

# An Improved Slice Reconciliation Protocol for Continuous-Variable Quantum Key Distribution

Xuan Wen <sup>1</sup>, Qiong Li <sup>1,\*</sup> , Haokun Mao <sup>1</sup>, Xiaojun Wen <sup>2,\*</sup> and Nan Chen <sup>3</sup>

<sup>1</sup> Department of Computer Science and Technology, Harbin Institute of Technology, Harbin 150000, China; wenxuan@hit.edu.cn (X.W.); 14B303003@hit.edu.cn (H.M.)

<sup>2</sup> School of Electronics and Information Engineering, Shenzhen Polytechnic, Shenzhen 518000, China

<sup>3</sup> School of Foreign Languages, Harbin Institute of Technology, Harbin 150000, China; chennan2010@hit.edu.cn

\* Correspondence: qiongli@hit.edu.cn (Q.L.); wxjun@szpt.edu.cn (X.W.)

**Abstract:** Reconciliation is an essential procedure for continuous-variable quantum key distribution (CV-QKD). As the most commonly used reconciliation protocol in short-distance CV-QKD, the slice error correction (SEC) allows a system to distill more than 1 bit from each pulse. However, the quantization efficiency is greatly affected by the noisy channel with a low signal-to-noise ratio (SNR), which usually limits the secure distance to about 30 km. In this paper, an improved SEC protocol, named Rotated-SEC (RSEC), is proposed through performing a random orthogonal rotation on the raw data before quantization, and deducing a new estimator for the quantized sequences. Moreover, the RSEC protocol is implemented with polar codes. The experimental results show that the proposed protocol can reach up to a quantization efficiency of about 99%, and maintain at around 96% even at the relatively low SNRs (0.5, 1), which theoretically extends the secure distance to about 45 km. When implemented with the polar codes with a block length of 16 Mb, the RSEC achieved a reconciliation efficiency of above 95%, which outperforms all previous SEC schemes. In terms of finite-size effects, we achieved a secret key rate of  $7.83 \times 10^{-3}$  bits/pulse at a distance of 33.93 km (the corresponding SNR value is 1). These results indicate that the proposed protocol significantly improves the performance of SEC and is a competitive reconciliation scheme for the CV-QKD system.

**Keywords:** continuous-variable quantum key distribution; reconciliation; slice error correction; polar codes; finite-size effect



**Citation:** Wen, X.; Li, Q.; Mao, H.; Wen, X.; Chen, N. An Improved Slice Reconciliation Protocol for Continuous-Variable Quantum Key Distribution. *Entropy* **2021**, *23*, 1317. <https://doi.org/10.3390/e23101317>

Academic Editor: Rosario Lo Franco

Received: 25 August 2021

Accepted: 7 October 2021

Published: 9 October 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

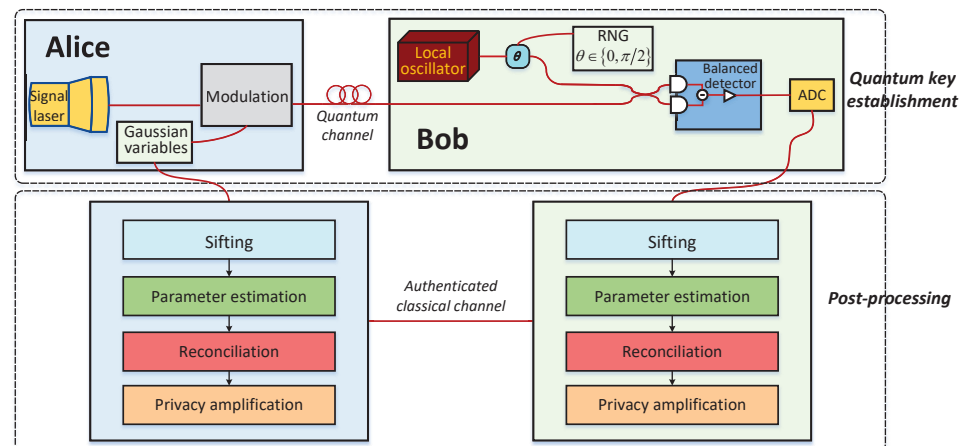
## 1. Introduction

Quantum key distribution (QKD), which enables two remote legitimate parties (i.e., Alice and Bob) to share unconditional secret keys against a potential eavesdropper, is a major practical quantum cryptography technology in quantum information [1]. There are mainly two categories of QKD protocols, namely discrete-variable (DV) protocols [2–6] and continuous-variable (CV) protocols [7–11], which, respectively, encode information on discrete variables (such as the polarization or the phase of single photons) and continuous variables (such as the quadratures of coherent states).

The DV-QKD needs a high-cost single-photon detector requiring cryogenic temperatures to measure the received quantum state, which presents a challenge for its widespread implementation. Compared to the DV-QKD, the CV-QKD takes the advantage of using a standard and cost-effective detector that is routinely deployed in standard telecom components working at room temperature. The security proof of CV-QKD against general attacks has been provided [12–15]. Many experiments of CV-QKD have been successfully implemented, especially the integrated silicon photonic chip for CV-QKD that offers new possibilities for low-cost and portable quantum communication [16].

A CV-QKD system mainly includes two consecutive phases [7–9]: the quantum key establishment phase and the classical post-processing phase, which are illustrated in Figure 1. In the first phase, Alice prepares a coherent state using two Gaussian variables

and sends it to Bob through the quantum channel. Then, Bob randomly chooses one of the two variables to measure his received coherent state and informs Alice of his choice. Owing to the physical noises or the existence of Eve [17] in a quantum channel, the raw data of the legitimate parties obtained from the first phase are weakly correlated and weakly secure continuous variables.



**Figure 1.** Schematic diagram of the CV-QKD system.

To extract identical secret keys from their raw data, Alice and Bob subsequently perform a phase called post-processing, including four main stages: sifting, parameter estimation [18–20], reconciliation [21–25], and privacy amplification [26–28]. Reconciliation is a crucial stage for CV-QKD, which allows the legitimate parties to distill the corrected keys from their raw data via an authentic classical channel. Its performance affects the secret key rate and the secure distance of the practical CV-QKD system [29–32].

Up to now, various reconciliation schemes have been proposed for reconciling the raw data of CV-QKD. Originally, C. Silberhorn et al. proposed sign reconciliation that first quantifies the raw data to bit string by using the sign and then corrects the error bits [21]; however, its low reconciliation efficiency limits its application. Subsequently, V. Assche et al. proposed SEC, which chooses a set of quantization functions to convert a continuous variable into binary-value slices and then executes error correction on the quantized slices [22,23].

Soon after, many researchers apply code-modulated techniques, including multilevel coding (MLC) and multistage decoding (MSD) in SEC with Low Density Parity Check (LDPC) codes to improve the reconciliation performance at high SNRs [33,34]. The SEC scheme allows one to extract more than one bit of key from per pulse, especially at the high SNRs; however, its quantization performance is poor at the low SNR of long-distance CV-QKD, which limits its secure distance to about 30 km. Afterward multidimensional reconciliation was proposed by Anthony Leverrier et al. [24], which extends the secure distance from 30 km to above 50 km. Since the code rate of multidimensional reconciliation is limited to 1 bit per pulse, its related research is mainly focused on improving the reconciliation efficiency with LDPC codes, and especially with Multi-edge type LDPC (MET-LDPC) codes at low SNRs [35–41].

In summary, the existing research on reconciliation is mainly based on SEC and multidimensional reconciliation. These two schemes have their own advantages and disadvantages. Multidimensional reconciliation has a better quantization scenario than SEC reconciliation, and thus it can still achieve a high-efficiency reconciliation for a long-distance CV-QKD system with a noisy channel. However, its code rate is limited to 1 bit per pulse, which makes it more suitable for a long-distance CV-QKD system. Compared with multidimensional reconciliation, the SEC has advantages in extracting more than 1 bit of secret key per channel use.

Limited by its quantization performance, the SEC protocol is more suitable for the short-distance CV-QKD system. As is known, the secret key rate of a QKD system will decrease rapidly with the increase of distance [29]. Due to the technology immaturity of the physical device, the key generation rate of the long-distance CV-QKD system is generally low [32,42], which obviously cannot satisfy the communication demand.

Therefore, to establish QKD networks [43–46] with the short-distance QKD system is a practical scheme to provide relatively high-speed keys for secure communication at present [47]. In addition, the LDPC code is usually chosen to pursue a high reconciliation efficiency, but its matrix design is extremely difficult. By contrast, another common family of codes, polar codes, is relatively easier to construct and their recursive structure delivers excellent performance in practice.

In this research, our work focus on the improvement of the SEC protocol and the reconciliation of the data with polar codes. The main contributions of this paper are as follows: (i) We improve the SEC protocol by first performing a random orthogonal rotation on the raw data before slice quantization and then provide a novel estimator for the quantized slices. Compared with the SEC protocol, the improved protocol, named RSEC, has a higher quantization efficiency, which then increases the secret key rate and reconciliation efficiency. (ii) In order to accomplish the reconciliation of the correlated continuous variable in CV-QKD, we implement the RSEC protocol by combining the polar codes, achieving a high-efficiency reconciliation.

The rest of this paper is organized as follows: In Section 2, the RSEC protocol is proposed to improve the SEC protocol. In Section 3, the implementation of the RSEC protocol with polar codes is described. In Section 4, the experimental results and analysis of RSEC are given. Finally, our conclusions are drawn in Section 5.

## 2. Rotated Slice Error Correction (RSEC) Protocol

In this section, we briefly review the SEC reconciliation and then put forward RSEC to improve the current SEC. After the quantum key establishment phase of the Gaussian-modulated coherent state CV-QKD protocol, Alice and Bob share weakly correlated continuous-variable raw data due to the noises during the quantum transmission. The noises can safely be assumed to be Gaussian since they correspond to the case of the optimal attack for Eve [12].

The correlated raw data are obtained by randomly measuring either the the amplitude and phase quadratures for each coherent state. Moreover, the information encoded on the two quadratures follow the same Gaussian distribution. For the convenience of description, let  $X = (x_1, x_2, \dots)$  and  $Y = (y_1, y_2, \dots)$  correspond to the correlated Gaussian random variables of Alice and Bob, respectively. Then, the correlated raw data can be modeled as  $Y = X + Z$  with  $x_i \sim N(0, \delta^2)$ ,  $z_i \sim N(0, \sigma^2)$ , where  $Z = (z_1, z_2, \dots)$ ,  $\delta^2$  and  $\sigma^2$  denote Alice's modulation variance and the noise variance, respectively.

In the direct reconciliation scenario, Alice's sequence is used as the target to correct Bob's sequence. On the contrary, the reverse reconciliation scenario uses Bob's sequence as the target to correct Alice's sequence. Generally, the latter scenario can obtain a higher secret key rate [35,38]. Without loss of generality, we only consider the reverse reconciliation in this research.

