Universally composable and customizable post-processing for practical quantum key distribution

Xiongfeng Ma[a], Chi-Hang Fred Fung[b,*], Jean-Christian Boileau[c], H. F. Chau[b]

[a] *Institute for Quantum Computing and Department of Physics and Astronomy, University of Waterloo, 200 University Ave W., Waterloo, ON, Canada N2L 3G1*
[b] *Department of Physics and Center of Theoretical and Computational Physics, University of Hong Kong, Pokfulam Road, Hong Kong*
[c] *Center for Quantum Information and Quantum Control, Department of Electrical & Computer Engineering and Department of Physics, University of Toronto, Toronto, Ontario, Canada, M5S 1A7*

Abstract

In quantum key distribution (QKD), a secret key is generated between two distant parties by transmitting quantum states. Experimental measurements on the quantum states are then transformed to a secret key by classical post-processing. Here, we propose a construction framework in which QKD classical post-processing can be custom made. Though seemingly obvious, the concept of concatenating classical blocks to form a whole procedure does not automatically apply to the formation of a quantum cryptographic procedure since the security of the entire QKD procedure rests on the laws of quantum mechanics and classical blocks are originally designed and characterized without regard to any properties of these laws. Nevertheless, we justify such

*Corresponding author. Department of Physics, University of Hong Kong, Pokfulam Road, Hong Kong

Email addresses: xfma@iqc.ca (Xiongfeng Ma), chffung@hkucc.hku.hk (Chi-Hang Fred Fung)

concept of concatenating classical blocks in constructing QKD classical post-processing procedures, along with a relation to the universal-composability security parameter. Consequently, effects arising from an actual QKD experiment, such as those due to the finiteness of the number of signals used, can be dealt with by employing suitable post-processing blocks. Lastly, we use our proposed customizable framework to build a comprehensive generic recipe for classical post-processing that one can follow to derive a secret key from the measurement outcomes in an actual experiment.
Keywords:
quantum cryptography, key distribution, post-processing procedure, security quantification, universally composable security, IT security

1. Introduction

Quantum key distribution (QKD) (Bennett and Brassard (1984); Ekert (1991)) allows two distant users to generate a secret key by wisely exploiting properties of quantum mechanics. Initial work on QKD has been focused on the investigation of its security and a few QKD protocols, such as the well-known BB84 protocol (Bennett and Brassard (1984)), have been proven to be secure in the last decade (Mayers (2001); Lo and Chau (1999); Shor and Preskill (2000)). Meanwhile, many QKD experiments have been performed (see, e.g., references in Gisin et al. (2002); Lo and Lütkenhaus (2007)).

In general, a QKD experiment involves a quantum state transmission step (where quantum states are transmitted and measured) and a classical post-processing step (where the measurement outcomes are processed classically with the help of classical communication to generate a final secret key).

Although standard security proofs (such as Shor and Preskill (2000)) imply a procedure for distilling a final secret key from measurement outcomes, such procedure cannot be directly carried out in an *actual* QKD experiment because many of the security proofs (i) highlight only essential operations that need to be done instead of explicitly listing every step in detail and (ii) focus on the case that the key is arbitrarily long. In practice, we need extra operations to support the basic ones, and different system designers may choose different sets of operations to suit the needs of their systems and situations. Also, due to the finite lengths of practical keys, all these operations will only succeed probabilistically in carrying out their intended functions.

Recently, significant efforts have been made to study the finite-length effect in QKD post-processing. Mayers (1996, 2001) first gave a security proof for the finite-key case and provided a key rate lower bound. Hayashi (2006) derived a bound on the eavesdropper's information under the finite-key case and gave a higher key rate lower bound compared to Mayers'. Scarani and Renner (2008a,b) derived security bounds in an information-theoretic approach, and Cai and Scarani (2009) further analyzed the finite-key security of the BB84 and BBM92 protocols with the same approach. Our work (Fung et al. (2010)) provided a detailed finite-key analysis of a practical QKD post-processing procedure for the BB84 protocol using a entanglement-distillation-protocol-based approach. In all the previous works, the post-processing steps of a given analysis are quite fixed, without much flexibility for change. In this paper, we discuss the idea of a framework in which the post-processing elements can be added or modified easily. We remark that a tailor-made

post-processing procedure is highly desirable for an actual QKD experiment, since various post-processing steps may be employed to deal with realistic effects (such as the finite-length effect in realistic situations) and different system designers may have different requirements.

