Information Leakage due to Revealing Randomly Selected Bits

Arash Atashpendar[1(⋆)], A. W. Roscoe[2], and Peter Y. A. Ryan[1]

[1] SnT, University of Luxembourg, Luxembourg
{arash.atashpendar,peter.ryan}@uni.lu
[2] Department of Computer Science, University of Oxford, UK
bill.roscoe@cs.ox.ac.uk

Abstract. This note describes an information theory problem that arose from some analysis of quantum key distribution protocols. The problem seems very natural and is very easy to state but has not to our knowledge been addressed before in the information theory literature: suppose that we have a random bit string y of length n and we reveal k bits at random positions, preserving the order but without revealing the positions, how much information about y is revealed? We show that while the cardinality of the set of compatible y strings depends only on n and k, the amount of leakage does depend on the exact revealed x string. We observe that the maximal leakage, measured as decrease in the Shannon entropy of the space of possible bit strings, corresponds to the x string being all zeros or all ones and that the minimum leakage corresponds to the alternating x strings. We derive a formula for the maximum leakage (minimal entropy) in terms of n and k. We discuss the relevance of other measures of information, in particular min-entropy, in a cryptographic context. Finally, we describe a simulation tool to explore these results.

Keywords: Information Leakage, Quantum Key Distribution, Entropy, Subsequence, Supersequence, Deletion Channel, Simulation

1 Introduction

The problem that we investigate here arose from some analysis of quantum key distribution (QKD) protocols. We do not go into the details of the motivating context here, more detail can be found at [1]. For the moment we just remark that in QKD protocols it is typical for the parties, after the quantum phase, to compare bits of the fresh session key at randomly sampled positions in order to obtain an estimate of the Quantum Bit Error Rate (QBER). This indicates the proportion of bits that have been flipped as the result of either noise or eavesdropping on the quantum channel. This serves to bound the amount of information leakage to any eavesdropper, and as long as this falls below an appropriate threshold the parties continue with the key reconciliation and secrecy amplification steps.

Usually, the sample set is agreed and the bits compared using un-encrypted but authenticated exchanges over a classical channel, hence the positions of

the compared bits are known to a potential eavesdropper and these bits are discarded. In [1], it is suggested that the sample set be computed secretly by the parties based on prior shared secrets. They still compare the bits over an un-encrypted channel, but now an eavesdropper does not learn where the bits lie in the key stream. This prompts the possibility of retaining these bits, but now we must be careful to bound the information leakage and ensure that later privacy amplification takes account of this leakage.

Further advantages of the above approach are that it provides implicit authentication at a very early stage and it ensures fairness in the selection of the sampling, i.e. neither party controls the selection.

In practice it would probably be judged too risky to retain these bits on forward secrecy grounds: leakage of the prior secret string at a later time would compromise these bits. Nonetheless, the possibility does present the rather intriguing mathematical challenge that we address in this paper.

The structure of the paper is as follows: in the next section we give the problem statement, some notation and the necessary background theory. Section 3 presents our approach for solving the problem as well as the obtained results, followed by a few discussions on privacy amplification and alternative approaches in Section 4. Section 5 describes how we use simulations to obtain numerical results and to tackle some of the problems addressed in this paper for which deriving analytic expressions proved to be difficult. Finally, we conclude by summarizing our contributions in Section 6.

2 Problem Statement

Given an alphabet $\Sigma = \{0, 1\}$, Σ^n denotes the set of all Σ-strings of length n. Consider a bit string y of length n chosen at random from the space of all possible bit strings of length n, i.e. $y \in \Sigma^n$. More precisely, we assume that the probability distribution over the n-bit strings is flat. We assume that the bits are indexed 1 through n and a subset S of $\{1, ..., n\}$ of size k ($k \leq n$) is chosen at random and we reveal the bits of y at these indices, preserving the order of the bits but without revealing S. Call the resulting, revealed string x. We assume that S is chosen with a flat distribution over the set of subsets of $\{1, ...n\}$ of size k, thus every subset of size k is equally probable. As an example, suppose that for $n = 12$ and $k = 4$ we have:

$$y = \langle 011000011001 \rangle$$

and we choose $S = \{2, 4, 5, 8\}$, then $x = \langle 1001 \rangle$.

The question now is, what is the resulting information leakage about y? We assume that the "adversary" knows the rules of the game, i.e. she knows n and she knows that the leaked string preserves the order but she does not know the chosen S mask. In particular, can we write the leakage as a function purely of k and n or does it depend on the exact form of x? If it does depend on x, can we bound this?

To illustrate: if you reveal 0 bits then obviously you reveal nothing about the full string. If you reveal just one bit ($k = 1$) and suppose that it is a 0, then essentially all you have revealed about the full string is that the all 1 string is not possible. At the other extreme, if you reveal all the bits ($k = n$) then obviously you reveal all n bits of the original string. For $k = n/2$, we see that from Theorem 1 the number of possible y strings is $(2^n)/2$, which for a flat distribution would correspond to exactly 1 bit of leakage. However, in our problem the distribution departs from flat so the leakage is in fact a little more than 1 bit. So intuitively the function starts off very shallow but rises very fast as k approaches n.