### 2.1. Review of Slice Error Correction

In information reconciliation, Alice and Bob first perform an operation called quantization to convert the correlated values into binary sequences and then choose an error correction scheme to correct the binary sequence over an authenticated classical channel. SEC is a generic reconciliation protocol [22]. Its underlying idea is to convert Alice's and Bob's values into bit strings with the slice function (i.e., quantization function) and then apply an error correction scheme as a primitive, taking advantage of all available information to minimize the number of exchanged reconciliation messages.

This works in two steps: First, Bob chooses a quantization function  $\mathcal{S}(x) : \mathbb{R} \rightarrow \{0, 1\}^m$  to map his raw data to  $m$ -slices binary digits, and informs Alice of the first  $t$  slices (usually  $t = 2$  or  $3$ ),  $\mathcal{S}(x)$  is a vector of slices  $\mathcal{S}(x) = (\mathcal{S}_1(x), \dots, \mathcal{S}_m(x))$ ; then, Bob sequentially deals with the remaining slice  $k$  ( $t + 1 \leq k \leq m$ ) by sending a syndrome of  $\mathcal{S}_k(x)$  to Alice so that Alice can recover  $\mathcal{S}_k(x)$  with a high probability.

The quantization function is to divide the set of real numbers  $\mathbb{R}$  into  $2^m$  intervals and then to assign different binary values to each of these intervals. There are two different schemes to construct the quantization function. The first construction scheme is to divide  $\mathbb{R}$  with  $2^m - 1$  equidistant points. The second construction scheme freely chooses  $2^m - 1$  points to divide  $\mathbb{R}$ , which performs better but has a much higher computational complexity. The previous work indicated that the second scheme does not improve as much as the quantization efficiency compared with the first scheme [33]. Therefore, we use the first scheme to construct the quantization function in this research.

In addition, previous studies have shown that the best bit assignment method is to assign the least significant bit of the binary representation of  $a - 1$  ( $0 \leq a - 1 \leq 2^m - 1$ ) to the first slice  $\mathcal{S}_1(x)$  when  $\tau_{a-1} \leq x < \tau_a$  [22]. The variables  $\tau_j$  divide the real numbers  $\mathbb{R}$  into  $2^m$  intervals, where  $1 \leq j \leq 2^m - 1$ ,  $\tau_0 = -\infty$ ,  $\tau_{2^m} = +\infty$ . Then, each bit of  $a - 1$  is subsequently assigned to the remaining slices. More specifically,

$$\mathcal{S}_i(x) = \begin{cases} 0 & , \text{if } \tau_{2^{i-1}n} \leq x < \tau_{2^i n + 2^{i-1}} \\ 1 & , \text{otherwise} \end{cases} \quad (1)$$

where  $1 \leq i \leq m$  and  $n$  is a nonnegative integer.

## 2.2. Improvement of Slice Error Correction

In the decoding process of SEC, the slice sequences are corrected in sequence; hence, the estimation of the current slice recursively depends on all previous slices. For this reason, the performance of SEC can be improved by reducing the bit error rate (BER)  $e_i$  of the previously decoded slices,  $e_i$  denotes the probability that Alice makes a wrong estimate of Bob's slice value  $\mathcal{S}_i(x)$ . According to the characteristics of quantization function  $\mathcal{S}(x)$ , it is not hard to find that the last slice  $\mathcal{S}_m(x)$  corresponds exactly to the sign of input variable  $x$ . Therefore, the quantization scheme of the last slice is similar to the multidimensional reconciliation, which uses the sign of the rotated data as the target sequence.

As is known, multidimensional reconciliation typically performs better than the SEC reconciliation in estimating the quantized values, especially at a low SNR [24]. For each slice, although having obtained the first few slices, Alice still needs to infer Bob's slice value in a certain number of intervals. Taking the case of  $m = 4$  slices as an example, if Alice has the first two slices  $(\mathcal{S}_1(x), \mathcal{S}_2(x)) = (0, 1)$ , she needs to estimate Bob's slice  $\mathcal{S}_3(x)$  among four intervals, i.e.,  $(\tau_2, \tau_3)$ ,  $(\tau_6, \tau_7)$ ,  $(\tau_{10}, \tau_{11})$ ,  $(\tau_{14}, \tau_{15})$  to satisfy  $(\mathcal{S}_1(x), \mathcal{S}_2(x)) = (0, 1)$ . However, multidimensional reconciliation calculates the probabilities of Bob's quantized value with the joint density function directly, which leads to more accurate estimations.

Consequently, to reduce the BER of the slice, we could execute a random orthogonal rotation on the raw data before the slice quantization and then infer the last slice  $\mathcal{S}_m(x)$  according to multidimensional reconciliation. After decoding the  $m$ -th slice, Alice corrects the remaining slices in order. Assuming that Alice and Bob agree on the quantization function  $\mathcal{S}(x)$  and the dimension  $d$  of the orthogonal matrix, the procedure of our improved protocol for reverse reconciliation is shown in Figure 2. The detailed process is described as follows:

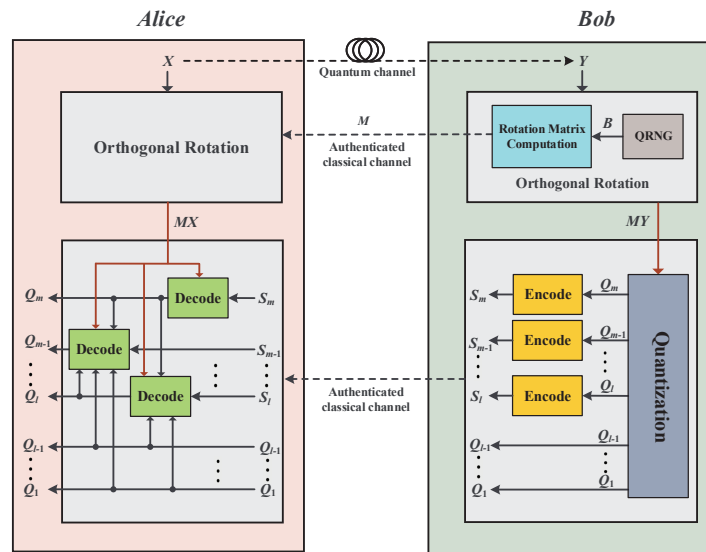


Figure 2. Procedure of the RSEC protocol for the continuous variables of the CV-QKD system.

**Step 1:** Alice and Bob divide their raw data into  $d$ -dimensional vectors as  $X = \{x_i\}^d$ ,  $Y = \{y_i\}^d$ . Bob randomly generates a bit string  $B = \{b_i\}^d$  and chooses a point  $U = \{\mu_i\}^d$  on the unit sphere  $\mathcal{O}^{d-1}$  adjacent to the point  $U_B = \{\frac{(-1)^{b_i}}{\sqrt{d}}\}^d$ . Then, he calculates an orthogonal matrix  $M$  satisfying  $MT = U$  for rotating  $Y$  to  $Y' = MY$ , and informs Alice of the matrix  $M$ , where  $T = \{t_i\}^d$ ,  $t_i = \frac{y_i}{\|Y\|}$ .

**Step 2:** After receiving Bob's orthogonal matrix  $M$ , Alice performs the same rotation on  $X$  and has the rotated data  $X' = MX$ .

**Step 3:** Bob quantizes his rotated data  $Y'$  into  $m$ -slice bit vectors with the quantization function  $\mathcal{S}(x)$ , such as Equation (1), and sends the quantized slice values of the  $1 \sim (l-1)$  slices  $Q_1, \dots, Q_{l-1}$  to Alice, where  $Q_i = \mathcal{S}_i(Y')$ .

**Step 4:** Alice constructs a bit string  $\hat{Q}_m$  of the  $m$ -th slice  $Q_m$  from her rotated data  $X'$  using the slice estimator  $\hat{\mathcal{S}}$  as Equation (11) in Section 2.3. Subsequently, Bob uses a chosen error correction codes to generate a syndrome  $S_m$  so that Alice aligns her bit string  $\hat{Q}_m$  on the sequence  $Q_m$ .

**Step 5:** For each subsequent slice  $k$ ,  $l \leq k < m$ , Alice constructs a new string by applying the slice estimator  $\hat{\mathcal{S}}$  to  $X'$ , and taking into account the disclosed slices  $Q_1, \dots, Q_{l-1}$  and the previously corrected bit strings  $Q_l, \dots, Q_{k-1}$ ,  $Q_m$ . Again, Alice aligns her bit string to Bob's sequence  $Q_k$  using their chosen error correction codes and corresponding syndrome  $S_k$ .

### 2.3. Slice Estimator of RSEC

In the decoding stage of the RSEC reconciliation, we need to use the side information to estimate Bob's quantized slices first. Let us now detail the expressions we proposed. According to the decoding process, we first estimate the last slice  $m$  of Bob. As is known,

$$\begin{aligned} Y' - X' &= MY - MX \\ &= MZ. \end{aligned} \tag{2}$$

where  $M = (m_{ij})_{d \times d}$  is the rotation matrix, and  $Z = \{z_i\}^d$  follows the Gaussian distribution,  $z_i \sim N(0, \sigma^2)$ .

As Gaussian variables have linear translation invariance—i.e., the linear combination of the independent Gaussian variables is still a Gaussian random variable—then  $z'_i = y'_i - x'_i$  follows the following Gaussian distribution

$$z'_i \sim N(0, \sum_{j=1}^d m_{ij}^2 \sigma^2). \tag{3}$$

It is known that  $\sum_{j=1}^d m_{ij}^2 = 1, (i = 1, 2, \dots, d)$  since  $M$  is an orthogonal matrix. Therefore, the random variable  $Z' = Y' - X'$  has the same probability distribution as  $Z$ , i.e.,

$$Y' - X' \sim N(0, \sigma^2)^d, \tag{4}$$

where  $Z' = \{z'_i\}^d$ . Similarly, Bob’s rotated data  $Y' = MY$  follows the distribution  $Y' \sim N(0, \delta^2 + \sigma^2)^d$ , and  $X' = MX$  follows the distribution  $X' \sim N(0, \delta^2)^d$ . In addition, according to the characteristics of the quantization function  $\mathcal{S}(x)$ , the bit string  $Q_m = (Q_m^1, Q_m^2, \dots)$  corresponds to the sign of the rotated data  $Y'$ , i.e., if  $y'_i \geq 0$ ,  $Q_m^i = 1$ , else,  $Q_m^i = 0, i = 1, 2, \dots$ . Here, we use  $Q_k^j$  to denote the  $k$ -th slice of  $y'_j$ . Hence, we obtain the conditional probability of  $Q_m^i$  as follows

$$P_m(Q_m^i | x'_i) = \frac{K}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\mathcal{J}(Q_m^i) | y'_i | - x'_i)^2}{2\sigma^2}}, \tag{5}$$

where  $\mathcal{J}(x) = (-1)^{x+1}$ ,  $K$  is the normalization factor  $P_m(Q_m^i = 0 | x_i) + P_m(Q_m^i = 1 | x_i) = 1$ . By integrating the conditional probability into a parameter, we find the soft information called the log likelihood ratio (LLR), which is a very useful parameter for estimation, as follows

$$\ln \frac{P_m(Q_m^i = 0 | x'_i)}{P_m(Q_m^i = 1 | x'_i)} = -\frac{2x'_i | y'_i |}{\sigma^2}. \tag{6}$$

Given the transformation characteristics of the orthogonal rotation process, it is not difficult to deduce  $y'_i = \mu_i ||Y||$ . If estimating  $Q_m$  with Equation (6), Alice needs Bob to send his norm information  $||Y||$ , which will lead to heavy communication traffic and storage resource requirements. Fortunately, we proposed a method that calculates the LLR without using the norm information of an encoder in our previous work [40]. Therefore, our protocol uses this improved method to calculate the LLR of  $Q_m$  as follows

$$LLR(Q_m^i) = S_{nr} ||X|| \ln \frac{1 - v(x'_i)}{1 + v(x'_i)}, \tag{7}$$

where  $S_{nr}$  is the SNR of the quantum channel, and  $v(x'_i) = \frac{x'_i}{||X||}$ .

