npj | Quantum Information

www.nature.com/npjqi

ARTICLE OPEN

Check for updates

Guessing probability in quantum key distribution

Xiang-Bin Wang[1,2,3✉], Jing-Tao Wang[1✉], Ji-Qian Qin[1], Cong Jiang[1] and Zong-Wen Yu[1,4✉]

On the basis of the existing trace distance result, we present a simple and efficient method to tighten the upper bound of the guessing probability. The guessing probability of the final key $\mathbf{k}$ can be upper bounded by the guessing probability of another key $\mathbf{k}'$, if $\mathbf{k}'$ can be mapped from the final key $\mathbf{k}$. Compared with the known methods, our result is more tightened by thousands of orders of magnitude. For example, given a 10^{-9}-secure key from the sifted key, the upper bound of the guessing probability obtained using our method is 2×10^{-3277}. This value is smaller than the existing result 10^{-9} by more than 3000 orders of magnitude. Our result shows that from the perspective of guessing probability, the performance of the existing trace distance security is actually much better than what was assumed in the past.

npj Quantum Information (2020)6:45 ; https://doi.org/10.1038/s41534-020-0267-3

INTRODUCTION

The first quantum key distribution (QKD) protocol has been proposed by Bennett and Brassard in 1984; the protocol was based on the fundamentals of quantum mechanics[1]. Since then, the security of QKD has always been the central issue in the quantum cryptographic field[2]. Trace distance is a very important security criterion[3,4]. It provides the universal composable security[5,6], which can guarantee the security of key regardless of its application such as one-time pad (OTP). This is why many studies choose trace distance for the security criterion[3,4,7,8].

In a classical practical cryptosystem, the impact of guessing probability on security is very important[9,10]. Specifically, the key generated by the QKD protocol is not based on the presumed hardness of mathematical problems; thus, the eavesdropper Eve can only guess the final key via the measurement result of her probe. The guessing probability intuitively describes the probability that Eve can correctly guess the final key, which can reflect the number of guesses that Eve requires to obtain the final key.

There are few studies on the guessing probability of QKD. Because there are more rigorous security criterions, such as the trace distance[5,6], which gives the composable security. This makes the theoretical foundation for security of QKD crucially important. However, in the real application of QKD projects, customers often ask the question of guessing probability. The existing prior art results cannot give them a satisfactory upper bound[11]. Consequently, some people questioned the security of QKD by relying on the prior art results of guessing probability[12]. For example, according to the existing result[11], the guessing probability of the ε-secure key is approximately 10^{-9} if ε is approximately 10^{-9}. From the perspective of guessing probability, the security of the value 10^{-9} is equivalent to that of a 30 perfect bits. The existing classical computer systems can easily crack such key. In practice, it is not unusual to request a much smaller guessing probability such as 10^{-100} or 10^{-1000}. Therefore, it is beneficial to find a more tightened upper bound of guessing probability.

As an important criterion in cryptography, guessing probability alone cannot guarantee the security of the final key. However, the large value of the loose upper bound of the guessing probability does not indicate the insecurity of the final key[12] because the value is not achievable by Eve, and one can find a more tightened value for the upper bound of the guessing probability. Here, by applying the trace distance criterion[2], we find such tightened bound. We show that the guessing probability is actually smaller than the existing bound values by many orders of magnitude if one takes the privacy amplification by Toeplitz matrix. This shows that the trace distance criterion[2] can actually produce a much better result than what was assumed previously in the viewpoint of guessing probability.

RESULTS

We consider the security definitions of a practical QKD protocol with finite size under the framework of composable security[3,4,13,14]. Suppose that Alice and Bob get two N-bit sifted key strings, $\mathbf{s}$ and $\mathbf{s}'$. By performing an error correction and private amplification scheme, Alice gets an n_1-bit key $\mathbf{k}$, and Bob gets an estimate key $\hat{\mathbf{k}}$ of $\mathbf{k}$ from $\mathbf{s}$ and $\mathbf{s}'$. The protocol is $\varepsilon_{\rm cor}$-correct if $P[\mathbf{k}\neq\hat{\mathbf{k}}]\leq\varepsilon_{\rm cor}$. In general, the key $\mathbf{k}$ of Alice can be correlated with an eavesdropper system, and the density matrix of Alice and Eve is $\rho_{\rm AE}$. The protocol outputs an ε-secure key[7], if

$$\frac{1}{2}\|\rho_{\rm AE}-\rho_{\rm U}\otimes\rho_{\rm E}\|_1\leq\varepsilon, \tag{1}$$

where $\|\cdot\|_1$ denotes the trace norm, $\rho_{\rm U}$ is the fully mixed state of Alice's system. The protocol is $\varepsilon_{\rm tol}$-secure if $\varepsilon_{\rm cor}$ and ε satisfy $\varepsilon_{\rm cor}+\varepsilon\leq\varepsilon_{\rm tol}$, which means that it is $\varepsilon_{\rm tol}$-indistinguishable from a perfect protocol (which is correct and secret). Without any loss of generality, we consider the case of $\varepsilon_{\rm cor}=\varepsilon$ in this article.

