Weak randomness seriously limits the security of quantum key distribution

Jan Bouda,[1,*] Matej Pivoluska,[1] Martin Plesch,[1,2] and Colin Wilmott[1]
[1]*Faculty of Informatics, Masaryk University, Brno, Czech Republic*
[2]*Institute of Physics, Slovak Academy of Sciences, Bratislava, Slovakia*
(Received 13 June 2012; published 7 December 2012)

In usual security proofs of quantum protocols the adversary (Eve) is expected to have full control over any quantum communication between any communicating parties (Alice and Bob). Eve is also expected to have full access to an authenticated classical channel between Alice and Bob. Unconditional security against any attack by Eve can be proved even in the realistic setting of device and channel imperfection. In this paper we show that the security of quantum key distribution protocols is ruined if one allows Eve to possess a very limited access to the random sources used by Alice. Such knowledge should always be expected in realistic experimental conditions via different side channels.

DOI: 10.1103/PhysRevA.86.062308 PACS number(s): 03.67.Dd, 03.67.Ac, 03.67.Hk

I. INTRODUCTION

The emergence of quantum theory in the early 20th century led to a revolution in many areas of physics. One of its main features was the introduction of intrinsic randomness, originating from the very nature of the theory. This probabilistic nature led to questioning of concepts of (macro)realism and locality [1] which was considered as an unwanted consequence of quantum theory. True randomness, much undesired from the point of view of classical physics, serves as a valuable resource in many cryptographic protocols. It is for this reason that quantum random number generators (QRNGs) were one of the first commercially available devices utilizing basic principles of quantum physics in its elementary nature.

Towards the latter part of the 20th century it was recognized that quantum mechanics could lead another revolution and dramatically extend the premise of information processing. Classical notions of security underpinned by computational conditions were seriously threatened by the results of quantum-information processing and by the emergence of Shor's algorithm [2]. However, quantum mechanics offered a new security paradigm whereby the use of quantum states imparted unconditional secure communication through *quantum key distribution (QKD)* [3]. QKD protocols enable two communicating parties to produce a shared random secret key. The secret key can be used later to implement an unconditionally secure encryption [4].

The security of QKD has not only been established for an ideal noiseless experimental setting, it has also been proven robust within more realistic settings to the extent that QKD systems are now commercially available [5]. Interestingly, the robustness of QKD protocols has only been proven with respect to possible attacks on quantum data exchanged by the communicating parties with the assumption that a third party possesses knowledge of all exchanged classical data.

Sources of classical random bits, repeatedly used during different phases of quantum protocols, were implicitly considered perfect (unbiased). An assumption in the standard proofs of security [6–8] is that the source of random bits used in the protocol is unbiased and completely unaccessible to the adversary. Unfortunately, however, perfect (unbiased) randomness is very difficult to obtain in practice. All classical sources of random bits provide in fact rather pseudorandom bit strings, which might be fully accessible to the adversary together with knowledge of its preparation procedure and input bits. Even specialized QRNG devices produce weak (biased) randomness and require classical postprocessing [9], something one has to consider as accessible to the adversary. Real-world random number generators leak information via side channels and, thus, may be vulnerable to outside conditions (e.g., temperature, input power, EM radiation, etc.) which are potentially controlled by the adversary.

Although the problem of weak randomness has been broadly studied and is relatively well understood in classical-information processing [10–13], only a handful of results have been extended to the quantum domain (see, e.g., Refs. [14,15]). Recent investigations have shown that quantum-information processing can help to increase security of communication using weak randomness even for regions of parameters where purely classical processing would inevitably reveal all information to the adversary [16,17]. Here we show that on the contrary, weak randomness can negatively influence the security and correctness of existing quantum protocols.

