the following property: For every $x \in \{0,1\}^n$ with $C(x \mid n) \geq m(n) + O(1)$, there exists a string $\alpha_x \in \{0,1\}^{h(n)}$ such that $C(E(x, \alpha_x) \mid m) \geq m - o(m)$.

Proof. Let $k = m(n), \delta = \frac{n}{2^{0.5h(n)}}, d = \delta + c + \log 1.03$ (where c is the constant from Lemma 4.4), and $n_1 = h(n)$.

Let G be the circuit promised by Lemma 4.7 for these parameters. Let d_{Ajtai} be the depth of the circuit G and let α_{NW} be the constant corresponding to d_{Ajtai} in Theorem 2.2.

Let $\tilde{N} = N \cdot N_1 \cdot m$. This is the size of an encoding of a table $E : [N] \times [N_1] \to [M]$. Let NW-gen : $\{0,1\}^{\log^{2d_{Ajtai}+6}(\tilde{N})} \to \{0,1\}^{\tilde{N}}$ be the Nisan-Wigderson pseudo-random generator given by Theorem 2.2 for the depth parameter equal to d_{Ajtai}. Note that $\text{poly}(N^K) \leq 2^{\tilde{N}^{\alpha_{NW}}}$, where $\text{poly}(N^K)$ is the bound from Lemma 4.7 for the size of the circuit G.

The probabilistic argument in Lemma 4.5 can be modified to show that among the tables of type $E : [N] \times [N_1] \to [M]$ the fraction of those which are (K, D, Δ)-balanced is at least 0.51. Since G accepts all such tables,

$$\text{Prob}_{E \in \{0,1\}^{\tilde{N}}}[G(E) = 1] \geq 0.51.$$

Since the circuit G has depth equal to d_{Ajtai} and size bounded by $2^{\tilde{N}^{\alpha_{NW}}}$, it follows that if we replace a random $E \in \{0,1\}^{\tilde{N}}$ by NW-gen(s) for a random seed $s \in \{0,1\}^{\log^{2d_{Ajtai}+6}(\tilde{N})}$, we obtain

$$\text{Prob}_{s \in \{0,1\}^{\log^{2d_{Ajtai}+6}(\tilde{N})}}[G(\text{NW-gen}(s)) = 1] \geq 0.5.$$

We only need the fact that there exists a string $s \in \{0,1\}^{\log^{2d_{Ajtai}+6}(\tilde{N})}$ such that NW-gen(s) is a table $E : [N] \times [N_1] \to [M]$ that is $(K, D, 1.03\Delta)$-balanced. We fix such an s that is computable from n (say, the smallest s that has the property) and the corresponding table E produced by the Nisan-Wigderson pseudo-random generator on seed s.

Let us consider $x \in \{0,1\}^n$ with $C(x \mid n) \geq k$ and $\alpha_x \in \{0,1\}^{n_1}$ with $C(\alpha_x \mid y) \geq n_1$. Since E is $(K, D, 1.03\Delta)$-balanced, it follows from Lemma 4.4 that $C(E(x, \alpha_x) \mid m) \geq m - d = m - o(m)$.

Now, let us view E (which is NW-gen(s)) as a function E : $\{0,1\}^n \times \{0,1\}^{n_1} \to \{0,1\}^m$. From the properties (i.e., the "Moreover ..." in Theorem 2.2) of the Nisan-Wigderson pseudo-random generator, it follows that this function can be computed by a polynomial-size circuit which has s hardwired. Since s is also computable from n, one can compute a description of the circuit, i.e., the circuit is effectively constructible. ∎

References

[Ajt93] M. Ajtai. Approximate counting with uniform constant-depth circuits. *Advances in computational complexity*, pages 1–20, 1993.

[FHP+06] L. Fortnow, J. Hitchcock, A. Pavan, N.V. Vinodchandran, and F. Wang. Extracting Kolmogorov complexity with applications to dimension zero-one laws. In *Proceedings of the 33rd International Colloquium on Automata, Languages, and Programming*, pages 335–345, Berlin, 2006. Springer-Verlag *Lecture Notes in Computer Science #4051.*

[HPV09] John M. Hitchcock, Aduri Pavan, and N. V. Vinodchandran. Kolmogorov complexity in randomness extraction. In *FSTTCS*, pages 215–226, 2009.

[LV93] M. Li and P. Vitanyi. *An introduction to Kolmogorov complexity and its applications*. Springer-Verlag, 1993. 1st edition.

[Mus10] Daniil Musatov. Improving the space-bounded version of Muchnik's conditional complexity theory via "naive" derandomization. *CoRR*, abs/1009.5108, 2010. To appear in CSR 2011.

[NW94] N. Nisan and A. Wigderson. Hardness vs. randomness. *Journal of Computer and System Sciences*, 49:149–167, 1994.

[Rei04] J. Reimann. Computability and fractal dimension. Technical report, Universität Heidelberg, 2004. Ph.D. thesis.

