High-efficiency rate-adaptive reconciliation in continuous-variable quantum key distribution

Hongzhao Yang,[1,2] Shuaishuai Liu,[1,2] Shenshen Yang,[3] Zhenguo Lu,[1,2] Yanxia Li,[4] and Yongmin Li[1,2,5,*]

[1]*State Key Laboratory of Quantum Optics and Quantum Optics Devices, Institute of Opto-Electronics, Shanxi University, Taiyuan 030006, China*

[2]*Collaborative Innovation Center of Extreme Optics, Shanxi University, Taiyuan 030006, China*

[3]*College of Physics and Information Engineering, Shanxi Normal University, Taiyuan 030031, China*

[4]*Bei Jing Institute of Remote Sensing Equipment, Beijing 100143, China*

[5]*Hefei National Laboratory, Hefei 230088, China*

(Received 12 September 2023; accepted 6 December 2023; published 4 January 2024)

In continuous-variable (CV) quantum key distribution (QKD), the reconciliation efficiency and frame error rate of the information reconciliation have a significant impact on the secret key rate. For a fixed-rate multiedge-type low-density parity-check (LDPC) code, the reconciliation efficiency and frame error rate will inevitably vary due to the fluctuations of the signal-to-noise ratio (SNR) of the CV QKD system, which degrades the performance of the system. We propose a high-efficiency rate-adaptive information reconciliation scheme by combining raptorlike (RL) LDPC codes with the addition of trusted noises. We establish the model of adding trusted noises and combine it with the RL LDPC codes to optimize the secret key rate under the time-varying channel. The simulation results show that our scheme can maintain a high reconciliation efficiency of more than 94.4% and a low frame error rate within 15% fluctuation range of the SNR. Furthermore, we implement the hardware acceleration of the proposed rate-adaptive scheme on a graphics processing unit and achieve a decoding throughput of 65.5 Mbits/s by optimizing the storage of the parity-check matrix. Our results are useful for the practical CV QKD under the realistic time-varying channel.

DOI: 10.1103/PhysRevA.109.012604

I. INTRODUCTION

In the modern information society, information security is crucial and imposes a challenge for humans. Quantum key distribution (QKD) relies on the fundamental laws of quantum mechanics to enable two users to share information-theoretically secure keys [1–3]. According to the different information encoding carriers and measurement methods, QKD technology is divided into two types, namely, discrete-variable QKD and continuous-variable (CV) QKD. Discrete-variable QKD employs the single-photon detection and has the advantages of long transmission distance and relatively simple data postprocessing [4–10]. In contrast, CV QKD encodes information on the quadratures of quantized optical fields and can provide a high secret key rate over metropolitan areas. Furthermore, it has good compatibility with the current coherent optical communication technology and has seen rapid progress in recent years [11–25].

In a typical CV QKD protocol, Alice encodes the classical information onto the quadratures of the optical field and sends the encoded optical fields to Bob through a quantum channel. Bob randomly selects a measurement basis and measures the received states with homodyne detection. Then Bob sends his measurement basis to Alice through the classical channel and Alice keeps the data according to the measurement basis information sent by Bob. At this phase, the two users share a series of interrelated variables. Finally, the final secret key is extracted through the data postprocessing. The postprocessing of CV QKD is mainly divided into four steps, namely, (i) data sifting, (ii) parameter estimation, (iii) information reconciliation, and (iv) privacy amplification. Among them, the most complicated step is the information reconciliation [26]. In the information reconciliation process, Alice and Bob use the classic error-correcting code to correct the correlated raw keys and obtain a set of completely consistent bit string. It has high computational complexity and is one of the key bottlenecks restricting the performance of the QKD system. There are two main types of information reconciliation schemes in CV QKD: slice reconciliation [27,28] and multidimensional reconciliation [29]. The first is suitable for short distances, i.e., high signal-to-noise ratio (SNR), while the latter is suitable for longer distances (low SNR). The multiedge-type (MET) low-density parity-check (LDPC) code is an extension of conventional LDPC code with the performance close to the Shannon limit [30,31]. Its combination with the multidimensional reconciliation greatly improves the efficiency of the information reconciliation and dramatically increases the transmission distance of the CV QKD [32,33].