2.1 Notation

Let us introduce some notation. First we define $x = Mask(y, S)$ to mean that the string y filtered by the mask S gives the string x. Now we define the *uncertainty set*: given an x and n, this is the set of y strings that could project to x for some mask S.

$$\Upsilon_{n,x} := \{y \in \{0,1\}^n, \exists S \bullet Mask(y, S) = x\}$$

- Let $\omega_x(y)$ denote the number of distinct ways that y can project onto x. We will refer to this as the weight of y (w.r.t x).

$$\omega_x(y) := |\{S \in \mathcal{P}(N) : Mask(y, S) = x\}| .$$

- and $\mu_{n,x}$ the number of configurations for n and x, i.e. the number of pairs $\{y, S\}$ such that $Mask(y, S) = x$. It is easy to see that this is given by:

$$\mu_{n,k} = \binom{n}{k} \cdot 2^{n-k} \tag{1}$$

The concepts presented here are closely related to the notions of subsequences, here denoted by x strings, and supersequences (y strings), in formal languages and combinatorics on words. They also crop up in coding theory as a maximum likelihood decoding in the context of deletion and insertion channels.

2.2 Related Work

The closest results to our work are mainly found either in studies dealing with subsequences and supersequences or in the context of *deletion channels*. However, the questions addressed in this paper remain open in related studies. The main problem boils down to determining the probability distribution discussed in the previous section. Here we give a brief survey of the most relevant and closely related results in the literature.

This is a copy of the author preprint. The final authenticated version is available online at:
https://doi.org/10.1007/978-3-319-26096-9_33

Subsequences and Supersequences: Despite their rather simple descriptions, the spaces of subsequences and supersequences remain largely unexplored and present many unanswered questions. Fundamental results can be found in the works of Levenshtein, Hirschberg and Calabi [2,3,4,5] who provide tight upper and lower bounds on the number of distinct subsequences. Furthermore, it was proved by Chase [6] that the number of distinct k-long subsequences is maximized by repeated permutations of an alphabet Σ, i.e. no letter appears twice without all of the other letters of Σ intervening. Flaxman et al. [7] also provide a probabilistic method for determining the string that maximizes the number of distinct subsequences. Results for the mean and the variance of subsequences for the sequence searching problem, also known as the hidden pattern problem, can be found at [8].

For a thorough presentation of efficient algorithms for computing the number of distinct subsequences, e.g. using dynamic programming, and related problems in the realm of DNA sequencing, we refer the reader to [9,10,11,12].

Maximum Likelihood Decoding in Deletion Channels: In a deletion channel [13], for a received sequence, the probability that it arose from a given codeword is proportional to the number of different ways it is contained as a subsequence in that codeword. This translates into a maximum likelihood decoding for deletion channels as follows: For a received sequence, we count the number of times it appears as a subsequence of each codeword and we choose the codeword that admits the largest count. The problem of determining and bounding these particular distributions remains unexplored and presents a considerable number of open questions. Case-specific results for double insertion/deletion channels can be found in [14]. Moreover, improved bounds for the number of subsequences obtained via the deletion channel and proofs for how balanced and unbalanced strings lead to the highest and lowest number of distinct subsequences are given in [15].

2.3 Entropy Measures

The obvious follow-on question to the problem posed at the start of this section is: what is the appropriate measure of information to use? Perhaps the simplest measure is the Hartley measure, the log of the cardinality of the uncertainty set. This coincides with the Shannon measure if the probability distribution is uniform. In this case the solution is simple as we will see below: the cardinality of the uncertainty set is a simple function of n and k. However, the probability distribution turns out to be rather far from uniform, so Hartley does not seem appropriate here.

Thought of purely as an information theory puzzle, the standard commonly used measure is Shannon's [16]. For this we have a number of interesting results and observations. In particular, our observations suggest that the maximum leakage for all n and k occurs for the all zero or all one x strings and we have a formula for the leakage in these cases. However, we have not yet been able to

prove this conjecture, although we do have intuitions as to why this appears to be the case.

Given the cryptographic motivation for the problem, it is worth considering whether alternative information measures are in fact more appropriate. The Shannon measure has a very specific interpretation: the expected number of binary questions required to identify the exact value of the variable. In various cryptographic contexts, this might not be the most appropriate interpretation. For example, in some situations it might not be necessary to pin down the exact value and a good approximation may be damaging. In our context, the session key derived from the key reconciliation phase will be subjected to privacy amplification to reduce the adversary's knowledge of the key to a negligible amount. What we really need therefore is a measure of the leakage that can be used to control the degree of amplification required. This question has been extensively studied in [17,18,19,20,21], and below we summarize the key results.