[Vio10] Emanuele Viola. Randomness buys depth for approximate counting. *Electronic Colloquium on Computational Complexity (ECCC)*, 17:175, 2010.

[VV02] Nikolai K. Vereshchagin and Michael V. Vyugin. Independent minimum length programs to translate between given strings. *Theor. Comput. Sci.*, 271(1-2):131–143, 2002.

[Zim10] M. Zimand. Possibilities and impossibilities in Kolmogorov complexity extraction. *SIGACT News*, 41(4), December 2010.

[ZL70] A. Zvonkin and L. Levin. The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms. *Russian Mathematical Surveys*, 25(6):83–124, 1970.

A Appendix

Proof of Claim 3.2.

We use the probabilistic method. It is enough to show the assertion for all B_1, B_2 and A having sizes exactly K_x, K_y, and respectively $\frac{M}{D}$. Let us consider a random function $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^m$. Fix B_1, B_2 and A, satisfying the above requirement on their sizes. By the Chernoff's bound,

$$\begin{aligned}\text{Prob}[|A\text{-cells in } B_1 \times B_2| \geq 2 \cdot \tfrac{|A|}{M} \cdot |B_1| \cdot |B_2|] \\ \leq e^{-(1/3)(|A|/M)|B_1||B_2|} = e^{-(1/3)(1/D)K_xK_y}.\end{aligned}$$

The sets B_1, B_2, and A can be chosen in $\binom{N}{K_x}\cdot\binom{N}{K_y}\cdot\binom{M}{M/D} \leq N^{K_x}\cdot N^{K_y}\cdot(eD)^{M/D} = e^{K_x \ln N + K_y \ln N + (M/D) + (M/D)\ln D}$ ways. Since $t_x \geq 13\log n + I(x:y) + O(1)$ and $d < 6\log n + I(x:y) + O(1)$, it follows that $t_x \geq d + 7\log n + O(1)$ and from here $k_x \geq d + \log n + O(1)$. Since $t_y \geq 7\log n + I(x:y) + O(1)$, it follows that $k_y \geq d + \log n + O(1)$. It can be easily checked that, given these bounds for k_x and k_y and for an appropriate choice of the constant in the definition of d,

$$e^{-(1/3)(1/D)K_xK_y} \cdot e^{K_x \ln N + K_y \ln N + (M/D) + (M/D)\ln D} < 1.$$

Thus, the probability that a random E satisfies the requirements is less than 1, which implies that there exists an E satisfying the claim. ∎

Proof of Lemma 4.5 .

The proof is by the probabilistic method. Consider a random function $E : [N] \times [N_1] \to [M]$. We evaluate the probability that E fails to be (K, D, Δ)-balanced. Note that if E fails to be (K, D, Δ)-balanced, then there exists a set $B \subseteq [N]$ of size exactly K and a set $A \subseteq [M]$ of size exactly M/D such that the fraction of A cells in the $B \times [N_1]$ rectangle of E is greater than $\Delta \cdot |A|/M$. Let us call this latter event $\mathcal{S}$. We show that the probability of $\mathcal{S}$ is less than 1. Fix $B \subseteq [N]$ of size K and $A \subseteq [M]$ of size M/D. For a fixed $(x, y) \in B \times [N_1]$, $\text{Prob}[E(x, y) \in A] = |A|/M$. The expected number of A-cells in $B \times [N_1]$ is $\mu = |B| \cdot N_1 \cdot |A|/M$. Let $\Delta' = \Delta - 1$.

We use the following version of the Chernoff bound. If X is a sum of independent Bernoulli random variables, and the expected value $E[X] = \mu$, then $\text{Prob}[X \geq (1 + \Delta)\mu] \leq e^{-\Delta(\ln(\Delta/3))\mu}$.[3]

Using these Chernoff bounds,

$$\text{Prob}[|A\text{-cells in } B \times [N_1]| > (1 + \Delta')\mu] \leq e^{-\Delta'(\ln(\Delta'/3))\mu}.$$

The set B can be chosen in $\binom{N}{K} \leq N^K$ ways. The set A can be chosen in $\binom{M}{M/D} \leq (eD)^{M/D}$ ways. It follows that the probability of $\mathcal{S}$ is bounded by

$$N^K \cdot (eD)^{M/D} \cdot e^{-\Delta'(\ln(\Delta'/3))\cdot K \cdot N_1 \cdot (1/D)},$$

which, taking into account the relations between parameters, is less than 1. ∎

[3]The standard Chernoff inequality $\text{Prob}(X \geq (1 + \Delta)\mu] \leq \left(\frac{e^\Delta}{(1+\Delta)^{(1+\Delta)}}\right)^\mu$ is presented in many textbooks. It can be checked easily that $\frac{e^\Delta}{(1+\Delta)^{(1+\Delta)}} < e^{-\Delta\ln(\Delta/3)}$.