In real application scenarios, due to the variations of the surrounding environment and the imperfections of the devices, the SNR of the QKD system will inevitably fluctuate. If a fixed-rate MET LDPC code is employed, even a slight variation of the SNR will cause significant changes in the reconciliation efficiency and frame error rate, which will in turn severely degrade the secret key rate of the system [34]. Therefore, it requires that the rate of the MET LDPC code should be adjusted accordingly with the SNR variations

*yongmin@sxu.edu.cn

during the process of information reconciliation to ensure the security of the system and maintain a high secret key rate. When extensive changes are occurred for the SNR, the reconciliation can be switched between the slice reconciliation and the multidimensional reconciliation. For continuous and small fluctuations of SNRs, it is necessary to adjust the rate of the error-correction code by rate-adaptive technology. Several rate-adaptive reconciliation techniques in CV QKD have been reported, such as puncturing and shortening [34], rateless reconciliation protocol [35], adding a controlled amount of digital noise [36], raptorlike (RL) LDPC code [37], and polar-coding-based rate-adaptive reconciliation [38].

In this paper, we propose a high-efficiency rate-adaptive scheme for CV QKD by combining RL LDPC codes with the addition of trusted noises. First, we construct a RL LDPC code [37,39] with a rate of 0.1, which is further extended to a series of codes with different rates by adjusting its original rate. For each code, we add appropriate trusted noises to Bob's raw data in terms of the varying SNR to stabilize the SNR. In this way, a certain range of SNR can be covered by our information reconciliation method, which ensures a stable and optimized secret key rate. The simulation results show that our scheme performs well: It could maintain the reconciliation efficiency over 94.4% and frame error rate less than 9.3% when the fluctuation range of the SNR is within 15%. By using GPU hardware acceleration, the decoding speed of our rate-adaptive reconciliation reaches above 65.5 Mbits/s.

This paper is organized as follows. In Sec. II we present the concrete construction process of the RL LDPC code and the rate-adaptive method based on it. In Sec. III we analyze the model of adding trusted noises to Bob and then present the mechanism that combines it with RL LDPC codes. In Sec. IV we present the simulation results of our scheme. Furthermore, the GPU hardware acceleration of the decoding algorithm is investigated. We summarize in Sec. V.

II. RL LDPC CODE IN INFORMATION RECONCILIATION OF CV QKD

The RL LDPC code was proposed for the cloud transmission system [39] and recently applied to the CV QKD system [37]. The RL LDPC codes have a similar performance to LDPC codes, provide a coding gain close to the Shannon limit, and have the rateless characteristics of raptor codes.

The RL LDPC code can also be regarded as a special MET LDPC code. The degree distribution of nodes is the basic parameter of MET LDPC codes. We can obtain the degree distribution of MET LDPC codes by density evolution [40–42], generalized extrinsic information transfer charts [43], or other methods. The MET LDPC code defines the degree distribution from the perspective of variable nodes and check nodes. We use the vector $\vec{b} := (b_0, b_1, \ldots, b_{n\tau})$ to represent the parameters of the receiving signal channel, the vector $\vec{d} := (d_1, d_2, \ldots, d_{ne})$ to represent the degree of the edge type, and the vector $\vec{r} := (r_0, r_1, \ldots, r_{n\tau})$ to represent the number of receiving channels. Here ne denotes the number of edge types and $n\tau$ denotes the number of channels for transmitting information. By using these parameters, the degree distributions of MET LDPC codes are defined by

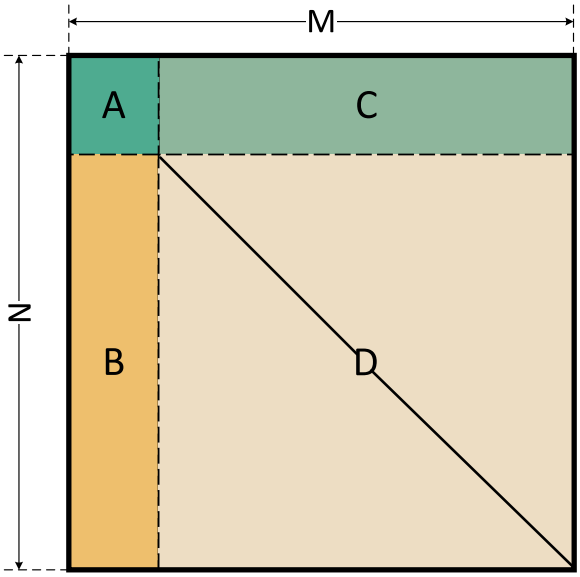

FIG. 1. Structure of the parity-check matrix of the RL LDPC code, where the size of the matrix is $M \times N$. It is formed by cascading four submatrices A, B, C, and D. Among them, C is an all-zero matrix, D is an identity matrix, and A and B are constructed according to the specific degree distribution.

$$v(r, x) = \sum v_i r^b x^d, \tag{1}$$

$$\mu(x) = \sum \mu_i x^d, \tag{2}$$

where $v(r, x)$ denotes the degree distribution of variable nodes, $\mu(x)$ denotes the degree distribution of check nodes, v_i denotes the proportion of variable nodes of type i to the total number of variable nodes, and μ_i denotes the proportion of check nodes of type i to the total number of variable nodes.