In this paper we examine the security setting of QKD in which the adversary, aside from having full control of the quantum and classical channel, has some limited control over the sources of randomness that the communicating parties employ during the protocol (Fig. 1). We show that with an increasing key length, only negligible control of the randomness is necessary to render the QKD insecure. In particular, we demonstrate that the secret key individually held by communicating parties will differ significantly. Moreover, knowledge pertaining to the secret key held by the adversary will be comparable to the knowledge held by the receiving party. This is achieved by the fact that the adversary will be able to exclude a portion of exchanged qubits from testing by communicating parties.

II. WEAK SOURCES

Random processes are usually described by their probability distributions. However, it is insufficient to model a weak

*Present address: Física Teòrica: Informació i Fenòmens Quàntics, Departament de Física, Universitat Autònoma de Barcelona, E-08193 Bellaterra, Barcelona, Spain.

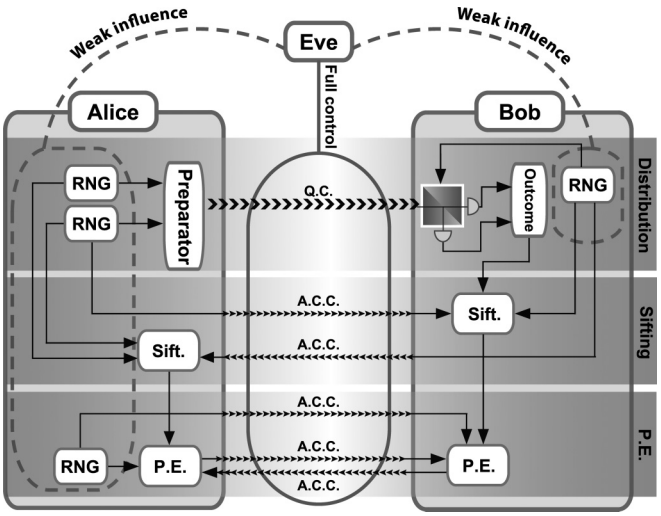

FIG. 1. A sketch version of the BB84 protocol. Eve has full access to both the quantum channel (Q.C.) and to authenticated classical channels (A.C.C.) and possesses partial access to random sources of Alice and Bob.

random source by a single probability distribution because the bias of the source is typically unknown. The only information usually known about the source is that it is random to a certain extent; thus, we allow the output of the weak random source to be distributed according to any probability distribution containing sufficient randomness. We quantify the amount of randomness of a distribution by the *min-entropy* of its source. The min-entropy of the random variable $\mathbf{X}$ is defined by

$$H_\infty(\mathbf{X}) = \min_{x\in X}[-\log_2 Pr(\mathbf{X}=x)]. \quad (1)$$

A nonuniform source of randomness is an (N,b) *source* if it emits N-bit strings drawn according to a probability distribution with a min-entropy of at least b bits. Thus, every specific N-bit sequence is drawn with a probability smaller than or equal to 2^{-b}. For $b = N$, one obtains a perfect source where all sequences are drawn with the same probability.

The bias of the source can be easily quantified by the *min-entropy loss* denoted $c = N - b$. A distribution is (N,b) *flat* if it is an (N,b) source and it is uniform on a subset of 2^b sample points; i.e., each string is output with a probability of either zero or 2^{-b}.

Min-entropy is a prominent measure of weak randomness and is used in many important papers (e.g., Ref. [12]), and its use is justified by its many desirable properties. Min-entropy is sufficiently general and most real-world sources can be described as min-entropy sources with sufficiently low min-entropy. In fact, it nicely models the most general information leak, since the drop of the min-entropy directly relates to the number of bits learned by the adversary. Last, but not least, it is also very convenient for calculations.

On the other hand it is fair to denote that even for a relatively small min-entropy decrease the adversary might get locally very strong information. In particular, the adversary is able to exclude some of the possible sequence completely, thus knowing with certainty that a specific sequence will never appear. This might help the adversary to design specific attacks utilizing this knowledge.

The quantity b/N is called the *min-entropy rate* and it achieves unity for perfect random sources that deliver one bit of entropy per bit produced. We are particularly interested in the *min-entropy loss rate*, which is denoted by quantity c/N. This quantity is (almost) zero for (almost) perfect random sources and approaches unity as the quality of the source decreases.