For the remaining slices  $k (l \leq k < m)$ , we derive their LLR with the corrected slices and the received  $l - 1$  slices as prior information. According to the previous analysis, we find the joint density function of the rotated data  $X'$  and  $Y'$  as Equation (8). Hence, the random variables  $X'$  and  $Y'$  follow the joint density function,

$$f_{X'Y'}(x, y) = \frac{1}{2\pi\delta\sigma} e^{-x^2/2\delta^2} e^{-(y-x)^2/2\sigma^2}. \tag{8}$$

According to Equation (8) and the characteristics of the quantization function Equation (1), we derive that the conditional probability of  $Q_k^i$  is expressed as

$$P_k(Q_k^i = b | x'_i, Q_{1, \dots, k-1, m}^i) = \sum_{\tau} \int_{\tau_{a-1}}^{\tau_a} f_{X'Y'}(x'_i, y) dy, \tag{9}$$

where  $\tau$  represents those quantization intervals satisfying  $\mathcal{S}_{1,\dots,k-1}(y) = B, \mathcal{S}_k(y) = b$ , i.e.,  $\tau = \{(\tau_{a-1}, \tau_a) | \forall y \in (\tau_{a-1}, \tau_a), \mathcal{S}_{1,\dots,k-1,m}(y) = B, \mathcal{S}_k(y) = b\}$ ,  $b = 0$  or  $1$ ,  $B = (Q_{1,\dots,k-1}^i, Q_m^i)$  denotes the disclosed and corrected slices.

Accordingly, we find the initial LLR of  $Q_k^i$  as Equation (10) to preliminarily estimate the rotated results of the  $k$ -th slice,

$$LLR(Q_k^i) = \ln \left( \frac{\sum_{\tau^0}^{\tau_a} \int_{\tau_{a-1}}^{\tau_a} e^{-x_i'^2/2\delta^2} e^{-(y-x_i')^2/2\sigma^2} dy}{\sum_{\tau^1}^{\tau_{a'}} \int_{\tau_{a'-1}}^{\tau_{a'}} e^{-x_i'^2/2\delta^2} e^{-(y-x_i')^2/2\sigma^2} dy} \right), \quad l \leq k < m, \quad (10)$$

where  $\tau^0$  represents the quantization intervals that satisfy  $\mathcal{S}_{1,\dots,k-1,m}(y) = B, \mathcal{S}_k(y) = 0$ , and  $\tau^1$  satisfies  $\mathcal{S}_{1,\dots,k-1,m}(y) = B, \mathcal{S}_k(y) = 1$ , respectively.

Based on the derived LLRs of each slice, the estimator  $\hat{S}$  of our RSEC reconciliation is constructed as follows

$$\hat{S}(Q_j^i) = \begin{cases} 0 & , \text{if } LLR(Q_j^i) > 0 \\ 1 & , \text{otherwise} \end{cases}. \quad (11)$$

Then, Alice can use Equation (11) to construct an initial estimation  $\hat{Q}_j^i = \hat{S}(Q_j^i)$  for Bob's slice value  $Q_j^i$ .

#### 2.4. Error Probability of Slice Estimator

In order to evaluate the accuracy of the proposed slice estimator, we theoretically analyze and compare the error probability of the SEC protocol and the RSEC protocol. The error probability of the slice estimator denotes the theoretical probability that Alice's slice estimator yields a result different from Bob's slice. For the SEC protocol, the error probability in slice  $i$  ( $1 \leq i \leq m$ ) can be expressed as

$$e_i^w = P[\mathcal{S}_i(Y) \neq \hat{\mathcal{S}}_i^w(X)] = \sum_{D_{b_1 \dots b_{i-1}}} (P[\mathcal{S}_i(Y) = 0] \cdot P[\hat{\mathcal{S}}_i^w(X) = 1 | \mathcal{S}_{1 \dots i-1}(Y)] + P[\mathcal{S}_i(Y) = 1] \cdot P[\hat{\mathcal{S}}_i^w(X) = 0 | \mathcal{S}_{1 \dots i-1}(Y)]), \quad (12)$$

where  $b_1 \dots b_{i-1} \in GF(2)^{i-1}$  and  $D_{b_1 \dots b_{i-1}} = \{t | \mathcal{S}_{1 \dots i-1}(t) = b_1 \dots b_{i-1}\}$ , the superscript  $w$  characterize the variables of the SEC protocol. As the random variables  $X$  and  $Y$  follow the Gaussian distribution symmetrical about the coordinate axis, it is easy to obtain that  $P[\mathcal{S}_i(Y) = 0] \cdot P[\hat{\mathcal{S}}_i^w(X, \mathcal{S}_{1 \dots i-1}(Y)) = 1] = P[\mathcal{S}_i(Y) = 1] \cdot P[\hat{\mathcal{S}}_i^w(X, \mathcal{S}_{1 \dots i-1}(Y)) = 0]$ , and then  $e_i^w$  can be further written as

$$e_i^w = 2 \sum_{D_{b_1 \dots b_{i-1}}} P[\mathcal{S}_i(Y) = 0] \cdot P[\hat{\mathcal{S}}_i^w(X) = 1 | \mathcal{S}_{1 \dots i-1}(Y)], \quad (13)$$

each of these terms can be expanded as

$$P[\mathcal{S}_i(Y) = 0] = \int_{\mathcal{A}_{b_1 \dots b_{i-1}0}} \frac{1}{\sqrt{2\pi(\delta^2 + \sigma^2)}} e^{-\frac{y^2}{2(\delta^2 + \sigma^2)}} dy,$$

$$P[\hat{\mathcal{S}}_i^w(X) = 1 | \mathcal{S}_{1 \dots i-1}(Y)] = \int_{\mathcal{B}_{b_1 \dots b_{i-1}}} \frac{1}{\sqrt{2\pi\delta^2}} e^{-\frac{x^2}{2\delta^2}} dx,$$

where  $\mathcal{B}_{b_1 \dots b_{i-1}} = \{x | \int_{\mathcal{A}_{b_1 \dots b_{i-1}1}} f_{XY}(x, y) dy > \int_{\mathcal{A}_{b_1 \dots b_{i-1}0}} f_{XY}(x, y) dy\}$  denotes the set in which  $\hat{\mathcal{S}}_i^w(x) = 1$  has a higher probability than  $\hat{\mathcal{S}}_i^w(x) = 0$ , and  $f_{XY}(x, y) = \frac{1}{2\pi\delta\sigma} e^{-\frac{x^2}{2\delta^2} - \frac{(y-x)^2}{2\sigma^2}}$ ,  $\mathcal{A}_{b_1 \dots b_{i-1}a} = \{y | \mathcal{S}_{1 \dots i-1}(y) = b_1 \dots b_{i-1} \wedge \mathcal{S}_i(y) = a\}$ .

In the RSEC protocol, the quantized values in slice  $m$  are estimated and decoded first. According to the characteristics of the quantization function  $\mathcal{S}$ , Bob's last slice  $Q_m = 0$  when  $y < 0$  and  $Q_m = 1$  otherwise. On the other side, Alice constructs an estimation of Bob's quantized value  $Q_m$  with Equations (7) and (11). The result in the last slice yielded by our estimator  $\hat{\mathcal{S}}(x)$  corresponds to the sign of  $x$ :  $\hat{Q}_m = 0$  when  $x < 0$  and  $\hat{Q}_m = 1$  otherwise. Therefore, the error probability of the RSEC protocol in the last slice  $m$  can be expressed as

$$\begin{aligned} e_m^p &= P[\mathcal{S}_m(Y') \neq \hat{\mathcal{S}}_m(X')] \\ &= P(Y' > 0) \cdot P(X' < 0 | Y' > 0) + P(Y' < 0) \cdot P(X' > 0 | Y' < 0), \end{aligned} \quad (14)$$

the superscript  $p$  characterize the variables of the RSEC protocol. By applying the probability distribution of  $X'$  and  $Y'$  as analyzed in the previous section, we have  $P(Y' > 0) = P(Y' < 0) = \frac{1}{2}$ , and the  $e_m^p$  can be further calculated as follows

$$\begin{aligned} e_m^p &= \frac{1}{2} \int_{-\infty}^0 \frac{1}{\sqrt{2\pi\delta}} e^{-x^2/2\delta^2} dx \int_0^{+\infty} \frac{1}{\sqrt{2\pi\sigma}} e^{-(y-x)^2/2\delta^2} dy + \\ &\quad \frac{1}{2} \int_0^{+\infty} \frac{1}{\sqrt{2\pi\delta}} e^{-x^2/2\delta^2} dx \int_{-\infty}^0 \frac{1}{\sqrt{2\pi\sigma}} e^{-(y-x)^2/2\delta^2} dy \\ &= \int_0^{+\infty} \int_{-\infty}^0 f_{X'Y'}(x, y) dx dy, \end{aligned} \quad (15)$$

where  $f_{X'Y'}(x, y)$  is given in Equation (8), and the second equation in Equation (15) is transformed by using the symmetry of the probability distribution of  $X'$  and  $Y'$ .

Similar to the SEC protocol, the quantized values of the RSEC protocol in other slice  $k$  ( $l \leq k \leq m-1$ ) are estimated by using the disclosed slices and the successfully decoded slices. Thus, the error probability of these slices can be calculated according to Equation (13), where the last slice  $\mathcal{S}_m(Y)$  is taken as additional information.