We propose a framework for building customized classical post-processing procedures for QKD and quantifying the associated security. While it is easy to form any post-processing procedure (simply by concatenating operational blocks), it may not be clear how to quantify the security of the resultant procedure as a whole. It is obvious that adding up the failure probabilities of the blocks gives us the overall failure probability of the whole post-processing procedure. However, while this classical probability is a meaningful measure for the post-procedure procedure, it is not clear whether it is also a meaningful one for the final key generated by this QKD process. We emphasize that a quantification for the post-processing procedure does not necessarily become a quantification for the security of the key. In fact, a meaningful and widely adopted security measure, the universal composability security (Ben-Or et al. (2005); Renner and König (2005)), has no apparent connection with the aforementioned quantification for the post-processing procedure. Nevertheless, as we discuss below, the classical failure probability of the procedure can be related to the composability security of the key. This connection justifies that the constituent blocks of a procedure in our framework can be independently characterized by a failure probability through which each block affects the composability security of the final key.

The major point of this paper is that we advocate a construction method for post-processing procedures for QKD experiments that is general and cus-

tomizable. The experiment designers have the freedom to select the suitable classical post-processing blocks and use them directly in a QKD procedure — hence, customizable post-processing. While such a concept of concatenation is not new, especially in the classical processing domain, what is new is that we show that such concatenation of classical blocks can also be directly applied in the quantum setting in the same sense as in the classical setting. Such carrying over to the quantum domain is not automatic since it is possible that it can break the security of the whole QKD procedure as the security is now related to the purity of quantum states (which are completely out of the scope of any classical processing). In spite of this, we show that the security characteristic of the QKD procedure can still be preserved, and furthermore we can even quantify the security of the final key in the universal composability sense. We note that the concatenation of classical blocks referred here is not about classical cryptographic composability since these classical blocks (e.g., error correction) can be unrelated to cryptography at all.

In the following, we discuss our framework and use it to construct a post-processing procedure. This comprehensive procedure is designed to be directly used as a recipe in practical QKD systems with single- or entangled-photon sources, taking into account the finiteness of the number of signals used. This procedure also serves as a stepping stone for a QKD standard.

2. Security measure

Due to the finite-size effect, a secret key generated by a QKD system may not be perfect in the sense that Alice and Bob do not share the same key

and/or, Eve, an eavesdropper, possesses some information about the key. Nevertheless, the fact that the key is imperfect does not preclude it from being used in a subsequent task requiring a perfect key. In fact, if one can assign a probability that the key can be regarded as an ideal one, the use of the nonideal key as an ideal one can be justified. Indeed, this notion of security is captured by the widely adopted notion of the composability-security definition of QKD (Ben-Or et al. (2005); Renner and König (2005)). A security definition of QKD is composable in the sense that the final key generated is statistically indistinguishable from an ideal secret key except with a small probability (quantified by a security parameter). Thus, the secret key generated in one round of QKD can be used in another cryptographic function (including another round of QKD) where an ideal key is expected. Each function composed in this manner contributes its own security parameter to the overall one linearly. This linear dependence is an important feature of the composability security definition.

We adopt the following definition of composable security.