Throughout the paper the quality of the source used by communicating parties is the best, ultimate quality that is achievable for them. This includes any procedures they could potentially use to enhance their source(s) such as randomness extractors, as well as obtaining new sources such as direct use of their quantum source as QRNGs.

III. THE QKD PROTOCOL

Here we demonstrate the attack using a variation of the well-known BB84 protocol [3] which serves as a representative for the prepare-measure family of protocols. Note, however, that the same issue arises in entanglement-based protocols as well (when selecting a subset of systems to verify the entanglement).

A. Distribution phase

Using a random number generator Alice produces a $2n$-bit string X. Then depending on a $2n$-bit string from a random variable Y, Alice encodes each bit of X into a qubit from one of four possible states $\{|0\rangle,|1\rangle, |+\rangle,|-\rangle\}$. The state of the ith qubit is conditioned on the ith bits of both X and Y. In particular, each bit of X with value 0 is encoded into either $|0\rangle$ or $|+\rangle$ depending on whether the corresponding bit of Y is 0 or 1, respectively. A similar case holds for the bit 1 encoding into the states $|1\rangle$ and $|-\rangle$. Alice subsequently transmits all $2n$ qubits to Bob. In order to obtain information about the $2n$-bit string X, the adversary will be compelled to interact with these transmitted qubits which inevitably will lead to a disturbance in the transmitted sequences. Bob measures each received qubit to obtain a $2n$-bit string. Similar to the encoding procedure, a set of measurement bases are chosen according to a uniformly distributed random variable Z that outputs a $2n$-bit string. If the ith bit of Z has value 0, Bob measures in the computational basis, otherwise Bob measures in the diagonal basis.

B. Sifting phase

The sequence of measurement bases is revealed by Bob whereupon Alice then announces the locations of those qubits for which the corresponding preparation and measurement bases do not coincide. After discarding these qubits, Alice and Bob possess on average n-bit strings X_A and X_B. Following the sifting phase, the adversary has an estimate X_E of Alice's string X_A that depends on the degree to which the adversary interacted with the transmitted qubits. If there is no interaction then the adversary possesses no information on the n-bit string X_A. In the case of faultless quantum communication, X_A and X_B will be identical. However, in the case of the adversary choosing to interact with many qubits, the estimate X_E will be a good approximation to X_A, and this causes X_B to differ significantly from X_A.

C. Parameter estimation

The primary aim of parameter estimation is to approximate the number of errors between the n-bit strings X_A and X_B. The source of the errors may be attributed to a combination of quantum channel imperfections and eavesdropping by the adversary. However, in security proofs, one always considers the worst case scenario and, thus, assumes the adversary to be responsible for all errors.

Random sampling provides a way to estimate the number of errors between X_A and X_B. According to the output from a random variable T, Alice chooses a set of bit positions of X_A and assigns these as the test positions. Alice and Bob reveal the bit value in each test position. The number of errors t provides a reasonable estimate r on the actual number of errors in the remaining bits of X_A^R and X_B^R [7]. If the number of errors in the test positions is excessive then there is a high probability that the adversary is present and the protocol is aborted.

In any practical application one wants the test set to be relatively small in order to achieve a maximal possible key length. In existing QKD protocols, the size of the test set is typically on the order of $\sqrt{n}$ or $\log(n)$ [18–20]. In the following asymptotic analysis, we assume the most general case and post only a condition that the size of the test set is sublinear in n. In particular, we assume it is equal to $\Theta(n^{1-\alpha})$ with $0 < \alpha < 1$.

D. Information reconciliation and privacy amplification

Following parameter estimation, the bit strings X_A^R and X_B^R contain with a high probability up to r errors. The goal of the information reconciliation is to remove these errors even at the cost of revealing some information about X_A^R and X_B^R. This task is usually realized by one-way communication [7,8,20]. Such one-way information reconciliation can be implemented as long as Bob has more information than the adversary about Alice's string X_A^R [21].