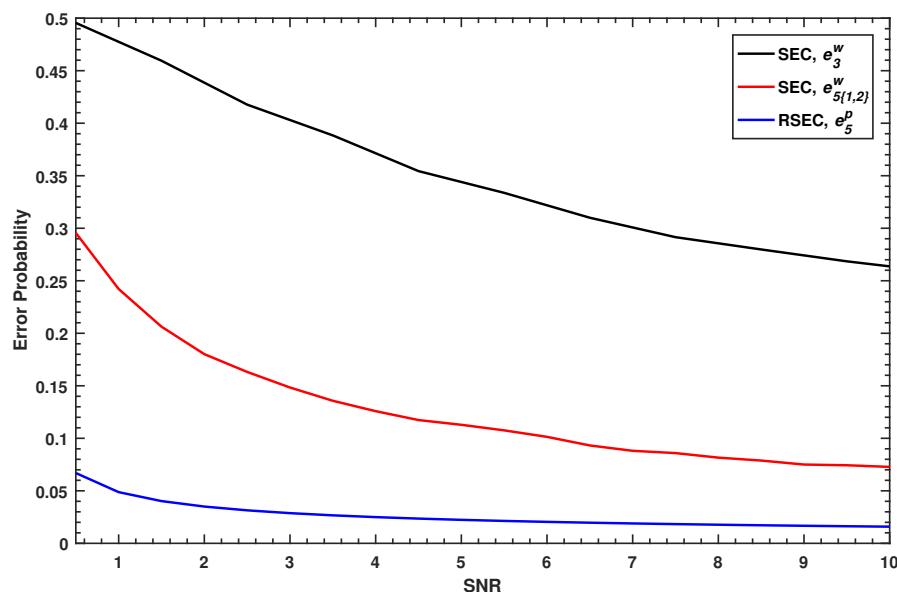
The error probabilities of the subsequent slices—which recursively depend on that of all the previous slices—are not simple to calculate [22]. Here, we compare the estimation accuracy of RSEC protocol with the SEC protocol only by the error probability in the first decoded slice (i.e., the  $l$ -th slice in SEC, the  $m$ -th slice in RSEC). Using the Equations (13) and the (15), we find the error probability in slice  $l = 3$  of the SEC protocol and the error probability in slice  $m = 5$  of RSEC protocol as shown in Figure 3, where the first two slices are disclosed and the quantization adopts the five-slices function.

$e_5^p$  corresponding to the blue curve is always smaller than  $e_3^w$  corresponding to the black curve, i.e., the estimator of RSEC in the first decoded slice always has a lower error probability than that of SEC. If the  $m$ -th slice in the SEC protocol is estimated first, its error probability  $e_{5\{1,2\}}^w$  corresponding to the red curve, of which the expression can be given by Equation (16), also always performs worse than that of the RSEC protocol.

$$e_{5\{1,2\}}^w = 2 \sum_{\mathcal{D}_{b_1 b_2}} P[\mathcal{S}_5(Y) = 0] \cdot P[\hat{\mathcal{S}}_5^w(X) = 1 | \mathcal{S}_{1,2}(Y)]. \quad (16)$$

From the above analysis, it is indicated that the proposed estimator of RSEC protocol has much higher estimation accuracy in the quantization phase than the original estimator of the SEC protocol. Note that the error probability represents the theoretical case of the bit error rate, and the bit error rate of each slice needs to be estimated separately in the practical reconciliation.





**Figure 3.** Comparison of the error probability between the RSEC and the SEC protocol at different SNRs.

2.5. Reconciliation Efficiency

Let us now discuss the reconciliation efficiency of the proposed protocol, which is an important indicator for evaluating the performance of the reconciliation procedure. As is known, the random orthogonal rotation operation on raw data does not expose any information of the rotated results [24]. According to the definition of efficiency in the slice reconciliation [33], the reconciliation efficiency  $\beta$  of the RSEC protocol can be expressed as

$$\beta = \frac{H(\mathcal{S}(Y')) - m + \sum_{i=1}^m R_i}{I(X, Y)}, \tag{17}$$

where  $I(X; Y) = \frac{1}{2} \log_2(1 + S_{nr})$  is the classical capacity of the quantum channel for Gaussian variables,  $m$  denotes the number of slices of quantization function, and  $R_i$  represents the code rate of the error correction scheme of the  $i$ -th slice.  $H(\mathcal{S}(Y'))$  is the entropy of the slice sequences  $\mathcal{S}(Y')$ , which can be calculated as follows

$$H(\mathcal{S}(Y')) = - \sum_a P_a \log_2 P_a, \tag{18}$$

with

$$P_a = \frac{1}{2} \left( \operatorname{erf} \left( \frac{\tau_a}{\sqrt{2(\delta^2 + \sigma^2)}} \right) - \operatorname{erf} \left( \frac{\tau_{a-1}}{\sqrt{2(\delta^2 + \sigma^2)}} \right) \right), \tag{19}$$

where  $\tau_a$  denotes the point dividing the real numbers  $\mathbb{R}$ ,  $1 \leq a \leq 2^m$ , and  $\tau_0 = -\infty$ ,  $\tau_{2^m} = +\infty$ .  $\delta^2$  and  $\sigma^2$  represent Alice’s modulation variance and the noise variance, respectively.

Generally, the code rate of the first  $l - 1$  slices are equal to 0 since they are disclosed via the authentic classical channel.

3. Implementation of RSEC with Polar Codes

After quantizing the continuous variables into strings of bits with slice functions, the legitimate parties are needed to further apply a classical error correction code to complete the reconciliation of the correlated raw data. In this section, we will implement the RSEC protocol with polar codes to distill the correct keys from the correlating raw data.

### 3.1. Review of Polar Codes

The polar code is an error correction code that has been strictly proven to achieve the Shannon capacity [48]. The recursive structure of its encoding and decoding gives them good practical performance. This is relatively easier to construct than another commonly used code, i.e., LDPC code. Therefore, we chose polar codes to implement the RSEC protocol in this research. The RSEC can also be implemented with other error correction codes. Now, we briefly review the encoding and decoding of polar codes in traditional communication.

#### 3.1.1. Encoding

The central idea of polar codes is to convert the  $N$  individual copies of the channel  $W$  into two different types of channels, i.e., error-free channel and completely noisy channel, through an operation called channel polarization—channel combining and channel splitting. The information sender chooses the positions corresponding to the error-free channel to place her message bits (called information bits), and usually sets the remaining positions corresponding to the completely noisy channel as 0 (called frozen bits). The information bits and frozen bits together form a sequence  $u_1^N$  of  $N$  bits. We use the notation  $u_1^n = (u_1, \dots, u_n)$  to denote a row vector of  $n$  bits. The sender encodes the sequence  $u_1^N$  to a codeword  $x_1^N$  by

$$x_1^N = u_1^N G, \tag{20}$$

where  $G$  is the generator matrix and defined as  $G = F^{\otimes \log_2 N} B$ ,  $F^{\otimes n}$  means to perform the Kronecker product  $n$  times on the matrix  $F \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ , and  $B$  is a permutation matrix for executing the bit-reversal operation [48]. Obtaining the codeword  $x_1^N$ , the sender transmits it to the information receiver for decoding.

#### 3.1.2. Decoding

After the codeword  $x_1^N$  is transmitted through the channel, the receiver obtains a sequence  $y_1^N$ , which is a noise version of  $x_1^N$ . Then, he uses successive cancellation (SC) or successive cancellation list (SCL) decoding algorithms to correct the error bits among  $y_1^N$  with the given frozen bits. We here describe the receiver’s decoding process with a SC decoding algorithm [48]:

1. Initialize the received information  $y_1^N$  with channel transition probability  $W(y|x)$  as

$$L_1^{(j)}(y_j) = \frac{W(y_j|0)}{W(y_j|1)}, \quad j = 1, 2, \dots, N. \tag{21}$$

2. Calculate the likelihood ratio (LR) of  $u_j$  with the decoding results  $\hat{u}_1^{j-1} = (\hat{u}_1, \hat{u}_2, \dots, \hat{u}_{j-1})$  of the previous  $j - 1$  bits as follows

$$L_N^{(j)}(y_j, \hat{u}_1^{j-1}) = \frac{W_N^{(j)}(y_1^N, \hat{u}_1^{j-1} | u_j = 0)}{W_N^{(j)}(y_1^N, \hat{u}_1^{j-1} | u_j = 1)}, \tag{22}$$

where

$$W_N^{(j)}(y_1^N, u_1^{j-1} | u_j) \triangleq \frac{1}{2^{N-1}} \sum_{u_{j+1}^N \in \{0,1\}^{N-j}} W_N(y_1^N | u_1^N), \tag{23}$$

and

$$W_N(y_1^N | u_1^N) = W^N(y_1^N | x_1^N = u_1^N G) = \prod_{i=1}^N W(y_i | x_i). \tag{24}$$

3. Generate the decision  $\hat{u}_j$  of  $u_j$  as

$$\hat{u}_j = \begin{cases} u_j & , \text{if } j \in \mathcal{A} \\ 0 & , \text{if } j \notin \mathcal{A} \text{ and } L_N^{(j)}(y_1^N, \hat{u}_1^{j-1}) \geq 1 \\ 1 & , \text{if } j \notin \mathcal{A} \text{ and } L_N^{(j)}(y_1^N, \hat{u}_1^{j-1}) < 1 \end{cases} \quad (25)$$

where  $\mathcal{A}$  is the position set of the frozen bits.

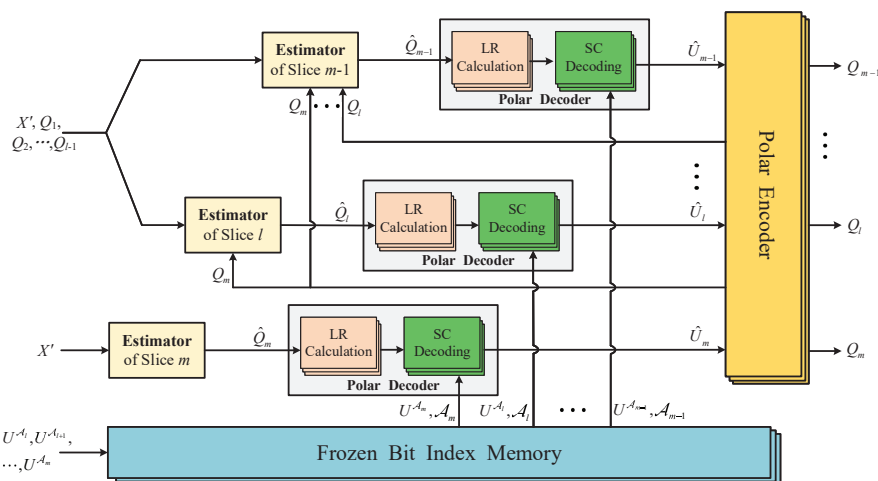
After getting the  $j$ -th bit by step (iii), the process returns to step (ii) to decode the  $(j + 1)$ -th bit.