Definition 1 (König et al. (2007); Renner and König (2005); Renner (2005))**.** A classical random variable K (representing the key) drawn from the set $\mathcal{K}$ is said to be ζ-secure with respect to an eavesdropper holding a quantum system E if

$$\frac{1}{2}\operatorname{Tr}|\rho_{KE} - \rho_U \otimes \rho_E| \leq \zeta \tag{1}$$

where $\rho_{KE} = \sum_{k\in\mathcal{K}} P_K(k)|k\rangle\langle k| \otimes \rho_{E|K=k}$ is the state of the systems K and E, $P_K(k)$ is the probability of having $K = k$, $\rho_U = \sum_{k\in\mathcal{K}} |k\rangle\langle k|/|\mathcal{K}|$ represents an ideal key taking values uniformly over $\mathcal{K}$, and $|\mathcal{K}|$ is the size of $\mathcal{K}$. Here, $\operatorname{Tr}|A| = \sum_i |\lambda_i|$ where λ_i's are the eigenvalues of A.

This definition has the operational meaning that the real situation using the real key K is the same as the ideal situation using the ideal key U except with probability ζ.

The composable security parameter ζ in the above definition is an important value for the final secret key at the end of the distillation and thus it is important to link the classical post-processing steps to this value. One difficulty in doing so is that composability deals with the quantum picture of the key and environment, which is in stark contrast to the pure classical nature of post-processing. Another difficulty stems from the flexibility of the classical post-processing procedure. QKD systems may differ and system designers may customize the procedure according to the system's need or their own taste. Thus, a framework for building procedures with great flexibility is proposed here.

3. Customizable framework

The entire post-processing procedure is composed of blocks, which are freely chosen by the system designer. Minimal classical post-processing as suggested by QKD security proofs (e.g. Lo and Chau (1999); Shor and Preskill (2000)) performs two basic functions: error correction and privacy amplification. However, in practice, more functions need to be performed to support these basic functions (e.g., authentication on the classical channel for assisting error correction). Consequently, systems designers may select various functional blocks to form their own post-processing procedure. For example, an error correction block that is based on convolutional codes can be substituted by another one based on low density parity check codes.

In order to facilitate an easy linkage with the final composable security parameter in Definition 1, we propose a universal language for describing the classical functional blocks. The universal language we advocate is the probability that a classical functional block fails to perform its intended function. We call this the failure probability of the block, and an example of a failure event is when an error correction block fails to detect or correct bit errors.

We are concerned with the failure probability of the whole post-processing procedure. This probability is upper bounded by the summation of the failure probabilities of individual blocks (via the union bound). The important feature is that each block can function independently of the others and also its failure probability can be characterized individually. Therefore, we can select a block from existing literature and employ it directly in a post-processing procedure.

We have been talking about characterizing the entire post-processing procedure with a failure probability, which is a classical measure. On the other hand, the ultimate concern is the security parameter (ζ in Definition 1) of the final key generated in the QKD process. We have shown that the connection between these two quantities with the following simple relation (Fung et al. (2010)):

Lemma 1. When the failure probability of the post-processing procedure is ε, the final key is secure with respect to universal composability with a security parameter $\sqrt{\varepsilon(2-\varepsilon)}$.

This means that the key generated by a post-processing procedure that fails with probability ε is $\sqrt{\varepsilon(2-\varepsilon)}$ secure in accordance with Definition 1.

The reason that the post-processing procedure with failure probability ε does not give rise to an ε secure key is because the security definition in Definition 1 is concerned with the whole quantum picture consisting of the key and the eavesdropper's quantum system while the failure probability is concerned with the post-processing procedure which is completely classical.

4. A practical post-processing recipe

To demonstrate the concept of our customizable framework, we construct a practical post-processing procedure assembled from blocks, some of which are taken from existing literature. Specifically, we assemble a post-processing procedure using an authentication scheme by Krawczyk (1994), privacy amplification based on universal hashing (Wegman and Carter (1979, 1981)), standard forward error correction (see, e.g., Cover and Thomas (2006)), and our own analysis on random sampling (Fung et al. (2010)). This post-processing procedure is directly applicable to realistic experiments for the BB84 protocol with a single or entangled photon source.