Various measures of entropy may be applicable depending on the parameters of the context in question, such as the scheme used for privacy amplification, e.g. universal hashing vs. randomness extractors or whether a distinction is made between passive adversaries and active adversaries [17]. As noted in the works of Bennett et al. [21,22], the Rényi entropy [23,24] provides a lower bound on the size of the secret key s' distillable from the partially secret key s initially shared by Alice and Bob. Moreover, it is shown in [17], that the min-entropy provides an upper bound on the amount of permissible leakage and specific constraints are derived as a function of the min-entropy of s and the length of the partially secret string. More recently, Renner and Wolf show in [18] that the Shannon entropy H can be generalized and extended to two simple quantities, H_0^ε and H_∞^ε, called smooth Rényi entropy values, which provide tight bounds for privacy amplification and information reconciliation in contexts such as QKD, where the assumption of having independent repetitions of a random experiment is generally not satisfied.

For the purpose of our study, we consider the following measures of information, which can be considered as special cases of the Rényi Entropy.

Rényi Entropy of order α**.** For $\alpha \geq 0$ and $\alpha \neq 1$, the Rényi entropy of order α of a random variable X is

$$H_\alpha(X) = \frac{1}{1-\alpha} \log_2 \sum_{x \in \mathcal{X}} P_X(x)^\alpha. \tag{2}$$

Hartley Entropy. The Hartley measure corresponds to Rényi entropy of order zero and is defined as

$$H_0(X) := -\log_2 |\mathcal{X}| \,. \tag{3}$$

Second-order Rényi Entropy. For $\alpha = 2$, we get the collision entropy, also simply referred to as the Rényi entropy

https://doi.org/10.1007/978-3-319-26096-9_33

$$R(x) = H_2(X) := -\log_2 \sum_{x \in \mathcal{X}} P_X(x)^2. \tag{4}$$

Shannon Entropy. As $\alpha \to 1$, in the limit we get the Shannon entropy of a random variable X

$$H(X) = -\sum_{x \in \mathcal{X}} P_X(x) \cdot \log_2 P_X(x). \tag{5}$$

Min-Entropy. In the limit, as $\alpha \to \infty$, H_α converges to the min-entropy of a random variable X

$$H_\infty(X) := -\log_2 \max_{x \in \mathcal{X}}(P_X(x)). \tag{6}$$

As noted in [17], the entropy measures given above satisfy

$$H(X) \geq H_2(X) \geq H_\infty(X) \tag{7}$$

3 Information Leakage

In this section we show that the size of the uncertainty set only depends on n and k and provide an expression for computing its cardinality, followed by a proof. We then analyze the amount of information leakage and observe that the maximal leakage corresponds to the x string being all zeros or all ones and that the minimum leakage corresponds to the alternating x strings. We also derive closed form expressions for the maximum leakage (minimal entropy) in terms of n and k for the measures of entropy introduced in Section 2.

3.1 Cardinality of the Uncertainty Set

Theorem 1. *For given n and k the cardinality of $\Upsilon_{n,x}$ is independent of the exact x string. Furthermore, $|\Upsilon_{n,x}|$ is given by:*

$$|\Upsilon_{n,x}| = \sum_{r=k}^{n} \binom{n}{r} \tag{8}$$

Proof. $\gamma_{n,k}$ satisfies the following recursion:

$$\gamma_{n,k} = \gamma_{n-1,k} + \gamma_{n-1,k-1} \tag{9}$$

with base cases: $\gamma_{n,n} = 1$ and $\gamma_{n,0} = 2^n$.

The base cases are immediate. To see how the recursion arises, consider the following cases:

– Partition the y strings into those that have a mask overlapping the first bit of y and those that do not.

- For the former, we can enumerate them simply as the number of y strings of length $n-1$ with ≥ 1 projections to the tail of x, i.e. $x*$, i.e. $\gamma_{n-1,k-1}$.
- For the latter, the number is just that of the set of y strings of length $n-1$ with ≥ 1 projection to x, which has length k, i.e. $\gamma_{n-1,k}$.

The solution to this recursion with the given base cases is:

$$\gamma_{n,k} = \sum_{r=k}^{n} \binom{n}{r} \tag{10}$$

This is most simply seen by observing that the recursion is independent of the exact x, hence we can choose the x string comprising k 0s. Now we see that $|\Upsilon_{n,x}|$ is simply the number of distinct y strings with at least k 0s, and the result follows immediately.

□

If the conditional distribution over $\Upsilon_{n,x}$ given the observation of x were flat, we would be done: we could compute the entropy immediately. However, it turns out the distribution is far from flat, and indeed its shape depends on the exact x string. This is due to the fact that given an observed x, the probability that a y gave rise to it is proportional to the weight of y, i.e. the number of ways that y could project to x, i.e. $|\{S|Mask(y,S) = x\}|$. This can vary between 1 and $\binom{n}{k}$.

3.2 Shannon Entropy

Here we will assume that the leakage is measured as the drop in the Shannon entropy of the space of possible y strings. Clearly, before any observation the entropy is n bits. We observe that the maximal leakage occurs when x is either the all 0 or the all 1 string and we derive an expression for the corresponding entropy of $\Upsilon_{n,x}$.