The schematic diagram of the parity-check matrix of the RL LDPC code is shown in Fig. 1. Taking the RL LDPC code with three types of edges as an example, its parity-check matrix consists of four submatrices A, B, C and D, where C is an all-zero matrix. The three submatrices A, B, and D correspond to parity-check matrices with different types of edges in the MET LDPC code and D is an identity matrix corresponding to the edge connected to a node with degree 1.

From the degree distribution, we can get the proportions of variable nodes and check nodes connected by different types of edges to the total number of variable nodes as well as the number of different types of edges so that we can obtain the relevant parameters of different types of edges. By using these parameters, each submatrix can be constructed and then connected as shown in Fig. 1 to form the complete RL LDPC code.

Note that MET LDPC codes with a fixed rate can only achieve the best performance at a fixed SNR. When the SNR changes, the code rate needs to be changed accordingly to maintain their superior performance. The adjustment method for the code rate is shown in Fig. 2.

The RL LDPC code rate can be increased by cutting the initial parity-check matrix. After cutting, the RL LDPC code

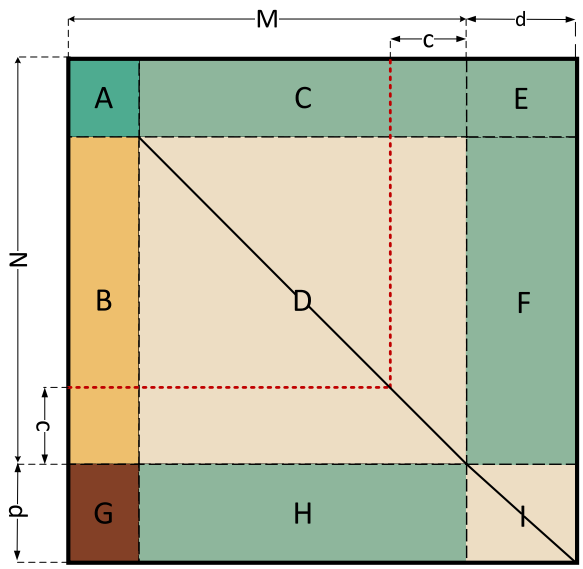

FIG. 2. Schematic diagram of the RL LDPC code rate adjustment. The RL LDPC code increases its rate by cutting the lower right part of the parity-check matrix with a cutting length of c (marked by the red dotted line). The code rate can be reduced by expanding the parity-check matrix with the submatrices E, F, G, and H and the extended length is d.

reduces the same number of variable nodes and check nodes and its code rate increases to

$$R = \frac{(n-c)-(m-c)}{n-c} = \frac{n-m}{n-c}, \tag{3}$$

where n is the number of variable nodes, m is the number of check nodes, and c is the cutting length of the matrix. The cutting of the matrix will inevitably change the submatrices of B, C, and D. Because D is an identity matrix and C is an all-zero matrix, they are immune to the cutting. However, this is not the case for matrix B. To ensure that the cut parity-check matrix can maintain good performance, we need to redesign the degree distribution when constructing the submatrix B. The design of the degree distribution can refer to the reference matrix

$$\lim_{R\to R_t} \mu(x) = \mu_t(x), \tag{4}$$

where R is the code rate of the reconstructed code, R_t is the code rate of the reference code, $\mu(x)$ denotes the degree distribution of the reconstructed code, and $\mu_t(x)$ denotes the degree distribution of the reference code. By searching the degree distribution of the reconstructed submatrix B in this way, the performance of the original matrix can be well maintained.

The rate of the RL LDPC code can be reduced by expanding the parity-check matrix as shown in Fig. 2. The reduced code rate is given by

$$R = \frac{(n+d)-(m+d)}{n+d} = \frac{n-m}{n+d}, \tag{5}$$

where d is the dimension of the extended matrix. After the extension, the submatrix E is an all-zero matrix, and the submatrix D and the extended submatrices F, H, and I form a new identity matrix together. Similar to the method of increasing the code rate, the degree distribution for constructing the extended submatrix G needs to change accordingly with the length of the extension. The specific design procedure can also refer to Eq. (4), where $\mu(x)$ denotes the degree distribution of the combined matrix of submatrix B and the extended matrix G.