Let us start by examining the underlying assumptions used here. We emphasize that in order to apply the scheme to a QKD system, one needs to compare these assumptions with the real setup. The assumptions used in the paper are listed as follows:

1. Alice and Bob perform the BB84 protocol with a perfect single photon source or a basis-independent (entangled) photon source (Koashi and Preskill (2003); Ma et al. (2007)).

2. The detection system is compatible with the squashing model (Tsurumaru and Tamaki (2008); Beaudry et al. (2008); see also, Koashi

(2006a)). For example, detection efficiency mismatch is not considered here (Fung et al. (2009)).

3. Alice and Bob use perfect random number generators and perfect key management. They share a certain amount of secure key prior to running their QKD system.

The post-processing scheme is based on a modified Shor-Prekill's security proof (Shor and Preskill (2000)), which is essentially Koashi's complimentary argument (Koashi (2006b)). In this approach, the secure key generation is equivalent to an entanglement distillation protocol, which involves bit and phase error corrections. In the post-processing, the bit error correction becomes classical error correction and the phase error correction becomes privacy amplification. We remark that our framework is applicable to any physical QKD implementations that comply with the above assumptions, and it does not depend on the implementation details.

The procedure serves as a guideline for QKD data post-processing. We start from raw data from measurements and some pre-shared secure key bits, and produce a composable secret key. The details of the procedure are presented in Fung et al. (2010).

The secure key used in the post-processing comes from a pre-shared secure key between Alice and Bob. For each step, we investigate the secure-key cost, k_{xx}, and the failure probability, ε_{xx}, where xx denotes the name of a step.

The post-processing procedure is listed as follows. Note that none of following classical communication is encrypted unless otherwise stated.

1. Key sift [not authenticated]: Bob discards no-click events and obtains

n-bit raw key by randomly assigning (Lütkenhaus (1999)) the double clicks [1]. Note that other key sift procedures might be applied as well, see for example, Ma et al. (2008).

2. Basis sift [authenticated]: Alice and Bob send each other n-bit basis information. Due to the symmetry, we can assume they pick up the same failure probability for this procedure (Krawczyk (1994))

$$\varepsilon_{bs} = n2^{-k_{bs}+1} \tag{2}$$

Here, Alice and Bob use a $2k_{bs}$-bit secure key to construct a Toeplitz matrix with a size of $(n \times k_{bs})$ by a LFSR. The authenticated tag is generated by multiplying the matrix and the message. Then they encrypt the two tags by two k_{bs}-bit secure keys. Since the tags are encrypted by a one-time pad, the $2k_{bs}$-bit key used for the Toeplitz matrix construction is still private. Hence, the total secure-key cost in this step is $2k_{bs}$ and the corresponding failure probability is $2\varepsilon_{bs}$. Note that when Alice and Bob use a biased basis choice (Lo et al. (2005)), they can exchange less than n-bit classical information for basis sift by data compression. Since the secure-key cost only logarithmically depends on the length of the message, we simply use n for the following discussion. In the end of this step, Alice and Bob obtain n_x (n_z)-bit sifted key in X (Z) basis. Define the bias ratio to be $q_x \equiv n_x/(n_x+n_z)$.

3. Error correction [not authenticated but encrypted[2]]: the secure-key

[1] In the case of a passive-basis-selection setup, Bob also randomly assigns basis value X or Z for double clicks (Beaudry et al. (2008)).

[2] The error correction step may be done without encryption using other security proof

cost is given by

$$k_{ec} = n_x f(e_{bx}) H(e_{bx}) + n_z f(e_{bz}) H(e_{bz}) \tag{3}$$

where $f(x)$ is the error correction efficiency and $H(x) = -x \log_2(x) - (1-x)\log_2(1-x)$ is the binary entropy function. In practice, Alice and Bob only need to count the amount of classical communication used in the error correction. That is, the value of k_{ec} can be directly obtained from the post-processing. After the error correction, Alice and Bob count the number of errors in X (Z) basis: $e_{bx}n_x$ ($e_{bz}n_z$).