We define the security level:

Definition 1. If key $\mathbf{k}$ is ε-secure, the *security level* of key $\mathbf{k}$ is ε. For symbol clarity, we will use notation $\varepsilon_{\mathbf{k}}$ for the security level of key $\mathbf{k}$. With this definition, we can say that the key $\mathbf{k}$ is $\varepsilon_{\mathbf{k}}$-secure or that its security level is $\varepsilon_{\mathbf{k}}$. We define the guessing probability:

Definition 2. Let the final key generated by the QKD protocol be $\mathbf{k}$; the *guessing probability* of $\mathbf{k}$ is defined as the success probability

[1]State Key Laboratory of Low Dimensional Quantum Physics, Department of Physics, Tsinghua University, 100084 Beijing, China. [2]Jinan Institute of Quantum Technology, SAICT, 250101 Jinan, China. [3]Shenzhen Institute for Quantum Science and Engineering, and Physics Department, Southern University of Science and Technology, 518055 Shenzhen, China. [4]Data Communication Science and Technology Research Institute, 100191 Beijing, China. ✉email: xbwang@mail.tsinghua.edu.cn; jingtao0621@mail.tsinghua.edu.cn; yuzongwen@yeah.net

of the attacker Eve guessing the final key via her measurement result and is denoted as $p(\mathbf{k})$.

Lemma 1. *The guessing probability of $\varepsilon_{\mathbf{k}}$-secure key* **k** *with length n_1 is not larger than* $\frac{1}{2^{n_1}} + \varepsilon_{\mathbf{k}}$. This is a conclusion from ref. [11]. The proof has been already given in ref. [11]; for the convenience of readers, we write the proof again in the "Methods" section. According to Lemma 1, the guessing probability of key **k** can be divided into two parts; one part 2^{-n_1} is related to the length of the key, the other part $\varepsilon_{\mathbf{k}}(n_1)$ is related to the security level. Under the framework of universally composable security, when calculating the final key length, we often make the security level to be between 10^{-9} and 10^{-24}, which is much bigger than 2^{-n_1} because n_1 is often 10^3, 10^4, or larger. Therefore, 2^{-n_1} can be ignored and $p(\mathbf{k}) \leq \overline{p}(\mathbf{k}) \sim \mathcal{O}(\varepsilon(\mathbf{k}))$. However, the guessing probability of a secure key with a length of tens of bits can also reach this magnitude. Therefore, when the secure requirements are very high, it is clearly not enough for a key with a length of thousands of bits or even longer if the upper bound of guessing probability only stops at this magnitude. Therefore, we cannot simply use this formula alone to obtain the upper bound of the guessing probability. Fortunately, we have a much better way for tightening the bound. The approach will be presented below.

Lemma 2. *If key* **k** *can be mapped to string* $\mathbf{k}'$ *by a map M that is known to Eve, then the guessing probability of* **k** *cannot be larger than the guessing probability of string* $\mathbf{k}'$*, i.e.,*

$$p(\mathbf{k}) \leq p(\mathbf{k}'). \tag{2}$$

Here $p(\mathbf{k}), p(\mathbf{k}')$ *are the guessing probabilities of* **k** *and* $\mathbf{k}'$*, respectively*. *Proof*. This lemma is clear because when Eve can correctly guess **k**, Eve can obtain $\mathbf{k}'$ by knowing the map *M*. Otherwise, Eve can still correctly guess the $\mathbf{k}'$ with a probability not less than 0, i.e., $p(\mathbf{k}') = p(\mathbf{k}) + \delta, \delta \geq 0$.

Theorem 1. *If the $\varepsilon_{\mathbf{k}}$-secure key* **k** *with a length n_1 can be mapped to the $\varepsilon_{\mathbf{k}'}$-secure key* $\mathbf{k}'$ *with length n_2, the guessing probability of* **k** *cannot be larger than* $\mathbf{k}'$*, i.e.,*

$$p(\mathbf{k}) \leq \overline{p}(\mathbf{k}') = \frac{1}{2^{n_2}} + \varepsilon_{\mathbf{k}'}. \tag{3}$$

Proof. This theorem actually requires two conditions:

(i) the final key **k** can be mapped to the string $\mathbf{k}'$,
(ii) the string $\mathbf{k}'$ can be regarded as a $\varepsilon_{\mathbf{k}'}$-secure key.