### 3.2. Implementation Process

The reconciliation mode of CV-QKD is different from traditional communication. In traditional communication, the codeword is mixed with noises during the reconciliation. However, in a CV-QKD system, the two parties have already shared inconsistent data before the post-processing phase, in other words, the noise in the codeword appeared before the reconciliation. Therefore, in order to correct the slice sequences of RSEC, it is necessary to establish a virtual channel for Alice and Bob to deal with the noise.

The encoding of polar codes is reversible: Encoding an input sequence  $x$  twice, one can recover this sequence, i.e.,  $xGG = x$ . This property can be used to establish a virtual channel as: Bob encodes a slice sequence  $x$  to another sequence  $u = xG$ , and then sends the bits  $u_{\mathcal{A}}$  corresponding to the frozen indices to Alice. Since  $uG = (xG)G = x$ , the slice sequence  $x$  can be regarded as a polar codeword, Alice's initial estimation  $\hat{S}(x)$  of Bob's slice value can be viewed as the received codeword, and  $u_{\mathcal{A}}$  corresponds to the frozen bits shared by the two parties. Therefore, a virtual channel can be established by using the above method.

Before launching the reconciliation with polar codes, Alice and Bob determine the code rate  $R'_i$  of each slice according to the SNR and share the corresponding frozen index set  $\mathcal{A}_i$ . The frozen index set can be selected by a construction algorithm with consideration to  $R'_i$ . Then, the logic structure of the RSEC reconciliation with polar codes is shown in Figure 4, in which the detailed implementation process is described as follows:



**Figure 4.** Logic structure of the RSEC reconciliation with polar codes. The LR calculation modules provide the initial LR  $L_{1,i}^{(j)}(\hat{q}_j)$ , and the frozen bit index memory stores the frozen bit  $U^{A_i}$  and frozen index set  $\mathcal{A}_i$ .

**Step 1:** Alice and Bob convert their correlated data  $X, Y$  to another continuous-variable sequence noted as  $X', Y'$  with random orthogonal rotation according to RSEC. Bob then quantizes  $Y'$  into  $m$  slice sequences  $Q_1, Q_2, \dots, Q_m$  with the slice function and sends the

first  $l - 1$  slices  $Q_1, Q_2, \dots, Q_{l-1}$  to Alice. Afterward, they begin to reconcile the remaining slice sequences with polar codes in the order of  $m, l, l + 1, \dots, m - 1$  slice.

**Step 2:** Alice uses the proposed estimator in Equation (11) to construct a bit string  $\hat{Q}_i$  corresponding to Bob's slice sequence  $Q_i$ . Meanwhile, Bob encodes his slice sequence to  $U = Q_i G$ , and sends the bits  $U^{A_i}$  at the frozen positions to Alice.

**Step 3:** Alice calculates the initial LR  $L_{1,i}^{(j)}(\hat{q}_j)$  as Equation (26),  $j = 1, 2, \dots, N$ , and then makes a decision  $\hat{U}$  on  $U$  after getting the final LR  $L_{N,i}^{(j)}(\hat{q}_1^N, \hat{u}_1^{j-1})$  in Equation (27). Afterward, she can recover Bob's sequence  $Q_i$  with a high probability by executing an encoding operation on  $\hat{U}$ .

$$L_{1,i}^{(j)}(\hat{q}_j) = \frac{W(\hat{q}_j|0)}{W(\hat{q}_j|1)}, \quad j = 1, 2, \dots, N, \tag{26}$$

where  $L_{1,i}^{(j)}(\hat{q}_j)$  is the initial LR corresponding to the  $j$ -th bit of  $U$ ,  $\hat{q}_j$  is the  $j$ -th bit of  $\hat{Q}_i$ ,  $U = (u_1, \dots, u_N)$ ,  $\hat{U} = (\hat{u}_1, \dots, \hat{u}_N)$ , the channel transition probability can be calculated as: if  $y = x$ ,  $W(y|x) = 1 - e_i$ , if  $y \neq x$ ,  $W(y|x) = e_i$ . The bit error rate  $e_i$  can be estimated in the stage of parameter estimation by executing the quantization operation on the extra raw data.

$$L_{N,i}^{(j)}(\hat{q}_1^N, \hat{u}_1^{j-1}) = \frac{W_N^{(j)}(\hat{q}_1^N, \hat{u}_1^{j-1}|0)}{W_N^{(j)}(\hat{q}_1^N, \hat{u}_1^{j-1}|1)}. \tag{27}$$

Moreover, Equation (27) can evolve in a recursive manner as if  $j$  is odd, i.e.,  $j = 2k - 1$ , then

$$L_{N,i}^{(2k-1)}(\hat{q}_1^N, \hat{u}_1^{2k-2}) = f\left(L_{N/2,i}^{(k)}(\hat{q}_1^{N/2}, \hat{u}_{1,o}^{2k-2} \oplus \hat{u}_{1,e}^{2k-2}), L_{N/2,i}^{(k)}(\hat{q}_{N/2+1}^N, \hat{u}_{1,e}^{2k-2})\right), \tag{28}$$

if  $j$  is even, i.e.,  $j = 2k$ , then

$$L_{N,i}^{(2k)}(\hat{u}_1^N, \hat{u}_1^{2k-1}) = g\left(L_{N/2,i}^{(j)}(\hat{q}_1^{N/2}, \hat{u}_{1,o}^{2k-2} \oplus \hat{u}_{1,e}^{2k-2}), L_{N/2,i}^{(k)}(\hat{q}_{N/2+1}^N, \hat{u}_{1,e}^{2k-2}), \hat{u}_{2k-1}\right), \tag{29}$$

where  $f(a, b) = \frac{a \cdot b + 1}{a + b}$ ,  $g(a, b, s) = a^{1-2s} \cdot b$ , we use  $x_{a,o}^b$  to denote the odd terms of  $x_a^b$ , and  $x_{a,e}^b$  denotes the even terms of  $x_a^b$ .

Alice can also use LLR as the soft information of polar codes for decoding. In this case, the initial LLR is calculated according to Equations (7) and (10).

In order to ensure that the equation  $\hat{U}G = Q_i$  holds with a high probability, Alice and Bob need to perform a cyclic redundancy check (CRC) to verify the decoding result  $\hat{U}$ . If  $\hat{U}$  fails to pass CRC check, Alice and Bob give up on this slice  $Q_i$ . Even if the decoding result passes the CRC check, undetected error bits may still exist. However, this situation rarely occurs and can be overlooked.

As the CRC values will leak the information about  $Q_i$ , it is necessary to discard them. Therefore, the code rate  $R_i$  of each slice is calculated as follows

$$R_i = R_i' - \frac{n_{crc}}{N}, \tag{30}$$

where  $n_{crc}$  is the length of the CRC values.

#### 4. Experiment Results and Analysis

To evaluate the performance of the RSEC protocol, we performed a series of experiments to compare their performances, including the quantization efficiency, reconciliation efficiency, and the secret key rate.

#### 4.1. Quantization Efficiency of RSEC

The principle of quantization is to minimize the information loss so that  $I(X'; \mathcal{S}(Y'))$  can be made arbitrarily close to the initially shared information  $I(X; Y)$ . After quantization, the mutual information  $I(X'; \mathcal{S}(Y'))$  shared by Alice and Bob can be expressed as

$$I(X'; \mathcal{S}(Y')) = H(\mathcal{S}(Y')) - \left[ H(\mathcal{S}_m(Y')|X') + \sum_{i=1}^{m-1} H(\mathcal{S}_i(Y')|X', \mathcal{S}_{1\dots i-1,m}(Y')) \right], \quad (31)$$

where  $\mathcal{S}(Y') = (\mathcal{S}_1(Y'), \dots, \mathcal{S}_m(Y'))$  are the slice values of Bob.

As the conditional entropy of Equation (31) recursively depends on all previously estimated results, calculating  $I(X'; \mathcal{S}(Y'))$  is not a simple task. For this reason, it is common practice to replace the conditional entropy with  $H(e_i)$  equivalently [22]. Then, the goal of quantization is simply to minimize each  $e_i$ , of which  $H(e_i)$  is an increasing function for  $0 \leq e_i < 0.5$ ,  $e_i$  is the BER of  $i$ -th slice. Therefore, the quantization efficiency  $\beta_s$  can be measured with Equation (32) equivalently [22],

$$\beta_s = \frac{H(\mathcal{S}(Y')) - H_e}{I(X; Y)}, \quad (32)$$

with  $H_e = \sum_{i=1}^m H(e_i)$ ,  $H(e_i) = -e_i \log_2(e_i) - (1 - e_i) \log_2(1 - e_i)$ .

Figure 5 shows the quantization efficiency curves of SEC and RSEC at different SNR when  $m = 4$  and  $m = 5$ . As can be seen from the figure, the quantization efficiency of SEC drops sharply for  $\text{SNR} < 3$ , which confirms that SEC reconciliation usually performs poorly at low SNRs. By contrast, RSEC can still maintain a high quantization efficiency  $\beta_s > 96\%$  for almost all SNRs  $< 3$ , and even achieves above 99% quantization efficiency in the range of  $\text{SNR} \in (1, 3)$  when adopting the five-slice scheme. The primary reason for the better performance of the proposed RSEC over the SEC protocol is attributed to our new estimator.

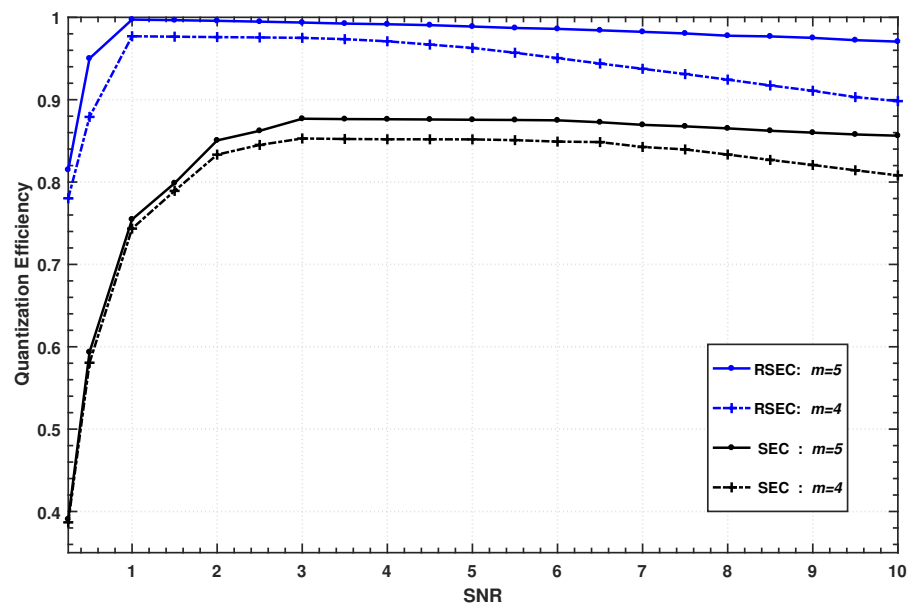


Figure 5. The quantization efficiency of the RSEC and the SEC protocol with four and five slices at different SNR. The upper two curves denote the values of the RSEC protocol; the lower two curves correspond to the values of the SEC protocol.