4. Error verification: Alice and Bob want to make sure (with a high probability) that their keys after the error correction step are identical. Note that the idea of using error verification to replace error testing is proposed by Lütkenhaus (1999).

   In this procedure, Alice sends an encrypted tag of an authentication scheme to Bob. Here, the message to the authentication is the key after error correction. If the tag passes Bob's verification, Alice and Bob share the same key except for a small failure probability,

   $$\varepsilon_{ev} = (n_x + n_z) 2^{-k_{ev}+1}, \tag{4}$$

   where k_{ev} is the secure key cost in this step. We remark that when error verification fails, Alice and Bob can go back to the error correction step.

5. Phase error rate estimation [no communication]: Alice and Bob can estimate the phase error rates in X and Z bases, e_{px} and e_{pz} separately.

techniques. In this case, there may be some restriction on the error correction procedure and more privacy amplification may be required.

Take the qubits measured in the Z-basis as an example. Since their phase error rate corresponds to the X-basis measurements, we apply a random sampling argument to infer it from the qubits measured in the X-basis. According to our random sampling argument (Fung et al. (2010)), the probability of $e_{pz} > e_{bx} + \theta_x$ denoted by $P_{\theta x}$ is bounded by

$$P_{\theta x} < \frac{\sqrt{n_x + n_z}}{\sqrt{n_x n_z e_{bx}(1 - e_{bx})}} 2^{-(n_x+n_z)\xi_x(\theta_x)}, \tag{5}$$

where $\xi_x(\theta_x)$ is defined by $\xi_x(\theta_x) \equiv H(e_{bx} + \theta_x - q_x\theta_x) - q_x H(e_{bx}) - (1 - q_x)H(e_{bx} + \theta_x)$ with $q_x = n_x/(n_x + n_z)$. A similar formula for $P_{\theta z}$ can also be derived. Then the total failure probability of phase error rate estimation, ε_{ph}, is given by

$$\varepsilon_{ph} \leq P_{\theta x} + P_{\theta z}. \tag{6}$$

In the case when $e_{bx} = 0$ ($e_{bz} = 0$), one can replace it by $n_x e_{bx} = 1$ ($n_z e_{bz} = 1$) to get around the singularity (Fung et al. (2010)). One can see that $\xi_x(\theta_x)$ is positive when $\theta_x > 0$ and $0 \leq e_{bx}, e_{bx} + \theta_x \leq 1$, due to concavity of the binary entropy function $H(x)$. Note that in the limit of a large n, θ can be chosen small. In this case, (6) yields a similar result used in the literature, such as Shor and Preskill (2000); Ma et al. (2007).

6. Privacy amplification [authenticated]: Alice generates an $(n_x + n_z + l - 1)$-bit random bit string and sends it to Bob through an authenticated channel. They use this random bit string to generate a Toeplitz matrix. The final key (with a size of l) will be the product of this matrix (with a size of $(n_x + n_z) \times l$) and the key string (with a size of $n_x + n_z$). The

failure probability of the privacy amplification is given by

$$\varepsilon_{pa} = (n_x + n_z + l - 1)2^{-k_{pa}+1} + 2^{-t_{oe}}, \tag{7}$$

where k_{pa} is the secure-key cost for the authentication and t_{oe} is defined by

$$l = n_x[1 - H(e_{bz} + \theta_z)] + n_z[1 - H(e_{bx} + \theta_x)] - t_{oe}, \tag{8}$$

The first term in (7) gives the failure probability of the authentication for the $(n_x + n_z + l - 1)$-bit random bit string transmission. The second term in (7) gives the failure probability of the privacy amplification[3].

7. The final secure key length (net growth[4]) is given by

$$NR \geq l - 2k_{bs} - k_{ec} - k_{ev} - k_{pa} \tag{9}$$

with a failure probability of

$$\varepsilon \leq 2\varepsilon_{bs} + \varepsilon_{ev} + \varepsilon_{ph} + \varepsilon_{pa}, \tag{10}$$

where l is given by (8).