Using the above-mentioned conditions, the proof is very simple. Given the condition (i), we can apply Lemma 2 to obtain

$$p(\mathbf{k}) \leq p(\mathbf{k}'). \tag{4}$$

Given the condition (ii), we can apply Lemma 1 to obtain

$$p(\mathbf{k}') \leq \overline{p}(\mathbf{k}') = \frac{1}{2^{n_2}} + \varepsilon_{\mathbf{k}'}, \tag{5}$$

where $\overline{p}(\mathbf{k}')$ is the upper bound of $p(\mathbf{k}')$. According to Eqs. (4) and (5), we can obtain

$$p(\mathbf{k}) \leq \overline{p}(\mathbf{k}') = \frac{1}{2^{n_2}} + \varepsilon_{\mathbf{k}'}. \tag{6}$$

This ends our proof of Theorem 1. As discussed above, if the length of the final key **k** and the string $\mathbf{k}'$ are very large, then 2^{-n_1} and 2^{-n_2} can be ignored. Meanwhile, if $n_2 < n_1$ and $\varepsilon_{\mathbf{k}'} < \varepsilon_{\mathbf{k}}$, then $2^{-n_2} + \varepsilon_{\mathbf{k}'} \sim \varepsilon_{\mathbf{k}'} \leq \varepsilon_{\mathbf{k}} \sim 2^{-n_1} + \varepsilon_{\mathbf{k}}$. Thus, Theorem 1 can provide a tighter upper bound of guessing probability.

Using Theorem 1, it is now possible for us to obtain the upper bound of the guessing probability of the $\varepsilon_{\mathbf{k}}$-secure key **k** more tightly. Instead of directly applying Lemma 1, we choose to first map *k* to an n_2-bit string $\mathbf{k}' = M(\mathbf{k})$. If the string $\mathbf{k}'$ itself can be regarded as an $\varepsilon_{\mathbf{k}'}$-secure final key, we can apply Theorem 1 by calculating $\overline{p}(\mathbf{k}')$. In addition, we can obtain a much smaller upper bound of the guessing probability of **k** if $\varepsilon_{\mathbf{k}'}$ is very small and n_2 is not too small. Now, the remaining problems are to determine the map *M*, to make sure that $\mathbf{k}' = M(\mathbf{k})$ is another key that is $\varepsilon_{\mathbf{k}'}$-secure, and to calculate $\varepsilon_{\mathbf{k}'}$. We start our method with the hashing function in the key distillation.

Our hashing function

We use the key distillation with the random matrix. Denote R_{nN} as the $n \times N$ random matrix with each element being randomly chosen to be either 0 or 1. In addition, we represent the *N*-bit sifted string **s** by a column vector, which contains *N* elements. To obtain the *n*-bit final key, we use the calculation $R_{nN}\mathbf{s}$. It can be easily confirmed that our random matrix belongs to the class of two-universal hashing function family[2].

Suppose we have distilled out the n_1-bit key **k** from the *N*-bit sifted key **s** through hashing by our random matrix R_{n_1N}. We can map the n_1-bit key **k** into the n_2-bit string $\mathbf{k}' = M(\mathbf{k})$ by deleting the last $n_1 - n_2$ bits from the key string **k**. Clearly, this string $\mathbf{k}'$ mapped from **k** can be also regarded as another final key distilled from the sift key **s** by the $n_2 \times N$ random hashing matrix R_{n_2N}, which is a submatrix of R_{n_1N}. In summary, we have

$$\mathbf{k}' = M(\mathbf{k}) = R_{n_2N}\mathbf{s}. \tag{7}$$

This means that $\mathbf{k}'$ is a string mapped from key **k**. Moreover, $\mathbf{k}'$ can be regarded as another final key of length n_2 distilled from the sifted key **s**. Because the two conditions in Theorem 1 are satisfied, according to Theorem 1, we can obtain a tightened upper bound of $p(\mathbf{k})$ with Eq. (3) if we know the security level of key $\mathbf{k}'$, i.e., the value of $\varepsilon_{\mathbf{k}'}$. Because our random matrix is a class of two-universal hashing function, the value $\varepsilon_{\mathbf{k}'}$ depends on n_2[4]. The details are shown in the "Methods" section and explain the calculation of $\varepsilon_{\mathbf{k}'}$ for n_2. Hence, in the QKD protocol that uses a random hashing matrix presented here, to obtain the upper bound of the guessing probability of the n_1-bit final key **k**, we can summarize the procedure above by the following scheme:

Scheme (1) Given the n_1-bit final key **k**, we delete its last $n_1 - n_2$ bits and obtain a string $\mathbf{k}'$. (2) We regard $\mathbf{k}'$ as another possible final key that is $\varepsilon_{\mathbf{k}'}$-secure. Compute the $\varepsilon_{\mathbf{k}'}$ value of $\mathbf{k}'$ with the input parameters *N* and n_2. (3) Calculate $\overline{p}(\mathbf{k})$ by Theorem 1 through Eq. (3).