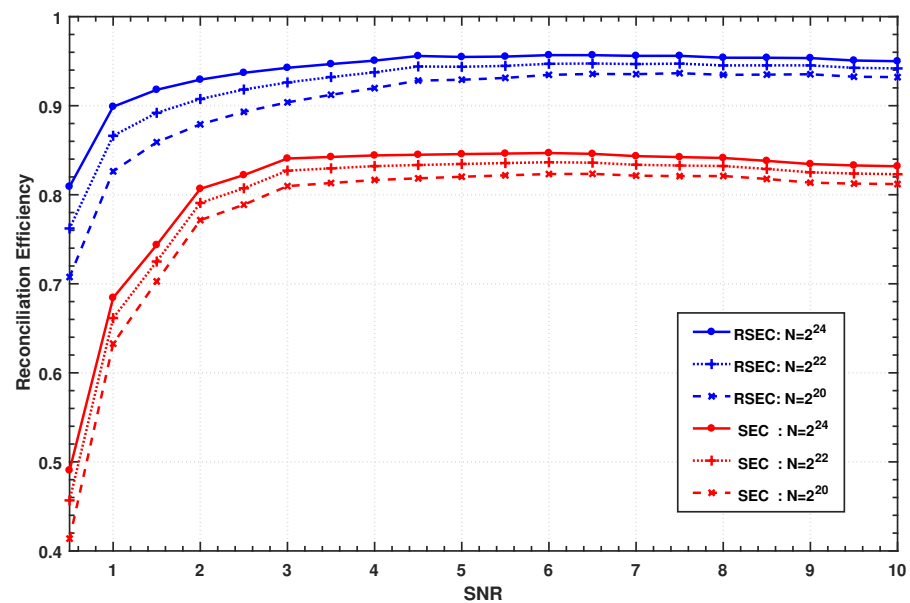
With the orthogonal rotation, our estimator can estimate Bob’s slice sequences more accurately, especially for the slice that is decoded first, and thus the error rate  $e_i$  decreases accordingly. Moreover, the result in Figure 5 also confirms the following basic facts.

For a fixed SNR, the higher the number of slices, the lower the information loss caused by quantization.

#### 4.2. Reconciliation Efficiency of RSEC with Polar Codes

It appears that the polarization speed of polar codes is highly dependent on the channel [49]. Compared with the Binary Input Additive White Gaussian Noise Channel (BIAWGNC), constructing polar code for a Binary Symmetric Channel (BSC) is relatively uncomplicated and more common. Moreover, a BSC can be established between the two parties if Alice makes an initial estimation of Bob's slice sequences using LLR values. Accordingly, in our experiments, we construct the polar codes on a BSC and calculate the initial LR as Equation (26) for decoding the slice sequences.

Figure 6 compares the reconciliation efficiencies of the RSEC and the SEC protocol with polar codes when  $m = 5$ . The 32-bit CRC is adopted for polar codes to check the decoding results, i.e.,  $n_{crc} = 32$ , and the eight-dimensional orthogonal matrix is used in rotation. For a fixed SNR value and different block length, 1000 blocks of raw data are generated to measure the reconciliation performance. The experimental results are obtained with frame error rate (FER)  $\leq 0.1$ , but a null BER in the blocks decoded successfully.



**Figure 6.** The reconciliation efficiency of the RSEC and SEC protocol with polar codes at different SNRs when the number of slices  $m = 5$ . The upper three curves show the values of the RSEC protocol; the lower three curves correspond to the values of the SEC protocol. For the RSEC and SEC protocols, their three curves from bottom to top represent the reconciliation efficiencies obtained with  $N = 2^{20}$ ,  $2^{22}$ ,  $2^{24}$ , respectively.

Combining Figures 5 and 6, it is not difficult to find that the curvilinear trend of the quantization efficiency is essentially consistent with that of the reconciliation efficiency; this is because the reconciliation scheme with good quantization performance usually performs better in reconciliation. Hence, the reconciliation efficiency of the proposed protocol is higher than the SEC protocol over the entire range in Figure 6 due to its higher quantization efficiency.

As shown in Figure 6, both the reconciliation efficiencies of RSEC and SEC increase with the increasing block length of polar codes since the decoding performance of polar codes will become better with the increase of its block size. The proposed RSEC protocol has an efficiency above 90% over almost the entire range  $\text{SNR} \geq 1$  for the block lengths starting from  $2^{24}$ , and even exceeds 95% at  $\text{SNR} > 3$ , which allows the system to distill more than 1 bit corrected key per raw data.

RSEC has a high quantization efficiency in the SNR range (1,3), whereas its reconciliation efficiency is not so perfect. The reason is that the relatively low SNR leads to a high BER > 10% in some noisy slices, and the decoding performance of polar codes decreases at high BER [50]. In fact, the high quantization efficiency of RSEC allows the system to achieve a higher reconciliation efficiency by using a high-performance code.

In addition, we compare the reconciliation efficiency values with the representative works on SEC in Table 1. As shown in the table, the proposed protocol improves almost all previously published reconciliation efficiencies in terms of the SEC protocol in the high SNR regime, which is the main focus of the SEC reconciliation. In fact, the reconciliation efficiency values of [33] listed in the table are obtained under an optimistic situation of adopting the optimal number of slices and specially designed high-performance codes. Nevertheless, our reconciliation scheme still has a competitive advantage over [33] on the whole.

**Table 1.** Comparison of the reconciliation efficiencies between RSEC and some representative reconciliation works.

SNR	Reconciliation Efficiency $\beta$			
	Ref. [33] <sup>a</sup>	Ref. [23] <sup>b</sup>	Ref. [29] <sup>c</sup>	This Work
3	94.1%	79%	88.7%	94.85%
5.12	94.4%	—	—	95.53%
7	—	84%	—	95.60%
14.57	95.8%	—	—	95.02%

<sup>a</sup> The slice number and error correction codes adopted in [33] are not reported in detail; <sup>b</sup> It implements the four-slice and five-slice SEC with the LDPC for blocks of  $2 \times 10^5$ ; <sup>c</sup> It implements the four-slice SEC with the LDPC and Bose–Chaudhuri–Hocquenghem (BCH) for blocks of  $2 \times 10^5$  in a 25 km all-fiber CV-QKD system.

Many achievements have also been made in multidimensional reconciliation, for example, [38] implements eight-dimensional reconciliation with  $\beta = 99\%$  and FER = 0.883 using QC MET-LDPC code at SNR = 0.0283, and [39] achieves  $\beta = 93.40\%$ , 95.84%, 96.99% and FER  $\leq 0.375$  with eight-dimensional reconciliation based on the MET-LDPC code at SNRs of = 0.160, 0.075, 0.029, respectively. However, unlike the SEC protocol, the multidimensional reconciliation protocol is more suitable for the low SNRs rather than the high SNR regime. The existing works on multidimensional reconciliation are aimed at the extremely low SNRs and rarely provide the experimental results in the high SNR regime. Therefore, we mainly give a comparison with the representative results of the SEC protocol.

#### 4.3. Secret Key Rate of RSEC

Assuming a collective Gaussian attack and accounting for the finite-size effects, the secret key rate of a CV-QKD system with reverse reconciliation can be expressed as [29]:

$$K_{finite} = \frac{N_{data}}{N_{total}} [\beta I_{AB} - \chi_{BE} - \Delta(N_{data})], \quad (33)$$

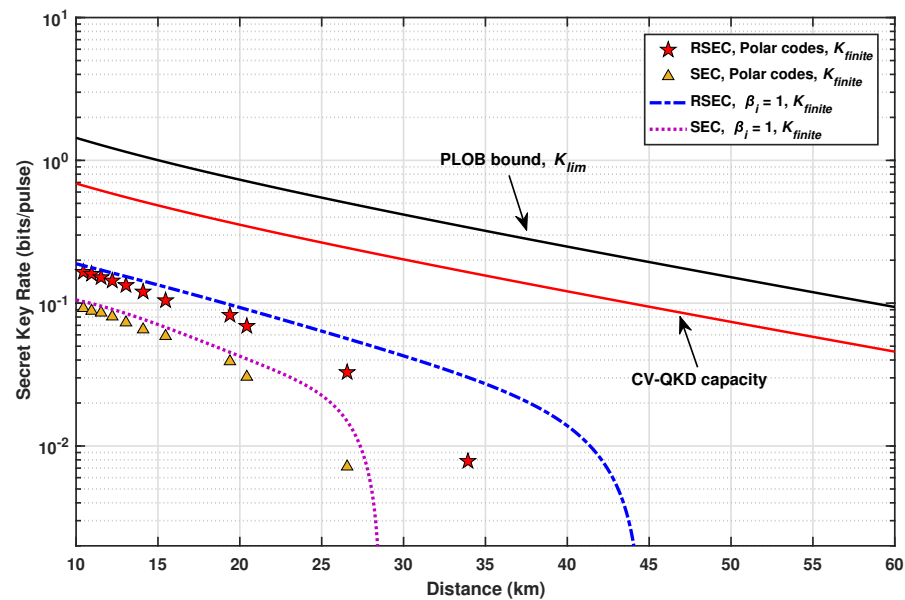
where  $N_{total}$  is the total number of symbols sent from Alice to Bob,  $N_{data}$  is the number of raw data used for key distillation,  $\beta$  is the reconciliation efficiency,  $I_{AB}$  denotes the mutual information between Alice and Bob,  $\chi_{BE}$  denotes the Holevo bound on the information that Eve can obtain, and  $\Delta(N_{data})$  is the finite-size offset factor.  $I_{AB}$  and  $\chi_{BE}$  are related to the physical parameters, including the transmittance  $T$ , the total noise  $\chi_{total}$ , and Alice's modulation variance  $V_A$ . The transmittance  $T$  of the quantum channel is defined as  $T = 10^{-\alpha\mathcal{L}/10}$ , where  $\alpha$  is the single-mode fiber transmission loss and  $\mathcal{L}$  is the transmission distance.