5. Parameter optimization

In order to maximize the final secure key length in the post-processing, Alice and Bob need to consider the failure probabilities from all steps and the

[3]In the equivalent entanglement distillation protocol used for the security proof (Shor and Preskill (2000); Koashi (2006b)), the second term in (7) gives the failure probability of the phase error correction.

[4]Since QKD is a key expansion process, it requires some pre-shared secret bits to start with and thus they have to be accounted for when calculating the final key length.

corresponding secure-key costs. That is, they need to optimize the key rate, equation (9), subject to (10). The parameters to be optimized are: bias ratio q_x, various secure-key costs (k_{bs}, k_{ec}, k_{ev}, k_{pa}, t_{oe}) and security parameters (ε_{bs}, ε_{ev}, ε_{ph}, ε_{pa}).

In practice, Alice and Bob can calibrate the QKD system to get an estimate of the transmittance η, the error rates e_{bx} and e_{bz}. Through some rough calculation of the target length of the final key, they decide the acceptable confidence interval $1-\varepsilon$ and fix the length of the experiment, N, the number pulses sent by Alice. Then roughly, the length of the raw key is $n = N\eta$. Thus, in the optimization procedure, the given values (constraints) are ε, n, e_{bx} and e_{bz}.

The failure probability ε is chosen by Alice and Bob according to the later practical use of the final key. The relation between this failure probability to the universal composability security definition (Ben-Or et al. (2005); Renner and König (2005)) is given by Lemma 1 (see also Fung et al. (2010)), i.e., the key is $\sqrt{\varepsilon(2-\varepsilon)}$ secure in accordance with Definition 1. For instance, suppose Alice and Bob plan to use a QKD system for a million times in the manner that the secret key output of one round is fed as input to the next, and set the target failure probability to be ε for each round. Then the security parameter for the key from the last round is $10^6\sqrt{\varepsilon(2-\varepsilon)}$, which should be below some threshold depending on the message security level. From here, one can see that the choice of ε is not strictly pre-determined. That is, the final security parameter, ε, can slightly deviate from the pre-determined one.

Denote the probability for Alice and Bob to choose X basis to be p_x. After the basis sift, Alice and Bob share an n_x-bit (n_z-bit) key in X (Z)

basis, where roughly (due to fluctuations) $n_x \approx p_x^2 n$ and $n_z \approx (1-p_x)^2 n$. Thus the bias ratio is given by $q_x \approx p_x^2/[p_x^2 + (1-p_x)^2]$. In a realistic case, Alice and Bob can optimize p_x first, and then optimize other parameters after the error verification part when the real values of n_x, n_z e_{bx} and e_{bz} are fixed (and known to them).

The error correction and phase error rate estimation mainly depend on the bias ratio. Thus, Alice and Bob can group the failure probabilities and secure key costs into two parts by defining $\varepsilon_3 \equiv 2\varepsilon_{bs} + \varepsilon_{ev} + \varepsilon_{pa}$ and $k_3 \equiv 2k_{bs} + k_{ev} + k_{pa} + t_{oe}$, see (8), (9) and (10). The final secure key length can be rewritten as

$$NR \geq n_x[1-f(e_{bx})H(e_{bx})-H(e_{bz}+\theta_z)]+n_z[1-f(e_{bz})H(e_{bz})-H(e_{bx}+\theta_x)]-k_3. \tag{11}$$

We remark that if the contribution from one basis is negative in (11), Alice and Bob should use the detections from this basis for the parameter estimation only, but not the key generation.