Because on our scheme the value of $\varepsilon_{\mathbf{k}'}$ is dependent on n_2, as shown in the "Methods" section, we can now replace $\varepsilon_{\mathbf{k}'}$ by a functional form, $\varepsilon_{\mathbf{k}'}(n_2)$. To obtain the tightened upper bound value of the guessing probability in scheme 1, we need to choose an appropriate n_2 value. In our calculation, we set the condition

$$2^{-n_2} = \varepsilon_{\mathbf{k}'}(n_2), \tag{8}$$

for the appropriate n_2.

For any $n > n_2$, we have $\varepsilon_{\mathbf{k}}(n) > \varepsilon_{\mathbf{k}'}(n_2) = 2^{-n_2}$; however, for any $n < n_2$, we have $2^{-n} > 2^{-n_2}$. In conclusion, if $n \neq n_2$, $2^{-n} + \varepsilon_{\mathbf{k}}(n) > 2^{-n_2}$. Therefore, in this study, we set $2^{-n_2} = \varepsilon_{\mathbf{k}'}(n_2)$, and obtain a tightened guessing probability 2^{-n_2+1}.

Once we determine the value n_2 and the corresponding $\varepsilon_{\mathbf{k}'}(n_2)$, we calculate $\overline{p}(\mathbf{k}')$ by Eq. (3). Clearly, this is the upper bound of the guessing probability of the final key **k** of length n_1 provided that

$$n_1 > n_2. \tag{9}$$

Thus, we can actually use a more efficient scheme to obtain the upper bound of the guessing probability of key **k**, as the following Theorem 2:

As shown in Fig. 1, the arrow between **s** and **k** indicates that the $\varepsilon_{\mathbf{k}}$-secure n_1-bit final key **k** can be distilled from the *N*-bit sifted key **s** using a random matrix R_{n_1N}, i.e. $\mathbf{k} = R_{n_1N}\mathbf{s}$. The arrow between **k** and $\mathbf{k}'$ indicates that there exists a map *M* that can map the key **k**

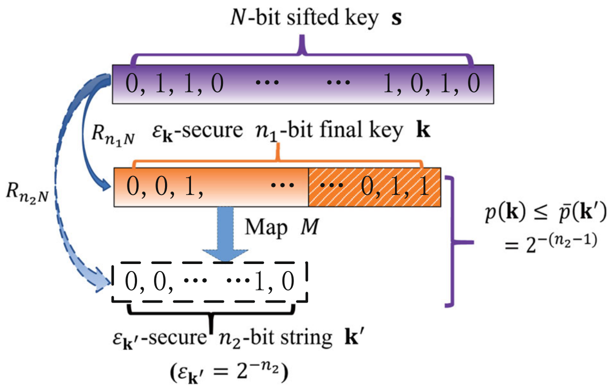

Fig. 1 Flow chart of our method of bounding the guessing probability. The arrow between **s** and **k** indicates that the $\varepsilon_{\mathbf{k}}$-secure n_1-bit final key **k** can be distilled from the N-bit sifted key **s** using a random matrix R_{n_1N}, i.e., $\mathbf{k} = R_{n_1N}\mathbf{s}$. The arrow between **k** and $\mathbf{k}'$ indicates that there exists a map M that can map the key **k** into $\mathbf{k}'$, i.e. $\mathbf{k}' = M(\mathbf{k})$. The arrow between the sifted key S and $\mathbf{k}'$ indicates that if a random hashing matrix R_{n_2N} is used to distill the final key, we have $\mathbf{k}' = R_{n_2N}\mathbf{s}$. Then, if n_2 satisfies the condition in Theorem 2, a tightened guessing probability of **k** can be obtained.

Table 1. Comparison of the guessing probability, where $Q_{\rm tol} = 2.14\%$ is the channel error tolerance, $N_z = 0.22N_{\rm tol}$ is the length of the string used to do parameter estimation, $N_{\rm tol}$ is the total length of the sifted key, $N = 0.78N_{\rm tol}$ is the length of the string for key generation, $\varepsilon = 10^{-9}$ is the security level, n is the length of the 10^{-9}-secure key, and p_g is the probability of correctly guessing the final key. Specifically, $p_g^{\rm Thm.2}$ is the result of Theorem 2 of this work.