3.3 Minimal Shannon Entropy

Assuming that the maximal leakage occurs for the all zero (or all one) x string we derive the formula for the maximal leakage (minimum entropy of $\Upsilon_{n,x}$) as follows: observe that the number of elements of $\Upsilon_{n,k}$ with j 1's is $\binom{n}{j}$. Note further that for given j the number of ways that a y string with j 1's can yield x is $\binom{n-j}{k}$. Consequently, the probability that y was a given string with j 1's given the observation of x is:

$$P(y_j|x) = \frac{\binom{n-j}{k}}{\mu_{n,k}} \tag{11}$$

Where $\mu_{n,k}$ is the normalization, i.e. the total number of configurations that could give rise to a given x:

$$\mu_{n,k} = \binom{n}{k} \times 2^{n-k}$$

https://doi.org/10.1007/978-3-319-26096-9_33

Now, inserting these terms into the formula for the Shannon entropy given in Eq. 5, we get:

$$H_{n,k} = -\sum_{j=0}^{n-k} \binom{n}{j} \times \frac{\binom{n-j}{k}}{\mu_{n,k}} \times \log_2 \left(\frac{\binom{n-j}{k}}{\mu_{n,k}} \right) \tag{12}$$

For the original cryptographic motivation of this problem, more specifically in the context of privacy amplification, it is arguably an upper bound on the maximum leakage or the amount of information that Eve has gained that we are after [25]. However, it is also interesting to better understand the mean and range of the entropy for given n and k, but coming up with analytic forms for these appears to be much harder. We switch therefore to simulations to give us a better feel for these functions.

3.4 Minimal Rényi Entropy

The expression provided here is also based on the empirical results that conjecture that the minimal Rényi entropy is attained by 0^k or 1^k.

Inserting the derived expression given in Eq. 11 corresponding to the maximal leakage into the formula of the second-order Rényi entropy given in Eq. 4, we obtain the following expression for the minimal Rényi entropy:

$$R(X) = H_2(X) := -\log_2 \sum_{j=0}^{n-k} \binom{n}{j} \cdot \left(\frac{\binom{n-j}{k}}{\mu_{n,k}} \right)^2 \tag{13}$$

The derived expression agrees with the experiments driven by the numerical computation presented in Section 5.

3.5 Min-Entropy

The most conservative measure of information in the Rényi family is the min-entropy, and this is of interest when it comes to privacy amplification.

This turns out to be more tractable than the Shannon entropy. In particular it is immediate that the smallest Min-Entropy is attained by the all zero or all one x strings: the largest weight of a y string, and hence probability, is $\binom{n}{k}$ and this is attained by $x = 0^k$ and $y = 0^n$. Thus we can derive an analytic form for the minimum Min-Entropy $H_\infty(X)$ by inserting the derived term for maximal probability given in Eq. 11 into Eq. 6, and thus we get:

$$Min(H_\infty(X)) := -\log_2 \left(\frac{\binom{n}{k}}{\mu_{n,k}} \right) \tag{14}$$

and this immediately simplifies to:

$$Min(H_\infty(X)) := n - k \tag{15}$$

It is clear that this indeed corresponds to the most pessimistic bound of the leakage and can be thought of as assuming that the adversary gets to know the exact positions of the leaked bits.

The Min-Entropy, $H_\infty(X)$, is based on the most likely event of a random variable X. Therefore, this term sets an upper bound on the number of leaked bits, which can be then used in the parameterization of the compression function used in privacy amplification as described in [22].

Using Eq. 7 and the analytic forms given above for the lower bound on the Shannon entropy as well as the min-entropy, we can effectively set loose upper and lower bounds on the Rényi entropy.

3.6 Maximum Entropy

Another observation derived from empirical results obtained by simulation is that it also appears that the minimal leakage (max H) occurs when x comprises alternating 0s and 1s, e.g. $x = 101010...$, as shown in Fig. 3. We have seen that for a given n and k, the total number of masks and the number of compatible y strings are constant for all x strings. Therefore, the change in entropy of the Υ space for different x strings is solely dictated by how the masks are distributed among the compatible y strings, i.e. the contribution of each $y \in \Upsilon_{n,x}$ to the total number of masks.

4 Privacy Amplification and Alternative Approaches

This section gives a brief overview of the context to which this study applies and also analyzes the presented problem from a Kolmogorov complexity point of view. We then propose an approach for estimating the expected leakage, and finally we point out a duality between our findings and similar results in the literature.

4.1 Privacy Amplification

PA involves a setting in which Alice and Bob start out by having a partially secret key denoted by the random variable W, e.g. a random n-bit string, about which Eve gains some partial information, denoted by a correlated random variable V. This leakage can be in the form of some bits or parities of blocks of bits of W or some function of W [22]. Provided that Eve's knowledge is at most $t < n$ bits of information about W, i.e. $R(W|V = v) \geq n - t$, with R denoting the second-order Rényi entropy, Alice and Bob can distill a secret key of length $r = n-t-s$ with s being a security parameter such that $s < n - t$. The security parameter s can be used to reduce Eve's knowledge to an arbitrarily small amount, e.g. in the context of universal hash functions, it can be used to adjust the reduction size of the chosen compression function $g : \{0,1\}^n \mapsto \{0,1\}^r$.