The total noise  $\chi_{total}$  consists of the channel added noise and the noise generated by Bob's detector, and be given by  $\chi_{total} = \chi_{line} + \frac{\chi_{hom}}{T}$ , where  $\chi_{line} = (\frac{1}{T} - 1) + \zeta$  and  $\chi_{hom} = \frac{1+V_{el}}{\eta} - 1$ ,  $\zeta$  is the excess channel noise,  $V_{el}$  denotes the added electronic noise of

Bob’s detector, and  $\eta$  represents the detector efficiency. The detailed calculation about  $I_{AB}$ ,  $\chi_{BE}$ , and  $\Delta(N_{data})$  can be found in Appendix A.

In our simulation, the experimental physical parameters reported in previously published work [29] were used to characterize the CV-QKD system and quantum channel. Optimizing the modulation variance  $V_A$  for each transmission distance can maximize the SNR of a quantum channel. The modulation variance  $V_A$  in our work is adjusted according to [38]. We chose  $N_{total} = 2N_{data}$  and the security parameter of  $10^{-10}$  for  $\Delta(N_{data})$  [18].

Figure 7 presents the finite secret key rates  $K_{finite}$  over the transmission distances with  $N_{finite} = 2^{40}$  bits. The five-pointed stars and triangle points compare the secret key rates achieved with polar codes of block lengths  $N = 2^{24}$  bits, where the CV-QKD system using RSEC always provides higher secret key rates than that using SEC at the same transmission distance. Using RSEC reconciliation, we achieve a secret key rate of  $7.83 \times 10^{-3}$  bits/pulse at a distance of 33.93 km, while the CV-QKD system using SEC cannot provide any secret key.



**Figure 7.** Finite secret key rate with  $N_{total} = 2^{40}$  vs. distance. The five-pointed stars correspond to the secret key rates using five-slice RSEC with polar codes of  $N = 2^{24}$ ; the triangle points correspond to the values using five-slice SEC with polar codes of  $N = 2^{24}$ ; the blue dotted line and purple dotted line represent the asymptotic theoretical secret key rates using five-slice RSEC and SEC with a perfect error correction scheme (i.e.,  $\beta_i = 1$ ), respectively. Other parameters are as follows [29]:  $\alpha = 0.2\text{dB/km}$ ,  $\zeta = 0.005$ ,  $V_{el} = 0.041$ ,  $\eta = 0.606$ .

In particular, assuming perfect error correction in the decoding of each slice, the RSEC protocol has more obvious advantages than the SEC protocol in the asymptotic secret key rate as the two dotted lines in Figure 7. A perfect error correction scheme allows each slice to achieve its Shannon capacity, i.e., the efficiency  $\beta_i$  of error correction in each slice is assumed as 1. With the increase of the transmission distance, the secret key rate of CV-QKD decreases. This is because the SNR becomes lower with the increase of transmission distance, which leads to the reduction of the quantification efficiency. Notably, when the transmission distance increases to about 30 km, the CV-QKD system using the SEC protocol can hardly generate any secret key. However, the RSEC protocol can theoretically extend the secure distance of the CV-QKD system to about 45 km.



There exists an upper bound called the Pirandola–Laurenza–Ottaviani–Banchi (PLOB) bound for the secret-key capacity of a lossy channel. The PLOB bound  $K_{lim}$  is determined by the transmittance  $T$  of channel and is given by [51]

$$K_{lim} = -\log_2(1 - T). \quad (34)$$

The black solid line in Figure 7 is the PLOB bound, which sets the fundamental rate limit for point-to-point QKD in the presence of loss. It is almost non-achievable for current protocols in the practical systems. Assuming the infinite-size keys and ideal conditions (such as unit detector efficiencies, zero dark count rates, zero intrinsic error, unit error correction efficiency, and zero excess noise), the maximum rate of the CV-QKD protocol (the red solid line) scales as  $T/\ln 4$ , which is 1/2 of the PLOB bound [51].

If taking the finite-size effect and the non-ideal factors of physical devices into account, the secret key rate of the practical CV-QKD systems will be much lower. As shown in Figure 7, considering the non-ideal condition, the finite secret key rate of the CV-QKD system using RSEC can achieve  $3.28 \times 10^{-2} \sim 1.652 \times 10^{-1}$  bits/pulse at  $10 \sim 27$  km, which is about 0.115 of the PLOB bound. However, the system using SEC has a lower rate, which is just about 0.064 of the PLOB bound, at  $7.1 \times 10^{-3} \sim 9.22 \times 10^{-2}$  bits/pulse.

The previous experimental results indicate that the proposed RSEC protocol is clearly advantageous. It significantly improves the quantization and reconciliation efficiency of SEC, which enables the CV-QKD system to achieve a higher secret key rate and a longer secure transmission distance. Overall, our work provides a better candidate for the application of the CV-QKD system.

## 5. Conclusions

In this research, we analyzed the strategy of the SEC protocol and proposed modifications to improve its anti-noise ability by performing a random orthogonal rotation on the correlated raw data and deducing a slice estimator. The experimental comparisons of the original SEC protocol and the proposed RSEC protocol show that the modifications can reduce the information loss of the quantization and release the performance limitation of SEC at the relatively low SNR.

Accordingly, both the secret key rate and the range of CV-QKD are increased. In order to accomplish the reconciliation of the raw data in CV-QKD, we implemented the RSEC protocol by combining with the polar codes. The reconciliation efficiency of RSEC protocol achieved above 95% when the input scale adopted 16 Mb. Both theoretical and experimental analysis showed that this is a more suitable reconciliation scheme for a practical CV-QKD system.

**Author Contributions:** Conceptualization, X.W. (Xuan Wen) and Q.L.; methodology, X.W. (Xuan Wen); software, X.W. (Xuan Wen), H.M. and X.W. (Xiaojun Wen); validation, X.W. (Xuan Wen), Q.L., H.M. and X.W. (Xiaojun Wen); formal analysis, X.W. (Xuan Wen) and Q.L.; investigation, X.W. (Xuan Wen), Q.L. and X.W. (Xiaojun Wen); resources, X.W. (Xuan Wen) and Q.L.; data curation, Q.L. and X.W. (Xiaojun Wen); writing—original draft preparation, X.W. (Xuan Wen); writing—review and editing, N.C., Q.L., H.M. and X.W. (Xiaojun Wen); supervision, Q.L. and X.W. (Xiaojun Wen). All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China grant number 61301099 and 62071151; by the Space Science and Technology Advance Research Joint Funds grant number 6141B06110105, and by the Shenzhen Discipline Layout Project grant number JCYJ20170815145900474.

**Data Availability Statement:** Data is contained within the article.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

The mutual information between Alice and Bob  $I_{AB}$  can be calculated by using Shannon's channel capacity [29],

$$I_{AB} = \frac{1}{2} \log_2(1 + S_{nr}) = \frac{1}{2} \log_2\left(\frac{V + \chi_{total}}{1 + \chi_{total}}\right), \quad (A1)$$

where  $V = V_A + 1$ ,  $V_A$  represents Alice's modulation variance, and  $\chi_{total} = \chi_{line} + \chi_{hom}/T$  represents the total noise between Alice and Bob as previously defined.

The Holevo bound on information available to Eve is given by

$$\chi_{BE} = G\left(\frac{\lambda_1 - 1}{2}\right) + G\left(\frac{\lambda_2 - 1}{2}\right) - G\left(\frac{\lambda_3 - 1}{2}\right) - G\left(\frac{\lambda_4 - 1}{2}\right), \quad (A2)$$

where  $G(x) = (x + 1)\log_2(x + 1) - x\log_2(x)$ , and the symplectic eigenvalues  $\lambda_{1,2,3,4}$  are given by

$$\lambda_{1,2}^2 = \frac{1}{2}(A \pm \sqrt{A^2 - 4B}), \lambda_{3,4}^2 = \frac{1}{2}(C \pm \sqrt{C^2 - 4D}), \quad (A3)$$

with

$$A = V^2(1 - 2T) + 2T + T^2(V + \chi_{line})^2, B = T^2(V\chi_{line} + 1)^2, \quad (A4)$$

$$C = \frac{V\sqrt{B} + T(V + \chi_{line}) + A\chi_{hom}}{T(V + \chi_{total})}, D = \frac{V\sqrt{B} + B\chi_{hom}}{T(V + \chi_{total})}. \quad (A5)$$

When  $N_{data} > 10^4$ , the finite-size offset factor  $\Delta(N_{data})$  can be approximated as follows[18],

$$\Delta(N_{data}) \approx 7\sqrt{\frac{\log_2(2/\epsilon)}{N_{data}}}, \quad (A6)$$

where  $\epsilon$  is the security parameter.

## References

1. Pirandola, S.; Andersen, U.L.; Banchi, L.; Berta, M.; Bunandar, D.; Colbeck, R.; Englund, D.; Gehring, T.; Lupo, C.; Ottaviani, C.; et al. Advances in quantum cryptography. *Adv. Opt. Photonics* **2020**, *12*, 1012–1236. [CrossRef]
2. Bennett, C.H. Quantum cryptography: Public key distribution and coin tossing. In Proceedings of the IEEE International Conference on Computers, Bangalore, India, 10–12 December 1984; pp. 175–179.
3. Bennett, C.H. Quantum cryptography using any two nonorthogonal states. *Phys. Rev. Lett.* **1992**, *68*, 3121. [CrossRef] [PubMed]
4. Lo, H.K.; Curty, M.; Bing, Q. Measurement-Device-Independent Quantum Key Distribution. *Phys. Rev. Lett.* **2012**, *108*, 130503. [CrossRef] [PubMed]
5. Lucamarini, M.; Yuan, Z.L.; Dynes, J.F.; Shields, A.J. Overcoming the rate–distance limit of quantum key distribution without quantum repeaters. *Nature* **2018**, *557*, 400–403. [CrossRef] [PubMed]
6. Wang, X.B.; Yu, Z.W.; Hu, X.L. Twin-field quantum key distribution with large misalignment error. *Phys. Rev. A* **2018**, *98*, 062323. [CrossRef]
7. Grosshans, F.; Grangier, P. Continuous Variable Quantum Cryptography Using Coherent States. *Phys. Rev. Lett.* **2002**, *88*, 057902. [CrossRef] [PubMed]
8. Weedbrook, C.; Lance, A.M.; Bowen, W.P.; Symul, T.; Ralph, T.C.; Lam, P.K. Quantum cryptography without switching. *Phys. Rev. Lett.* **2004**, *93*, 170504. [CrossRef] [PubMed]
9. Pirandola, S.; Mancini, S.; Lloyd, S.; Braunstein, S.L. Continuous-variable quantum cryptography using two-way quantum communication. *Nat. Phys.* **2008**, *4*, 726. [CrossRef]
10. Li, Z.; Zhang, Y.C.; Xu, F.; Peng, X.; Guo, H. Continuous-variable measurement-device-independent quantum key distribution. *Phys. Rev. A* **2014**, *89*, 052301. [CrossRef]
11. Djordjevic, I.B. Optimized-Eight-State CV-QKD Protocol Outperforming Gaussian Modulation Based Protocols. *IEEE Photonics J.* **2019**, *11*, 1–10. [CrossRef]
12. Leverrier, A.; Grangier, P. A simple proof that Gaussian attacks are optimal among collective attacks against continuous-variable quantum key distribution with a Gaussian modulation. *Phys. Rev. A* **2010**, *81*, 2112–2114. [CrossRef]
13. Ottaviani, C.; Mancini, S.; Pirandola, S. Two-way Gaussian quantum cryptography against coherent attacks in direct reconciliation. *Phys. Rev. A* **2015**, *92*, 062323. [CrossRef]
14. Ottaviani, C.; Pirandola, S. General immunity and superadditivity of two-way Gaussian quantum cryptography. *Sci. Rep.* **2016**, *6*, 22225. [CrossRef] [PubMed]