$N_{\rm tol}$	10^4	10^5	10^6
n	2.01×10^3	4.06×10^4	4.90×10^5
p_g[12]	10^{-6}	10^{-6}	10^{-6}
p_g[11]	10^{-9}	10^{-9}	10^{-9}
$p_g^{\rm Thm.2}$	2×10^{-32}	2×10^{-327}	2×10^{-3277}

into $\mathbf{k}'$, i.e., $\mathbf{k}' = M(\mathbf{k})$. The arrow between the sifted key **s** and $\mathbf{k}'$ indicates that if a random hashing matrix R_{n_2N} is used to distill the final key, we have $\mathbf{k}' = R_{n_2N}\mathbf{s}$. Then if n_2 satisfies the condition in Theorem 2, a tightened guessing probability of **k** can be obtained.

There are two important points need to be noticed. First, when applying our theorem to obtain the nontrivial upper bound of the guessing probability for the final key **k**, we do not really need to transform **k** to another string $\mathbf{k}'$, and we only need the existence of a map that can map **k** to $\mathbf{k}'$ mathematically. That is to say, we use the final key **k**, but its guessing probability is calculated from the shorter key $\mathbf{k}'$. As shown above, the existence has been proven. Second, in this study, we use the random matrix R_{nN} as a family of two-universal hash functions to distill the key to illustrate our conclusion more intuitively. Of course, we can also use the modified Toeplitz matrix[8] instead of the random matrix R_{nN}. Thus, the final key **k** can be also mapped to the string $\mathbf{k}'$, and the string $\mathbf{k}'$ can also be regarded as the $\varepsilon_{\mathbf{k}'}$-secure key. This means that the proposed theorem in this study still holds.

Theorem 2. *In the QKD protocol, if the n_1-bit final key **k** is distilled from the sifted key **s** using a random matrix R_{n_1N}, the guessing probability of k can be upper bounded by*

$$p(\mathbf{k}) \leq \overline{p}(\mathbf{k}') = 2^{-(n_2-1)}, \qquad (10)$$

where $\mathbf{k}' = M(\mathbf{k}) = R_{n_2N}\mathbf{s}$ and n_2 satisfies $2^{-n_2} = \varepsilon_{\mathbf{k}'}(n_2), n_2 < n_1$.

Table 2. Comparison of the rate $r = n/N_{\rm tol}$ and $r' = n'/N_{\rm tol}$ under the same parameters shows in Table 1. ε and ε' are the security levels, n and n' are the length of ε-secure key and the length of ε'-secure key, respectively.

$N_{\rm tol}$	10^4	10^5	10^6
ε	10^{-9}	10^{-9}	10^{-9}
n	2.01×10^3	4.06×10^4	4.90×10^5
r	0.20	0.41	0.49
ε'	10^{-32}	10^{-327}	10^{-3277}
n'	136	1.12×10^3	1.10×10^4
r'	0.01	0.01	0.01

DISCUSSION

Table 1 describes the upper bounds of the guessing probability calculated by different $N_{\rm tol}$, where $N_{\rm tol}$ is the length of the total string that includes the sifted keys for key generation and the string used to do parameter estimation. In Table 1, $N_{\rm tol} = 10^4$, 10^5, and 10^6. Table 1 shows that when $N_{\rm tol} = 10^6$, $n = 4.90\times10^5$ and the guessing probabilities obtained using the methods of ref. [12] and ref. [11] are approximately 10^{-6} and 10^{-9}, respectively. However, using our method, the guessing probability can be reduced to 2×10^{-3277}, which is more tightened by thousands of orders of magnitude than prior art methods. With an increase in the length of $N_{\rm tol}$, the length of the final key also increases; however, the guessing probabilities in ref. [12] and ref. [11] almost remain unchanged. Compared with ref. [12] and ref. [11], the guessing probability obtained by our method is considerably reduced, which is more realistic and tighter. It should be noted that we calculate the case without the known-plaintext attack (KPA) in Table 1. Now, we consider the case of KPA in QKD using our method. Suppose that Eve knows the t bits of the final n_2-bit key $\mathbf{k}'$; then, the guessing probability of the $\varepsilon_{\mathbf{k}'}$-secure key $\mathbf{k}'$ is $p_{\rm KPA}(\mathbf{k}') \leq 2^{-(n_2-t-1)}$. Now, the upper bound of the guessing probability of key $\mathbf{k}'$ is equal to that of an ideal $(n_2 - t - 1)$-bit key.