The function g is publicly chosen by Alice and Bob at random from a family of universal hash functions, here denoted by the random variable G, to obtain $K =$

$g(W)$, such that Eve's partial information on W and her complete information on g give her arbitrarily little information about K. The resulting secret key K is uniformly distributed given all her information. It is also shown by Bennett et al. in [22] that $H(K|G, V = v) \geq r - 2^{-s}/\ln 2$, provided only that $R(W|V = v) \geq n - t$. The value of s can be considered a fixed value and comparatively small, typically not larger than 30, as the key length increases.

It is worth noting that the measure of information used in privacy amplification for defining the bound on leakage or the minimum length of the secret key that can be extracted, may vary depending on criteria such as the algorithms used in the amplification scheme and the channel being authenticated or not. For instance, as shown in [17], when randomness extractors are used instead of universal hash functions, the bound for secure PA against an active adversary is defined by the adversary's min-entropy about W. Various schemes for performing PA over authenticated and non-authenticated channels have been extensively studied in [17],[21],[26].

In QKD, privacy amplification constitutes the last sub-protocol that is run in a session and thus it takes place after the information reconciliation phase. The leakage studied in this paper deals with reduced entropy before the information reconciliation phase. However, this simply means that the leakage quantified here would in fact contribute to the t bits leaked to Eve.

4.2 Kolmogorov-Chaitin Complexity

From a purely information theoretical point of view, quantifying the amount of information leakage in terms of various measures of entropy such as the Shannon entropy is arguably what interests us. However, from a cryptographic standpoint, a complexity analysis of exploring the search space by considering the Kolmogorov complexity, provides another perspective in terms of the amount of resources needed for describing an algorithm that reproduces a given string.

In such a context, what matters for an attacker is how efficiently a program can enumerate the elements of the search space. In other words, whether it can enumerate the space in the optimal way, to minimize the expected time to terminate successfully. To illustrate this point, consider the case of the all 0 x string for which we can start with the all 0 y string, then move to y strings with one 1, then two 1s, and so on and so forth. For other generic x strings, carrying out this procedure in an efficient manner becomes more involved.

4.3 Estimating Expected Leakage

Our primary goal was to compute the leakage for a given x and the maximum leakage for given n and k, however, estimating the average leakage might also be of some interest. Since an exact computation depends on a rigorous understanding of the Υ space and its governing probability distribution, we suggest an approach that moves the problem from the space of supersequences to that of subsequences such that further developments in the latter can enable a more fine-grained estimation of the expected leakage.

Let Y be the random variable denoting the original random sequence of n bits and X the random variable denoting the k bits of leakage from Y. The average leakage can be expressed in terms of the entropy of Y minus the conditional entropy of Y conditioned on the knowledge of X, i.e. $H(Y) - H(Y|X)$. While $H(Y|X)$ may seem hard to compute without the joint probability mass function of X and Y, we can use Bayes' rule for conditional entropy [27] to reformulate the expression as follows.

$$H(Y) - H(Y|X) = H(X) - H(X|Y)$$

With random Y, X is a uniformly distributed k-bit string and thus we have $H(X) = k$. This leaves us with $H(X|Y)$, and this reformulation allows us to define the entropy space in terms of projection weights, $\omega_x(y)$, assigned to the subsequences of each $Y = y$. Currently, as shown in [3], we only know the expected number of distinct subsequences given an n and t:

$$E_t(n) = \sum_{i=0}^{t} \binom{n-t-1+i}{i} \lambda^i.$$

with t being the number of deleted bits from the n-long y string, i.e. $t = n-k$, and $\lambda = 1-\frac{1}{|\Sigma|}$, which in the binary case, $\Sigma = \{0, 1\}$, would simply be $\lambda = 1-\frac{1}{2}$. With this measure, we can get a rough estimate on the expected weight, which can then be used to estimate the average entropy, but this only gives us a very coarse-grained estimation of the expected leakage. Therefore, a better understanding of the exact number of distinct subsequences would lead to a more fine-grained estimation of the expected leakage.

4.4 Duality: Subsequences vs. Supersequences

An interesting observation resulting from our findings is that the two x strings of interest in the space of supersequences, i.e. the all zero or all one strings (single run) $0^+|1^+$, denoted here by σ and the alternating x strings: $(\epsilon|1)(01)^+(\epsilon|0)$, denoted here by α also represent the most interesting strings in the space of distinct subsequences.

More precisely, in our study we observe that single run sequences σ lead to the least uniform distribution of masks over the compatible supersequences, whereas the alternating sequences α yield the distribution of masks closest to the uniform distribution. Similarly, in the space of subsequences, σ lead to the minimum number of k-long distinct subsequences and α generate the maximum number of k-long distinct subsequences.

5 Simulations

In this section we first give a brief description of how our simulator [28] carries out the numerical experiments and then we discuss the obtained results with the help of a few plots that are aimed at describing the structure of the Υ spaces.

https://doi.org/10.1007/978-3-319-26096-9_33

We will refrain from elaborating on all the functionalities of the simulator as this would be beyond the scope of this paper. Instead, we focus on a select few sets of empirical results that were obtained from our experiments. We refer to [28] for more information and details.