The optimized secure-key cost for each step is given by the following (see Fung et al. (2010)):

$$t_{oe} = \tfrac{k_3}{5} - \tfrac{4}{5} - \tfrac{1}{5}\log_2 A \tag{12}$$

$$k_{bs} = t_{oe} + 1 + \log_2 n \tag{13}$$

$$k_{ev} = t_{oe} + 1 + \log_2(n_x + n_z) \tag{14}$$

$$k_{pa} = t_{oe} + 1 + \log_2(n_x + n_z + l - 1), \tag{15}$$

where $A = n^2(n_x+n_z)(n_x+n_z+l-1)$. The corresponding failure probability is

$$\varepsilon_3 = 5A^{1/5}2^{-(k_3-4)/5}. \tag{16}$$

When the final key length is much larger than 37 bits, Alice and Bob can set

$$k_3 = -5\log_2\varepsilon + 4\log_2 n + 50 \tag{17}$$

and the corresponding failure probability is $\varepsilon_3 < 10^{-2}\varepsilon$. Since Alice and Bob will recalculate the failure probability in the end and allow the final ε to have a small deviation from the pre-determined value, they can safely use $\varepsilon_{ph} = \varepsilon$ in the optimization. Thus, the simplified optimization problem only has three parameters to optimize: q_x, θ_x and θ_z, given $\varepsilon_{ph} = \varepsilon - \varepsilon_3 \approx \varepsilon$.

Observation. *The main effect of the finite key analysis for the QKD post-processing stems from the phase error rate estimation. Inefficiencies due to authentication, error verification, and privacy amplification are relatively insignificant.*

This can be easily seen from (16) and (17). Even in an extreme case that $\varepsilon = 10^{-30}$ and $n = 10^{30}$, the secure key cost of all the parts other than the phase error rate estimation, given by (17), is 947 bits ($\ll n$) and its corresponding failure probability $\varepsilon_3 < 10^{-32}$.

6. Conclusion

We present a framework for building customizable post-processing schemes and use it to form a practical scheme with some of its constituent blocks borrowed from existing literature. The power of our framework lies in that fact that it facilitates a modular design of post-processing procedures and quantifies the final key with a composability-security parameter. We apply an authentication scheme for the error verification and derive a strict bound

for the phase error estimation. Furthermore, we investigate the efficiency of privacy amplification. Finally, we also study parameter optimization.

Acknowledgments

We thank C. Erven, N. Godbout, M. Hayashi, D. W. Leung, H.-K. Lo, N. Lütkenhaus, M. Koashi, X. Mo, B. Qi, R. Renner, V. Scarani, D. Stebila, K. Tamaki, W. Tittel, Q. Wang, Y. Zhao and other participants to the workshop *Quantum Works QKD Meeting (Waterloo, Canada)* and *Finite Size Effects in QKD (Singapore)* for enlightening discussions. X. Ma especially thanks H. F. Chau for hospitality and support during his visit at the University of Hong Kong. This work is supported from the NSERC, the OCE, and the RGC grant No. HKU 701007P of the HKSAR Government.

References

Beaudry, N. J., Moroder, T., Lütkenhaus, N., 2008. Squashing models for optical measurements in quantum communication. Phys. Rev. Lett. 101, 093601.

Ben-Or, M., Horodecki, M., Leung, D. W., Mayers, D., Oppenheim, J., 2005. The universal composable security of quantum key distribution. In: Second Theory of Cryptography Conference TCC 2005, Lecture Notes in Computer Science. Vol. 3378. Springer-Verlag, pp. 386–406.

Bennett, C. H., Brassard, G., 1984. Quantum cryptography: Public key distribution and coin tossing. In: Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing. IEEE, New York, Bangalore, India, pp. 175–179.

Cai, R. Y., Scarani, V., april 2009. Finite-key analysis for practical implementations of quantum key distribution. New J. Phys. 11, 045024.

Cover, T. M., Thomas, J. A., 2006. Elements of Information Theory, 2nd Edition. Wiley-Interscience, New York.

Ekert, A. K., 1991. Quantum cryptography based on Bell's theorem. Phys. Rev. Lett. 67, 661.

Fung, C.-H. F., Ma, X., Chau, H. F., Jan 2010. Practical issues in quantum-key-distribution postprocessing. Phys. Rev. A 81 (1), 012318.