Table 2 compares the length of the ε-secure key n and the length of ε'-secure key n' when the total length of the sifted key is 10^4, 10^5, and 10^6. This table shows that if only using Lemma 1 to obtain a smaller guessing probability, ε needs to be reduced. Accordingly, the length of the final key and the key rate will be considerably reduced. For example, from Table 2, when $N_{\rm tol} = 10^6$, if the customer wants to reduce the guessing probability from 10^{-9} to 2×10^{-3277}, the length of the key will become $n' = 1.1\times10^4$, and the key rate will become $r' = 0.01$. This result is much lower than the original key length $n = 4.9\times10^5$ and the key rate $r = 0.49$. Using our result, there is actually no bit cost for a much smaller bound value of guessing probability. For example, when $N_{\rm tol} = 10^6$, we can upper bound the guessing probability by 2×10^{-3277} by setting $\varepsilon = 10^{-9}$. Thus, without reducing the value of ε, we can obtain a tightened upper bound of guessing probability $p_g^{\rm Thm.2}$ of **k**, as can be seen from Table 1.

Our result shows that in terms of guessing probability, the performance of the existing trace distance security is much better than what has been assumed in the past. Incidentally, in ref. [11], a looser upper bound, 10^{-6} for Eve's guessing probability, was presented[12]. We emphasize that this looser upper bound does not in any sense challenge the validity of the existing security proof of QKD[11]. Although the large value of *lower bound* of Eve's guessing probability can show insecurity, the large value of *upper bound* cannot show insecurity. If one does not make any effort, one can also obtain a large-value upper bound of 100% for Eve's guessing probability. Such value is correct for the upper bound but not meaningful. If any new upper bound is larger than that in the prior art result, it means that the "new upper bound" is trivial and meaningless rather than the prior art result is invalid. Thus, the looser upper bound presented by ref.[12] only shows that Eve's guessing probability of the key is smaller than 10^{-6}. It does not conflict with more tightened results presented elsewhere.

In this study, our goal is to obtain a tightened guessing probability. On the basis of the existing secure criterion (Trace distance) and the general property of guessing probability, we propose a simple and efficient method to tighten the upper bound of the guessing probability. We find that the guessing probability $p(\mathbf{k})$ of $\mathbf{k}$ can be upper bounded by $2^{-(n_2-1)}$, where n_2 satisfies $2^{-n_2}=\varepsilon_{\mathbf{k}'}(n_2)$ and $n_2<n_1$. Specifically, a simple random matrix R_{nN} can be used to distill the final key. Compared with the prior art results, of which the upper bound of the guessing probability of the ε-secure key is approximately ε, our method provides a more tightened upper bound. Therefore, the loose upper bound for the guessing probability obtained in ref.[12] cannot be regarded as evidence to question the validity of existing the security proof of QKD.

METHODS

Proof of Lemma 1

Lemma 1. *The guessing probability of the $\varepsilon_{\mathbf{k}}$-secure key $\mathbf{k}$ with length n_1 is not larger than* $\frac{1}{2^{n_1}}+\varepsilon_{\mathbf{k}}$.

This is a conclusion obtained from ref.[11]. The proof has been already presented in ref.[11]. Here, for the convenience of the reader, we write the proof again.

Proof. Let the n-bit string $\mathbf{x}$ be the $\varepsilon_{\mathbf{x}}$-secure key in $\mathcal{X}$. The density matrix of Alice and Eve is ρ_{XE} and satisfies

$$\rho_{\mathrm{XE}}=\sum_{\mathbf{x}\in\mathcal{X}}|\mathbf{x}\rangle\langle\mathbf{x}|\otimes\rho_E^{\mathbf{x}}, \tag{11}$$

$$\frac{1}{2}\parallel\rho_{\mathrm{XE}}-\rho_{\mathrm{U_x}}\otimes\rho_{\mathrm{E}}\|_1\le\varepsilon_{\mathbf{x}},$$

where $\rho_{\mathrm{U_x}}$ is the fully mixed state in $\mathcal{X}$. Then we have

$$\begin{aligned}&\frac{1}{2}\parallel\rho_{\mathrm{XE}}-\rho_{\mathrm{U_x}}\otimes\rho_{\mathrm{E}}\|_1\\&\ge\frac{1}{2}\left\|\sum_{\mathbf{x}\in\mathcal{X}}q(\mathbf{x})|\mathbf{x}\rangle\langle\mathbf{x}|-\sum_{\mathbf{x}\in\mathcal{X}}\frac{1}{2^n}|\mathbf{x}\rangle\langle\mathbf{x}|\right\|_1\\&=\frac{1}{2}\sum_{\mathbf{x}\in\mathcal{X}}\left|q(\mathbf{x})-\frac{1}{2^n}\right|.\end{aligned} \tag{12}$$