The main motivation behind the numerical approach driven by simulations lies in the rather complicated structure of the Υ spaces. As deriving analytic forms for describing the entire space seems to be hard, we rely on simulating the spaces of interest in order to explore their structure.

5.1 Methodology

The simulator relies on parallel computations for generating, sampling and exploring the search spaces. The numerical experiments are carried out in two phases: First the simulator generates the Υ spaces that have various structures satisfying predefined constraints and then it proceeds to performing computations on the generated data sets.

The pseudo-code given in Alg. 1 provides an example that illustrates one of the main tasks accomplished by the simulator: Given an n and an x string, we generate the corresponding Υ space containing the compatible y strings, compute the projection count $\omega_x(y)$ of its members, and compute its exact entropy.

Algorithm 1 Compute $H_\alpha(\Upsilon_{n,x})$

```
1: function computeUpsilonEntropy(n, x)
2:     SN ← Generate the space of bit strings of length n
3:     Υ_{n,x} ← Filter SN and reduce it to {y||y| = n, ∃S • Mask(y, S) = x}
4:     probArray ← []
5:     for y_i in Υ_{n,x} do
6:         ω_x(y_i) ← computeProjectionCount(y_i, x)
7:         probArray[i] ← ω_x(y_i)/N
8:     H_α ← computeH_α(probArray)
9:     return H_α
```

5.2 Results Discussion

In this section, we present and discuss a select subset of our results with the help of plots generated by the simulator that provide a better insight into the structure of the Υ spaces.

As mentioned before, one of the main observations resulting from numerical simulations is that the shape of the probability distributions leading to the entropy values of x strings for a given n and k, is mainly determined by how evenly the number of projecting y strings are distributed across the possible projection counts for a given n and k. This observation is illustrated in Fig. 1.

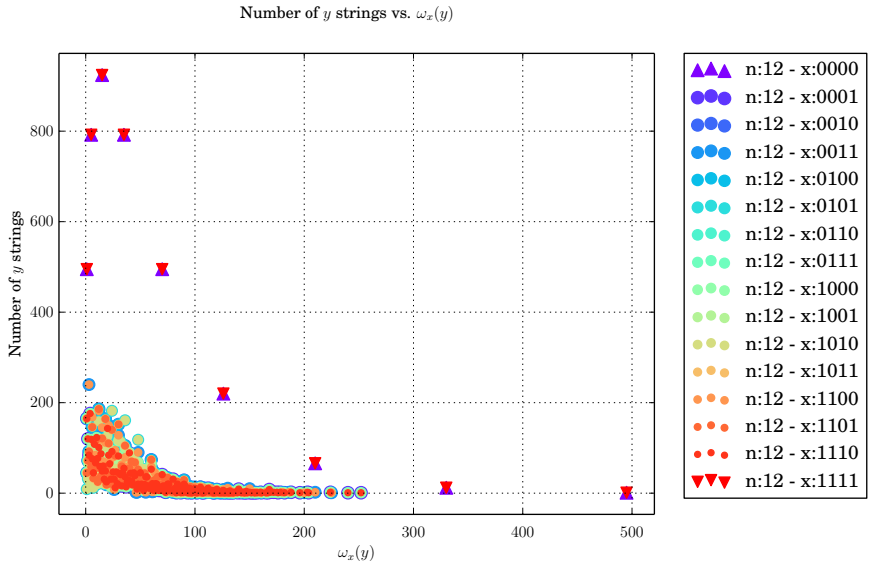

Fig. 1. Distinct count of y strings admitting the same $\omega_x(y)$ for $n = 12$ and $k = 4$.

Following from Theorem 1, for a given n and k, the observables computed and plotted in Fig. 1 for any x string satisfy the following

$$\sum_{i=1}^{g(n,x)} c_i = |\Upsilon_{n,k}| \tag{16}$$

Furthermore, the sum of the product of c_i and $\omega_x(y_i)$ is equal to a constant for all x strings:

$$\sum_{i=1}^{g(n,x)} c_i \cdot \omega_x(y_i) = \eta_{n,k} \tag{17}$$

With c denoting the values on the y-axis, i.e. the count of y strings projecting $\omega_x(y)$ times, and with $\omega_x(y)$ denoting the number of distinct ways that y can project onto x, and finally with $\eta_{n,k}$ being a constant for any given n and k and $g(n, x)$ being a function of n and x that denotes the number of data points corresponding to the distinct count of y strings that have the same $\omega_x(y)$.