The goal of the privacy amplification is to remove any knowledge possessed by Eve about the shared string X_A^R. A widely used method [22,23] is based on the random choice of a hashing function. In this case, Alice randomly chooses a hashing function f and sends it to Bob. The final shared key is $f(X_A^R) = f(X_B^R)$. Importantly, this method also uses one-way communication.

IV. THE ADVERSARY'S ATTACK

The use of uniform randomness is widespread throughout the various steps of the QKD protocol. The first instance of uniform randomness occurs during the distribution phase when Alice chooses $2n$-bit strings X and Y uniformly. Also, Bob must decide on a set of measurement bases which is again dependent upon a uniformly distributed random variable that outputs a $2n$-bit string. In the parameter estimation phase, a subset of the strings is chosen as a test set according to a uniformly distributed random variable T and, again, another source of random bits is used to select the hashing function. In light of these cases, we investigate a scenario in which Alice's randomness source—used to select the positions of test qubits—is biased.

This can be modeled by a scenario whereby the random variable T is distributed according to any $(n, n-c)$ distribution. We consider the worst-case approach (the worst distribution in the given range, where we attribute all randomness imperfections to the adversary) and assume that the adversary knows the actual distribution (what, e.g., represents the situation when the adversary learns c bits of information form a side channel). We assume that c is large enough to guarantee the existence of a distribution such that at least half of the qubits will not be tested. Later we calculate the required value of c.

Without the loss of generality, let us suppose that the first half of Bob's measurement outcomes will not be tested. The adversary can measure the first half of the $2n$ qubits in the $\{|0\rangle, |1\rangle\}$ basis. If Eve's measurement outcome is $|0\rangle$, she sends a state $|1\rangle$ to Bob, and if her measurement outcome is $|1\rangle$, she sends a state $|0\rangle$. Following this procedure and the sifting phase, the adversary has on average $n/2$ measurement outcomes. The adversary adds another $n/2$ bits chosen randomly and uniformly to obtain her estimate X_E of Alice's string X_A. Since Alice and Bob have not tested those bits measured by the adversary, the protocol will continue on to remaining phases.

We now quantify the amount of information that Bob and the adversary possess about Alice's n-bit string X_A. To obtain the result, we calculate the Hamming distance $D(A, B)$ between strings A and B. There are three cases to consider. First, the adversary may have measured a transmitted qubit in the correct basis. In such a case, the adversary obtains a bit value that coincides with the corresponding bit value in X_A with Bob then obtaining the bit complement. This happens on average in $n/4$ measurement cases. Second, it may happen that the adversary measures a transmitted qubit in an incorrect basis. Here both Bob and the adversary obtain the correct value with probability $1/2$. This happens on average in $n/4$ bits. The final situation to consider is the case in which the adversary does not perform a particular qubit measurement, which is the case in $n/2$ bits. The adversary then chooses random values for these bit positions and correctly guesses the value with probability $1/2$, giving a correct guess of $n/4$ positions. In this situation, Bob's measurement value is given by the measurement in the correct basis and, thus, he determines the value of Alice's bit with certainty.

The amount of information that Bob and the adversary possess about Alice's string X_A is given by $D(X_B, X_A)$ and $D(X_E, X_A)$, respectively. Both of these quantities are on average equal to $3n/8$. Consequently, the adversary and Bob possess on average the same level of knowledge about Alice's string. As the subsequent steps of the protocol demand that only Alice communicates information, it follows that, with the conclusion of the protocol, the adversary and Bob continue to share the same level of information about Alice's bit sting. This illustrates that ultimately there can be no privacy between Alice and Bob.