Eve's guessing probability of string $\mathbf{x}$ is $q(\mathbf{x})$, and the maximum guessing probability is $p_{\mathrm{g}}=\max_{\mathbf{x}\in\mathcal{X}}\{q(\mathbf{x})\}$. Without any loss of generality, it is possible to assume that the maximum guessing probability is $q(\mathbf{x}')$. Note that $\sum_{\mathbf{x}\in\mathcal{X}}q(\mathbf{x})=1$, then the following holds

$$\begin{aligned}&\frac{1}{2}\sum\nolimits_{\mathbf{x}\in\mathcal{X}}\left|q(\mathbf{x})-\frac{1}{2^n}\right|\\&=\frac{1}{2}\left|q(\mathbf{x}')-\frac{1}{2^n}\right|+\frac{1}{2}\sum_{\mathbf{x}\in\mathcal{X},\mathbf{x}\neq\mathbf{x}'}\left|q(\mathbf{x})-\frac{1}{2^n}\right|\\&\ge\frac{1}{2}\left|q(\mathbf{x}')-\frac{1}{2^n}\right|+\frac{1}{2}\left|\sum_{\mathbf{x}\in\mathcal{X},\mathbf{x}\neq\mathbf{x}'}[q(\mathbf{x})-\frac{1}{2^n}]\right|\\&=\left|q(\mathbf{x}')-\frac{1}{2^n}\right|.\end{aligned} \tag{13}$$

From Eqs. (11) to (13), we have $p_{\mathrm{g}}\le 2^{-n_1}+\varepsilon_{\mathbf{x}}$; thus, for the n_1-bit $\varepsilon_{\mathbf{k}}$-secure key $\mathbf{k}$, the guessing probability satisfies

$$p(\mathbf{k})\le\overline{p}(\mathbf{k})=\frac{1}{2^{n_1}}+\varepsilon_{\mathbf{k}}, \tag{14}$$

where $\overline{p}(\mathbf{k})$ is the upper bound of $p(\mathbf{k})$. This ends our proof of Lemma 1.

Calculation of $\varepsilon'_{\mathbf{k}}$

We consider the security definitions of a practical QKD protocol with a finite size under the framework of composable security[4,13,14]. Suppose that Alice and Bob get two N-bit sifted key strings. By performing an error correction and private amplification scheme, Alice get an n-bit final key $\mathbf{k}$ and Bob get an estimate $\hat{\mathbf{k}}$ of $\mathbf{k}$. The protocol is $\varepsilon_{\mathrm{cor}}$-correct if $P[\mathbf{k}\neq\hat{\mathbf{k}}]\le\varepsilon_{\mathrm{cor}}$. In general, the key $\mathbf{k}$ of Alice can be correlated with an eavesdropper system, and the density matrix of Alice and Eve is ρ_{AE}.

The protocol outputs an $\varepsilon_{\mathbf{k}}$-secure key[13], if

$$\frac{1}{2}\parallel\rho_{\mathrm{AE}}-\rho_{\mathrm{U}}\otimes\rho_{\mathrm{E}}\|_1\le\varepsilon_{\mathbf{k}}, \tag{15}$$

where $\|\cdot\|_1$ denotes the trace norm, ρ_{U} is the fully mixed state of Alice's system. The protocol is $\varepsilon_{\mathrm{tol}}$-secure if $\varepsilon_{\mathrm{cor}}$ and $\varepsilon_{\mathbf{k}}$ satisfies $\varepsilon_{\mathrm{cor}}+\varepsilon_{\mathbf{k}}\le\varepsilon_{\mathrm{tol}}$, which means that it is $\varepsilon_{\mathrm{tol}}$-indistinguishable from an ideal protocol. Without any loss of generality, we consider the case of $\varepsilon_{\mathrm{cor}}=\varepsilon_{\mathbf{k}}$.

From Lemma 1, we can calculate $\overline{p}(\mathbf{k})$ given the n-bit $\varepsilon_{\mathbf{k}}$-secure key $\mathbf{k}$. In this situation, $\overline{p}(\mathbf{k})=2^{-n}+\varepsilon_{\mathbf{k}}$. However, in our method, we only know N and n_2, which are the length of the sifted key and $\mathbf{k}'$. (The string $\mathbf{k}'$ itself can be also regarded as another final key distilled from the sifted key.) To get a tightened upper bound of the guessing probability of $\mathbf{k}$, we need to obtain the value of $\varepsilon_{\mathbf{k}'}$. According to ref.[4], with N and n_2, the final key is $\varepsilon_{\mathbf{k}'}$-secure if $\varepsilon_{\mathbf{k}'}$ satisfies the following equation:

$$n_2\le N[1-h(Q_{\mathrm{tol}}+\mu)]-fNh(Q_{\mathrm{tol}})-\log\frac{2}{\varepsilon_{\mathbf{k}'}^3}, \tag{16}$$