This means that for a given n and k, the total number of projection counts in the corresponding Υ space is independent of the x strings. We can see that for the x strings yielding the maximum amount of leakage, i.e. $x = 1^k|0^k$, the lower number of data points is compensated by larger values for the distinct number of y strings admitting larger projection count values, hence showing a much more biased structure in the distribution with respect to generic x strings. Conversely, the distributions for the remainder of the x strings are considerably dampened

https://doi.org/10.1007/978-3-319-26096-9_33

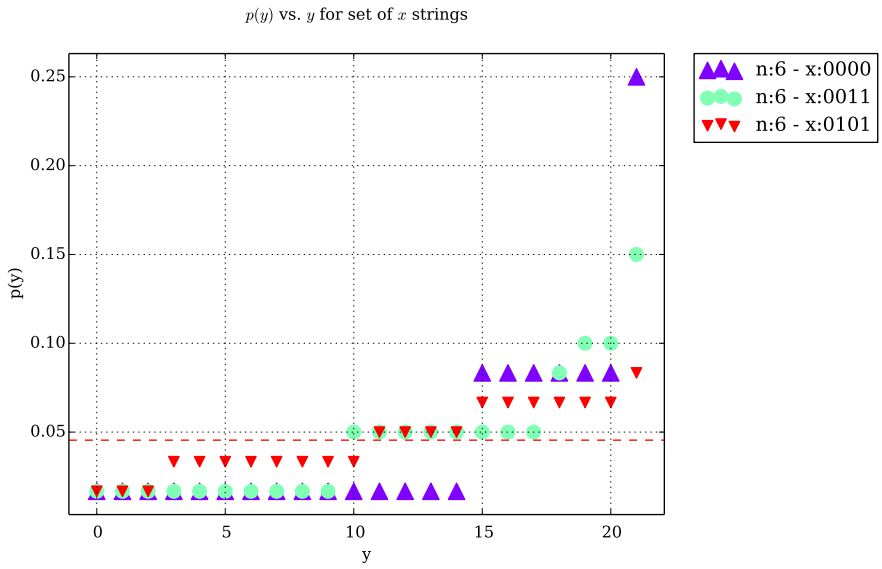

Fig. 2. Probability distributions of Υ spaces for given n and x with y strings enumerated by indices and the red dotted line showing the uniform distribution.

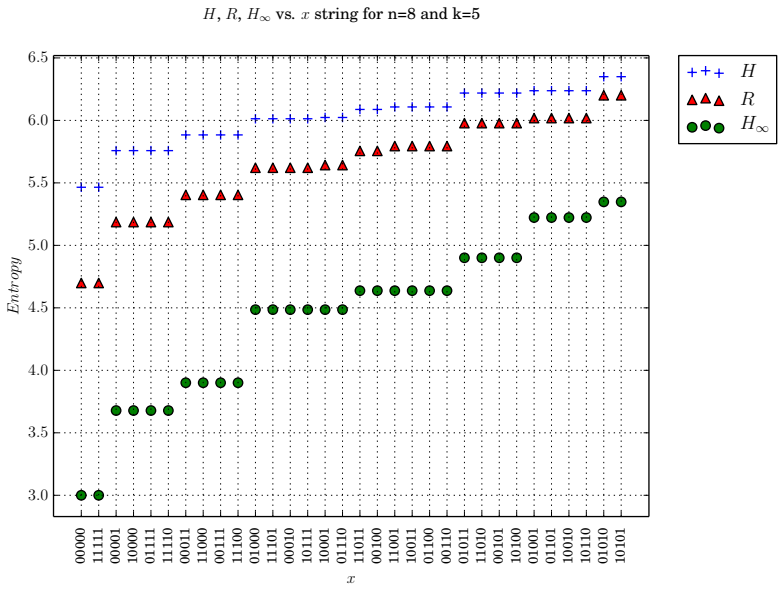

Fig. 3. H (Shannon), R_2 (second-order Rényi entropy) and H_∞ (min-entropy) vs. x strings for $n = 8$ and $k = 5$.

This is a copy of the author preprint. The final authenticated version is available online at:
https://doi.org/10.1007/978-3-319-26096-9_33

and noticeably closer to a flat distribution and are thus less biased compared to $x = 1^k|0^k$ strings, which in part explains the correspondingly higher entropy values. In particular, the alternating ones and zeros string admits the highest degree of dispersion in terms of the distribution of the masks and thus yields the lowest entropy.

The resulting probability distributions leading to the computed entropy values are illustrated in Fig. 2. An immediate observation is that the distribution of the projecting y strings for the 0^k or 1^k strings has the largest outliers. However, this alone does not capture the role of the shape of the probability distribution. Therefore, one could argue that the probability distribution that admits the largest Kullback-Leibler distance from the uniform distribution, i.e. the most biased distribution, yields the lowest entropy, and the conjecture that we put forth is that this distribution is given by the all 0 or all 1 x strings.

The plot shown in Fig. 3, illustrates three measures of entropy, namely the Shannon entropy (H), the second-order Rényi entropy (R) and the min-entropy (H_∞) as a function of n and k for all the 2^k x strings for $n = 8$ and $k = 5$. The presented empirical results validate our conjecture that the all zero or the all one strings yield the minimum entropy and that the alternating zeros and ones string gives the maximum entropy.

6 Conclusions

We have described an information theory problem that arose from some investigations into quantum key establishment protocols. As far as we are aware, the problem, despite its seeming to be very natural and simple to state, has not been investigated in the mathematical literature. We have shown that the maximum leakage, measured in terms of the drop in the entropy of the space of compatible y strings, corresponds to the all zero or all one observed strings.