V. THE STRENGTH OF THE ADVERSARY

It remains for us to quantify how much information in terms of min-entropy loss the adversary requires in order to prevent parameter estimation on half of the bit positions. Alice needs $\log\binom{n}{n^{1-\alpha}}$ bits to specify $n^{1-\alpha}$ positions out of n. On the other

hand, the adversary wants Alice to choose the $n^{1-\alpha}$ test bits only from $n/2$ of the positions. Apparently, the best option for the adversary—in terms of the smallest entropy loss—is to set any $(\log(\binom{n}{n^{1-\alpha}}), \log(\binom{n/2}{n^{1-\alpha}}))$-flat distribution to Alice's random number generator. Such a distribution would uniformly select test bits only within the preselected half of all positions.

Of particular importance here is an analysis of the relative behavior of two quantities: the first quantity is the length of the test-bit string $N = \log(\binom{n}{n^{1-\alpha}})$, and the second quantity is the min-entropy loss $c = \log(\binom{n}{n^{1-\alpha}}) - \log(\binom{n/2}{n^{1-\alpha}})$. Both of these quantities diverge since Alice demands an increased level of randomness to choose the test bits from an ever increasing set size. Nevertheless, this is not the case for the min-entropy loss rate c/N expressing the fraction of total randomness required to restrict all possible test-bit positions within a prescribed subset of the total bit set. We show that the rate c/N, which is given as

$$\frac{c}{N} = \frac{\log\left(\binom{n}{n^{1-\alpha}}\right) - \log\left(\binom{n/2}{n^{1-\alpha}}\right)}{\log\left(\binom{n}{n^{1-\alpha}}\right)}, \tag{2}$$

drops with n.

We consider this expression in the limit of large n as most of the current security proofs for various QKD protocols have only been proven in the asymptotic regime of infinite key length. In evaluating the min-entropy loss rate c/N in the limit of large n, we make use of the Stirling approximation of the factorial function $\log(n!) = (n + 1/2)\log(n) - n$. Furthermore, we can approximate the quantity c as $n^{1-\alpha}\log(2) + O(\log(n))$ while the quantity N can be approximated to $n^{1-\alpha}\log(n) + O(n^{1-\alpha})$. The min-entropy loss rate c/N in the limit of large n can be evaluated as

$$\frac{c}{N} \approx \frac{1}{\log(n)}. \tag{3}$$

Under the assumption of perfect randomness, all QKD protocols have been proven to be perfectly secure in the limit of an infinitely large key size. However, implementing perfect randomness is difficult. By relaxing such an assumption to reflect real life conditions, Eq. (3) illustrates that QKD no longer remains robust. In particular, negligible control on the source of randomness renders QKD insecure.

VI. ENTANGLEMENT-BASED PROTOCOLS

In entanglement-based protocols [6,24], parties share entangled pairs of photons and employ monogamy of entanglement to build up security. A portion of these states is used to check the monogamy—and, thus, exclude the presence of an adversary—while the remaining states are used to perform the protocol itself. The test pairs are selected by a random source exactly in the same way as in the prepare-measure-based protocols. Having access to the random source of the selecting party, Eve might easily perform an attack where she could entangle herself to pairs not being tested in the future and, thereby, obtain information about the secret key.

VII. CONCLUSION

In this paper we demonstrated that if one allows an adversary limited access to random sources used by the communicating parties then both the security (Eve learns a significant amount of information about Alice's bit string) and the correctness (Alice and Bob end with different bit strings) of QKD protocols are completely compromised. This is the case for almost all known QKD protocols that use only a sublinear part of the data set to test for an adversary. In such instances, the adversary is able to restrict the test sample efficiently.

The obvious countermeasure against such an attack is to increase the number of test states to a significant linear portion of the raw key. This would, however, profoundly decrease the length of the secret key. Note also that an optional testing of the final bit strings of Alice and Bob would be subject to weak randomness in the selection phase as well and hardly can be more efficient than testing bits in the parameter estimation phase.