where $\mu=\sqrt{\frac{N+N_z}{NN_z}\frac{N_z+1}{N_z}\ln\frac{2}{\varepsilon_{\mathbf{k}'}}}$, N_z is the length of string used for parameter estimation, $f=1.1$, h denotes the binary Shannon entropy function, $h(x)=-x\log x-(1-x)\log(1-x)$ and Q_{tol} represents the channel error tolerance. To obtain nontrivial results, we use equality in Eq. (16) to calculate the value of $\varepsilon_{\mathbf{k}'}$, given the input n_2. Since $\varepsilon_{\mathbf{k}'}$ is dependent on n_2, we use notation $\varepsilon_{\mathbf{k}'}(n_2)$ for $\varepsilon_{\mathbf{k}'}$. Here, $\varepsilon_{\mathbf{k}'}(n_2)$, if n_2 is given and we numerically find the value of $\varepsilon_{\mathbf{k}'}$ by Eq. (16).

In our calculation, we choose a specific n_2-value that satisfies

$$2^{-n_2}=\varepsilon_{\mathbf{k}'}(n_2). \tag{17}$$

In combination with Eq. (16), we obtain the following equation for the tightened $\varepsilon_{\mathbf{k}'}$ value:

$$-\log\varepsilon_{\mathbf{k}'}=N[1-h(Q_{\mathrm{tol}}+\mu)]-fNh(Q_{\mathrm{tol}})-\log\frac{2}{\varepsilon_{\mathbf{k}'}^3}, \tag{18}$$

and we can calculate the value of $\varepsilon_{\mathbf{k}'}$ and then calculate the guessing probability by Eq. (8) in our main body text.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding authors upon reasonable request.

Received: 26 September 2019; Accepted: 25 March 2020;
Published online: 22 May 2020

REFERENCES

1. Bennett, C. & Brassard, G. Quantum cryptography: public key distribution and coin tossing. In *Proc. IEEE International Conference on Computers, Systems, and Signal Processing*, Bangalore, India, 175–179 (IEEE Press, New York, 1984).
2. Renner, R. Security of quantum key distribution. *Int. J. Quantum Inf.* **6**, 1 (2008).
3. Curty, M. et al. Finite-key analysis for measurement-device-independent quantum key distribution. *Nat. Commun.* **5**, 3732 (2014).
4. Tomamichel, M., Lim, C. C. W., Gisin, N. & Renner, R. Tight finite-key analysis for quantum cryptography. *Nat. Commun.* **3**, 634 (2012).
5. Ben-Or, M., Horodecki, M., Leung, D. W., Mayers, D. & Oppenheim, J. In *Theory of Cryptography Conference*, 386–406 (Springer, 2005).
6. Renner, R. & König, R. In *Theory of Cryptography Conference*, 407–425 (Springer, 2005).
7. König, R., Renner, R., Bariska, A. & Maurer, U. Small accessible quantum information does not imply security. *Phys. Rev. Lett.* **98**, 140502 (2007).
8. Hayashi, M. & Tsurumaru, T. Concise and tight security analysis of the Bennett-Brassard 1984 protocol with finite key lengths. *N. J. Phys.* **14**, 093014 (2012).
9. Alimomeni, M. & Safavi-Naini, R. In *International Conference on Information Theoretic Security*, 1–13 (Springer, 2012).

10. Issa, I. & Wagner, A. B. Measuring secrecy by the probability of a successful guess. *IEEE Trans. Inf. Theory* **63**, 3783 (2017).
11. Portmann, C. & Renner, R. Cryptographic security of quantum key distribution. Preprint at: https://arxiv.org/abs/1409.3525 (2014).
12. Yuen, H. P. Security of quantum key distribution. *IEEE Access* **4**, 724 (2016).
13. Canetti, R. In *Proceedings 2001 IEEE International Conference on Cluster Computing*, 136−145 (IEEE, 2001).
14. Müller-Quade, J. & Renner, R. Composability in quantum cryptography. *N. J. Phys* **11**, 085006 (2009).

ACKNOWLEDGEMENTS

We acknowledge the financial support in part by the Ministry of Science and Technology of China through The National Key Research and Development Program of China grant no. 2017YFA0303901; National Natural Science Foundation of China grant nos. 11474182, 11774198, 11974204 and U1738142.

AUTHOR CONTRIBUTIONS

X.-B.W. developed the theory, J.-T.W. and J.-Q.Q. contributed equally to the calculation work, C.J. and Z.-W.Y. contributed to simulation work. All authors contributed to the manuscript.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Correspondence and requests for materials should be addressed to X.-B.W., J.-T.W. or Z.-W.Y.