15. Leverrier, A.; Anthony, S. Security of Continuous-Variable Quantum Key Distribution via a Gaussian de Finetti Reduction. *Phys. Rev. Lett.* **2017**, *118*, 200501. [[CrossRef](#)] [[PubMed](#)]
16. Zhang, G.; Haw, J.Y.; Cai, H.; Xu, F.; Assad, S.M.; Fitzsimons, J.F.; Zhou, X.; Zhang, Y.; Yu, S.; Wu, J. An integrated silicon photonic chip platform for continuous-variable quantum key distribution. *Nat. Photonics* **2019**, *13*, 839–842. [[CrossRef](#)]
17. Kraus, B.; Gisin, N.; Renner, R. Lower and upper bounds on the secret-key rate for quantum key distribution protocols using one-way classical communication. *Phys. Rev. Lett.* **2005**, *95*, 080501. [[CrossRef](#)]
18. Leverrier, A.; Grosshans, F.; Grangier, P. Finite-size analysis of a continuous-variable quantum key distribution. *Phys. Rev. A* **2010**, *81*, 062343. [[CrossRef](#)]
19. Papanastasiou, P.; Ottaviani, C.; Pirandola, S. Finite size analysis of measurement device independent quantum cryptography with continuous variables. *Phys. Rev. A* **2017**, *96*, 042332. [[CrossRef](#)]
20. Chen, Z.; Zhang, Y.; Wang, X.; Yu, S.; Guo, H. Improving Parameter Estimation of Entropic Uncertainty Relation in Continuous-Variable Quantum Key Distribution. *Entropy* **2019**, *21*, 652. [[CrossRef](#)]
21. Silberhorn, C.; Korolkova, N.; Leuchs, G. Quantum Key Distribution with Bright Entangled Beams. *Phys. Rev. Lett.* **2002**, *88*, 167902. [[CrossRef](#)]
22. Van Assche, G.; Cardinal, J.; Cerf, N.J. Reconciliation of a Quantum-Distributed Gaussian Key. *IEEE Trans. Inf. Theory* **2012**, *50*, 394–400. [[CrossRef](#)]
23. Bloch, M.; Thangaraj, A.; Laughlin, S.W.M.; Merolla, J.M. LDPC-based Gaussian key reconciliation. In Proceedings of the IEEE Information Theory Workshop, Punta del Este, Uruguay, 13–17 March 2006; pp. 116–120.
24. Leverrier, A.; Alléaume, R.; Boutros, J.; Zémor, G.; Grangier, P. Multidimensional reconciliation for a continuous-variable quantum key distribution. *Phys. Rev. A* **2008**, *77*, 042325. [[CrossRef](#)]
25. Wen, X.; Li, Q.; Mao, H.; Luo, Y.; Yan, B.; Huang, F. Novel reconciliation protocol based on spinal code for continuous-variable quantum key distribution. *Quantum Inf. Process.* **2020**, *19*, 850. [[CrossRef](#)]
26. Bennett, C.H.; Brassard, G.; Crépeau, C.; Maurer, U.M. Generalized privacy amplification. *IEEE Trans. Inf. Theory* **1995**, *41*, 1915–1923. [[CrossRef](#)]
27. Tang, B.Y.; Liu, B.; Zhai, Y.P.; Wu, C.Q.; Yu, W.R. High-speed and Large-scale Privacy Amplification Scheme for Quantum Key Distribution. *Sci. Rep.* **2019**, *9*, 15733. [[CrossRef](#)] [[PubMed](#)]
28. Yan, B.; Li, Q.; Mao, H.; Xue, X. High-speed Privacy Amplification Scheme Using GMP in Quantum Key Distribution. *IEEE Photonics J.* **2020**, *12*, 1–13. [[CrossRef](#)]
29. Lodewyck, J.; Bloch, M.; Garcíapatron, R.; Fossier, S.; Karpov, E.; Diamanti, E.; Debuisschert, T.; Cerf, N.J.; Tuallebrouri, R.; Mclaughlin, S.W. Quantum key distribution over 25 km with an all-fiber continuous-variable system. *Phys. Rev. A* **2007**, *76*, 042305. [[CrossRef](#)]
30. Jouguet, P.; Kunz-Jacques, S.; Debuisschert, T.; Fossier, S.; Diamanti, E.; Alléaume, R.; Tualle-Brouri, R.; Grangier, P.; Leverrier, A.; Pache, P. Field test of classical symmetric encryption with continuous variables quantum key distribution. *Opt. Express* **2012**, *20*, 14030–14041. [[CrossRef](#)]
31. Huang, D.; Huang, P.; Li, H.; Wang, T.; Zhou, Y.; Zeng, G. Field demonstration of a continuous-variable quantum key distribution network. *Opt. Lett.* **2016**, *41*, 3511–3514. [[CrossRef](#)]
32. Zhang, Y.; Li, Z.; Chen, Z.; Weedbrook, C.; Zhao, Y.; Wang, X.; Huang, Y.; Xu, C.; Zhang, X.; Wang, Z.; et al. Continuous-variable QKD over 50 km commercial fiber. *Quantum Sci. Technol.* **2019**, *4*, 035006. [[CrossRef](#)]
33. Jouguet, P.; Elkouss, D.; Kunz-Jacques, S. High-bit-rate continuous-variable quantum key distribution. *Phys. Rev. A* **2014**, *90*, 042329. [[CrossRef](#)]
34. Guo, D.; He, C.; Guo, T.; Xue, Z.; Feng, Q.; Mu, J. Comprehensive high-speed reconciliation for continuous-variable quantum key distribution. *Quantum Inf. Process.* **2020**, *19*, 320. [[CrossRef](#)]
35. Jouguet, P.; Kunz-Jacques, S.; Leverrier, A. Long-distance continuous-variable quantum key distribution with a Gaussian modulation. *Phys. Rev. A* **2011**, *84*, 062317. [[CrossRef](#)]
36. Jiang, X.Q.; Huang, P.; Huang, D.; Lin, D.; Zeng, G. Secret information reconciliation based on punctured low-density parity-check codes for continuous-variable quantum key distribution. *Phys. Rev. A* **2017**, *95*, 022318. [[CrossRef](#)]
37. Wang, X.; Zhang, Y.C.; Li, Z.; Xu, B.; Yu, S.; Guo, H. Efficient rate-adaptive reconciliation for continuous-variable quantum key distribution. *Quantum Inf. Comput.* **2017**, *17*, 1123–1134.
38. Milicevic, M.; Feng, C.; Zhang, L.M.; Gulak, P.G. Quasi-cyclic multi-edge LDPC codes for long-distance quantum cryptography. *NPJ Quantum Inf.* **2018**, *4*, 21. [[CrossRef](#)]
39. Xiangyu, W.; Yichen, Z.; Song, Y.; Hong, G. High speed error correction for continuous-variable quantum key distribution with multi-edge type LDPC code. *Sci. Rep.* **2018**, *8*, 10543.
40. Li, Q.; Wen, X.; Mao, H.; Wen, X. An improved multidimensional reconciliation algorithm for continuous-variable quantum key distribution. *Quantum Inf. Process.* **2019**, *18*, 25. [[CrossRef](#)]
41. Zhang, K.; Jiang, X.Q.; Feng, Y.; Qiu, R.; Bai, E. High Efficiency Continuous-Variable Quantum Key Distribution Based on ATSC 3.0 LDPC Codes. *Entropy* **2020**, *22*, 1087. [[CrossRef](#)]
42. Wang, C.; Huang, D.; Huang, P.; Lin, D.; Peng, J.; Zeng, G. 25MHz clock continuous-variable quantum key distribution system over 50km fiber channel. *Sci. Rep.* **2015**, *5*, 102–108.

43. Poppe, A.; Peev, M.; Maurhart, O. Outline of the SECOQC Quantum-Key-Distribution Network in Vienna. *Int. J. Quantum Inf.* **2008**, *6*, 209–218. [[CrossRef](#)]
44. Alleaume, R.; Roueff, F.; Diamanti, E.; Lutkenhaus, N. Topological optimization of quantum key distribution networks. *New J. Phys.* **2009**, *11*, 075002. [[CrossRef](#)]
45. Fujiwara, M.; Ishizuka, H.; Miki, S.; Yamashita, T.; Wang, Z.; Tanaka, A.; Yoshino, K.; Nambu, Y.; Takahashi, S.; Tajima, A.; et al. Field demonstration of quantum key distribution in the Tokyo QKD Network. In Proceedings of the International Quantum Electronics Conference, Sydney, Australia, 28 August–1 September 2011; p. 1403.
46. Wang, Y.; Li, Q.; Mao, H.; Han, Q.; Huang, F.; Xu, H. Topological optimization of hybrid quantum key distribution networks. *Opt. Express* **2020**, *28*, 26348–26358. [[CrossRef](#)]
47. Fossier, S.; Diamanti, E.; Debuisschert, T.; Villing, A.; Tualle-Brouri, R.; Grangier, P. Field test of a continuous-variable quantum key distribution prototype. *New J. Phys.* **2009**, *11*, 045023. [[CrossRef](#)]
48. Arikan, E. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Trans. Inf. Theory* **2009**, *55*, 3051–3073. [[CrossRef](#)]
49. Korada, S.B.; Montanari, A.; Telatar, E.; Urbanke, R. An empirical scaling law for polar codes. In Proceedings of the 2010 IEEE International Symposium on Information Theory, Austin, TX, USA, 13–18 June 2010; pp. 884–888
50. Jouguet, P.; Kunz-Jacques, S. High performance error correction for quantum key distribution using polar codes. *Quantum Inf. Comput.* **2014**, *14*, 329–338.
51. Pirandola, S.; Laurenza, R.; Ottaviani, C.; Banchi, L. Fundamental limits of repeaterless quantum communications. *Nat. Commun.* **2017**, *8*, 15043. [[CrossRef](#)] [[PubMed](#)]