Another possible countermeasure is to use interactive error-correcting procedure with two way communication [25]. In such a scheme Alice and Bob should be able to see, that the number of errors is much higher than the estimate would have suggested. This would be a sign that something is wrong in the system and would lead to an interruption of the protocol. On the other hand these schemes use further randomness that, if not perfect, influences its results.

The most important message of this Paper is that one should consider the presence of weak randomness even within quantum information processing tools as a slight bias can totally jeopardize their functionality.

ACKNOWLEDGMENTS

J.B. acknowledges the support of the Czech Science Foundation GAČR Project No. P202/12/G061 as well as by the ERC Advanced Grant "IRQUAT". M.P., M.Pl., and C.W. acknowledge the support of the Czech Science Foundation GAČR Project P202/12/1142 as well as projects CE SAS QUTE and VEGA 2/0072/12. M.Pl. acknowledges the support of the SoMoPro Project funded under FP7 (People) Grant No. 229603 and by the South Moravian Region. C.W. acknowledges support from the Marie Curie Actions Programme.

[1] A. Einstein, B. Podolsky, and N. Rosen, Phys. Rev. **47**, 777 (1935).
[2] P. W. Shor, SIAM J. Comput. **26**, 1484 (1997).
[3] C. H. Bennett and G. Brassard, in *Proceedings of the IEEE International Conference on Computers, Systems and Signal Processing*, Bangalore, India, 1984 (IEEE, New York, 1984), pp. 175–179; IBM Tech. Discl. Bull. **28**, 3153 (1985).
[4] G. S. Vernam, Trans. Am. Inst. Electr. Eng. **45**, 295 (1926).
[5] See, e.g., http://www.idquantique.com/true-random-number-generator/products-overview.html
[6] H.-K. Lo and H. F. Chau, Science **283**, 2050 (1999).
[7] D. Mayers, J. Assoc. Comput. Mach. **48**, 351 (2001).
[8] P. W. Shor and J. Preskill, Phys. Rev. Lett. **85**, 441 (2000).

[9] R. Solcà, Master's thesis, ETH Zürich, 2010.
[10] Y. Dodis, Jin Ong Shien, M. Prabhakaran, and A. Sahai, in *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science* (IEEE Computer Society, Washington, DC, USA, 2004), pp. 196–205.
[11] U. Maurer and S. Wolf, Lect. Notes Comput. Sci. **1294**, 307 (1997).
[12] J. L. McInnes and B. Pinkas, Lect. Notes Comput. Sci. **537**, 421 (1991).
[13] R. Renner and S. Wolf, Lect. Notes Comput. Sci. **2729**, 78 (2003).
[14] H. Zbinden, in *Proceedings of the 10th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, 2008. SYNASC '08 (IEEE Computer Society, Los Alamitos, CA, 2009), p. 19.
[15] X.-B. Wang, arXiv:quant-ph/0405182.
[16] J. Bouda, M. Pivoluska, and M. Plesch, Quantum Inf. Comput. **12**, 0395 (2012).
[17] J. Bouda, M. Pivoluska, M. Plesch, and C. Wilmott (unpublished).
[18] M. Christandl, R. Renner, and A. Ekert, arXiv:quant-ph/0402131.
[19] K. Horodecki, M. Horodecki, P. Horodecki, D. Leung, and J. Oppenheim, IEEE Trans. Inf. Theory **54**, 2604 (2008).
[20] H.-K. Lo, H. F. Chau, and M. Ardehali, J. Cryptology **18**, 133 (2005).
[21] I. Csiszar and J. Korner, IEEE Trans. Inf. Theory **24**, 339 (1978).
[22] C. Bennett, G. Brassard, C. Crepeau, and U. Maurer, IEEE Trans. Inf. Theory **41**, 1915 (1995).
[23] R. Renner and R. König, Lect. Notes Comput. Sci. **3378**, 407 (2005).
[24] A. K. Ekert, Phys. Rev. Lett. **67**, 661 (1991).
[25] G. Brassard and L. Salvail, Lect. Notes Comput. Sci. **765**, 410 (1994).