We have presented analytic forms for the Shannon entropy, the second-order Rényi entropy, and the min-entropy for these cases. Moreover, we have discussed the relevance of these measures specifically in the context of privacy amplification in QKD protocols. We have also noted that the simulations suggest that the minimal leakage corresponds to the x strings comprising alternating zeros and ones. Moreover, we pointed out an interesting duality between our results and existing results in the literature for the space of subsequences.

We have also described a simulation program to explore these results. This is available at [28].

Acknowledgments

We would like to thank Philip B. Stark, Jean-Sébastien Coron, Marc Pouly and Ulrich Sorger for all the helpful comments and discussions.

https://doi.org/10.1007/978-3-319-26096-9_33

References

1. Ryan, P.Y., Christianson, B.: Enhancements to prepare-and-measure based qkd protocols. In: Security Protocols XXI. Springer (2013) 123–133
2. Hirschberg, D.S.: Bounds on the number of string subsequences. In: Combinatorial Pattern Matching, Springer (1999) 115–122
3. Hirschberg, D.S., Regnier, M.: Tight bounds on the number of string subsequences. Journal of Discrete Algorithms **1**(1) (2000) 123–132
4. Calabi, L., Hartnett, W.: Some general results of coding theory with applications to the study of codes for the correction of synchronization errors. Information and Control **15**(3) (1969) 235–249
5. Levenshtein, V.I.: Efficient reconstruction of sequences from their subsequences or supersequences. Journal of Combinatorial Theory, Series A **93**(2) (2001) 310–332
6. Chase, P.J.: Subsequence numbers and logarithmic concavity. Discrete Mathematics **16**(2) (1976) 123–140
7. Flaxman, A., Harrow, A.W., Sorkin, G.B.: Strings with maximally many distinct subsequences and substrings. Electron. J. Combin **11**(1) (2004) R8
8. Flajolet, P., Guivarc'h, Y., Szpankowski, W., Vallée, B.: Hidden pattern statistics. In: Automata, Languages and Programming. Springer (2001) 152–165
9. Middendorf, M.: Supersequences, runs, and cd grammar systems. Developments in Theoretical Computer Science **6** (1994) 101–114
10. Lothaire, M.: Applied combinatorics on words. Volume 105. Cambridge University Press (2005)
11. Rahmann, S.: Subsequence combinatorics and applications to microarray production, dna sequencing and chaining algorithms. In: Combinatorial Pattern Matching, Springer (2006) 153–164
12. Elzinga, C., Rahmann, S., Wang, H.: Algorithms for subsequence combinatorics. Theoretical Computer Science **409**(3) (2008) 394–404
13. Mitzenmacher, M., et al.: A survey of results for deletion channels and related synchronization channels. Probability Surveys **6** (2009) 1–33
14. Swart, T.G., Ferreira, H.C.: A note on double insertion/deletion correcting codes. IEEE Transactions on Information Theory **49**(1) (2003) 269–273
15. Liron, Y., Langberg, M.: A characterization of the number of subsequences obtained via the deletion channel. In: Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on, IEEE (2012) 503–507
16. Shannon, C.E.: A mathematical theory of communication. ACM SIGMOBILE Mobile Computing and Communications Review **5**(1) (2001) 3–55
17. Maurer, U., Wolf, S.: Privacy amplification secure against active adversaries. In: Advances in Cryptology—CRYPTO'97. Springer (1997) 307–321
18. Renner, R., Wolf, S.: Simple and tight bounds for information reconciliation and privacy amplification. In: Advances in Cryptology-ASIACRYPT 2005. Springer (2005) 199–216
19. Maurer, U.M.: Secret key agreement by public discussion from common information. Information Theory, IEEE Transactions on **39**(3) (1993) 733–742
20. Cachin, C.: Entropy measures and unconditional security in cryptography. PhD thesis, Swiss Federal Institute of Technology Zurich (1997)
21. Cachin, C., Maurer, U.M.: Linking information reconciliation and privacy amplification. journal of Cryptology **10**(2) (1997) 97–110
22. Bennett, C.H., Brassard, G., Crépeau, C., Maurer, U.M.: Generalized privacy amplification. Information Theory, IEEE Transactions on **41**(6) (1995) 1915–1923

This is a copy of the author preprint. The final authenticated version is available online at:
https://doi.org/10.1007/978-3-319-26096-9_33

23. Renyi, A.: On measures of entropy and information. In: Fourth Berkeley symposium on mathematical statistics and probability. Volume 1. (1961) 547–561
24. MacKay, D.J.: Information theory, inference, and learning algorithms. Volume 7. Citeseer (2003)
25. Bennett, C.H., Brassard, G., Robert, J.M.: Privacy amplification by public discussion. SIAM journal on Computing **17**(2) (1988) 210–229
26. Carter, J.L., Wegman, M.N.: Universal classes of hash functions. In: Proceedings of the ninth annual ACM symposium on Theory of computing, ACM (1977) 106–112
27. Cover, T.M., Thomas, J.A.: Elements of information theory. John Wiley & Sons (2012)
28. Atashpendar, A., Ryan, P.Y.A.: Qkd and information leakage simulator (September 2014) Available at `http://www.qkdsimulator.com`.