Fung, C.-H. F., Tamaki, K., Qi, B., Lo, H.-K., Ma, X., 2009. Security proof of quantum key distribution with detection efficiency mismatch. Quant. Inf. Comput. 9, 0131.

Gisin, N., Ribordy, G., Tittel, W., Zbinden, H., Jan. 2002. Quantum cryptography. Rev. Mod. Phys. 74, 145–195.

Hayashi, M., 2006. Practical evaluation of security for quantum key distribution. Phys. Rev. A 74 (2), 022307.

Koashi, M., 2006a. Efficient quantum key distribution with practical sources and detectors. arXiv:quant-ph/0609180.

Koashi, M., 2006b. Unconditional security proof of quantum key distribution and the uncertainty principle. J. Phys. Conf. Ser. 36, 98.

Koashi, M., Preskill, J., 2003. Secure quantum key distribution with an uncharacterized source. Phys. Rev. Lett. 90, 057902.

König, R., Renner, R., Bariska, A., Maurer, U., april 2007. Small accessible quantum information does not imply security. Phys. Rev. Lett. 98, 140502.

Krawczyk, H., 1994. LFSR-based hashing and authentication. In: Advances in Cryptology - CRYPTO'94, Lecture Notes in Computer Science. Vol. 893. Springer-Verlag, pp. 129–139.

Lo, H.-K., Chau, H. F., 1999. Unconditional security of quantum key distribution over arbitrarily long distances. Science 283, 2050.

Lo, H.-K., Chau, H. F., Ardehali, M., 2005. Efficient quantum key distribution scheme and a proof of its unconditional security. J. Crypto. 18 (2), 133–165.

Lo, H.-K., Lütkenhaus, N., 2007. Quantum cryptography: from theory to practice. Phys. Canada 63, 191.

Lütkenhaus, N., May 1999. Estimates for practical quantum cryptography. Phys. Rev. A 59 (5), 3301–3319.

Ma, X., Fung, C.-H. F., Lo, H.-K., 2007. Quantum key distribution with entangled photon sources. Phys. Rev. A 76, 012307.

Ma, X., Moroder, T., Lütkenhaus, N., 2008. Quantum key distribution secure against the efficiency loophole. arXiv:0812.4301.

Mayers, D., 1996. Quantum key distribution and string oblivious transfer in noisy channels. In: Advances in Crytology-Crypto '96, Lecture Notes in Computer Science. Vol. 1109. Springer, Berlin, pp. 343–357.

Mayers, D., May 2001. Unconditional security in quantum cryptography. J. ACM 48 (3), 351406.

Renner, R., 2005. Security of quantum key distribution. Ph.D. thesis, Swiss Federal Institute of Technology, also available in Int. J. Quant. Inf. **6**, 1 (2008).

Renner, R., König, R., 2005. Universally composable privacy amplification against quantum adversaries. In: Second Theory of Cryptography Conference TCC 2005, Lecture Notes in Computer Science. Vol. 3378. Springer-Verlag, pp. 407–425.

Scarani, V., Renner, R., 2008a. Quantum cryptography with finite resources: Unconditional security bound for discrete-variable protocols with one-way postprocessing. Phys. Rev. Lett. 100 (20), 200501.

Scarani, V., Renner, R., 2008b. Security bounds for quantum cryptography with finite resources. Lecture Notes in Computer Science 5106, 83–95.

Shor, P. W., Preskill, J., July 2000. Simple proof of security of the BB84 quantum key distribution protocol. Phys. Rev. Lett. 85 (2), 441.

Tsurumaru, T., Tamaki, K., 2008. Security proof for QKD systems with threshold detectors. Phys. Rev. A 78, 032302.

Wegman, M. N., Carter, J. L., 1979. Universal classes of hash functions. Journal of Computer and System Sciences 18, 143–154.

Wegman, M. N., Carter, J. L., 1981. New hash functions and their use in

authentication and set equality. Journal of Computer and System Sciences 22, 265–279.