By using the above approaches of cutting and expanding, we can reduce or increase the rate of the RL LDPC code according to the change of the SNR.

III. RATE-ADAPTIVE METHOD WITH RL LDPC CODE AND ADDING TRUSTED NOISE

In continuous-variable quantum key distribution, the asymptotic secret key rate against collective attacks per signal pulse is written as [44]

$$K = (1-R_{FE})\left(\frac{n}{N}\right)(\beta I_{AB} - \chi_{BE}), \tag{6}$$

where R_{FE} is the frame error rate (FER), n is the number of data used to extract the key, N is the number of sifted data after quantum transmission and measurement, β is the reconciliation efficiency, I_{AB} is the mutual information of the data between Alice and Bob, and χ_{BE} is the upper bound on the information that the eavesdropper Eve may steal. It can be seen from Eq. (6) that the reconciliation efficiency and frame error rate are crucial to the secret key rate. The reconciliation efficiency of the multidimensional reconciliation is defined as

$$\beta = \frac{R}{I_{AB}}, \tag{7}$$

where R denotes the code rate of the error-correction code. For Gaussian modulated coherent state protocols and additive white Gaussian noise channel, I_{AB} equals the channel capacity

$$I_{AB} = C = \tfrac{1}{2}\log_2(1+\mathrm{SNR}). \tag{8}$$

The size of the mutual information affects the frame error rate of the error correction. Therefore, matching the code rate of the error-correction code with the SNR is critical to achieve the optimal secret key rate. However, it is infeasible to construct all required RL LDPC codes with different rates due to continuous variations of the SNR in real scenarios. In general, one constructs a series of discrete single-rate RL LDPC codes and each of them can cover a certain range of SNR. However, this will inevitably lead to fluctuations in reconciliation efficiency and frame error rate, which usually degrade the secret key rate. To overcome this drawback, we introduce the method of adding trusted noise to Bob's data to adjust the system's SNR and combine it with the RL LDPC codes to achieve high-efficiency rate-adaptive reconciliation.

In order to adjust the SNR to the target value, we need to find the variance of the added trusted noise. For a Gaussian modulated coherent state with homodyne detection protocol, the SNR of the system is given by

$$\mathrm{SNR} = \frac{V_B}{N_B} - 1, \tag{9}$$

$$V_B = T\eta V_A + T\eta\varepsilon + V_{el} + 1, \tag{10}$$

$$N_B = T\eta\varepsilon + V_{\mathrm{el}} + 1, \tag{11}$$

where V_B is the variance of Bob's data, N_B is the variance of the total noises of the system, V_A is the modulation variance, ε is the excess noise, V_{el} is the electrical noise (V_B, V_A, V_{el}, and ε are all in shot-noise units), T is the transmittance of the quantum channel, and η is the detection efficiency. When the channel transmittance T of the quantum channel changes to T' and the modulation variance V_A changes to V_A', V_B and N_B will change accordingly

$$V_B' = T'\eta V_A' + T'\eta\varepsilon + V_{\mathrm{el}} + 1, \tag{12}$$

$$N_B' = T'\eta\varepsilon + V_{\mathrm{el}} + 1. \tag{13}$$

From Eq. (9), the SNR of the system also changes from SNR to SNR$'$ due to the variation of V_B and N_B. To recover the initial SNR, we add trusted Gaussian noise with variance V_n to Bob's data

$$V_B'' = V_B' + V_n, \tag{14}$$

$$N_B'' = N_B' + V_n, \tag{15}$$

$$\mathrm{SNR} = \frac{V_B''}{N_B''} - 1. \tag{16}$$

By combining Eqs. (9)–(11) and (14)–(16), we can obtain the variance of the trusted noise that needs to be added

$$V_n = \frac{T'\eta V_A'}{\mathrm{SNR}} - T'\eta\varepsilon - V_{\mathrm{el}} - 1. \tag{17}$$

Equation (17) can also be expressed as

$$V_n = \frac{\rho^2 V_B'}{\mathrm{SNR}} - V_B'(1-\rho^2), \tag{18}$$

$$\rho^2 = \frac{\langle x_A x_B\rangle^2}{V_A V_B}, \tag{19}$$

where x_A and x_B are Alice and Bob's data. Equation (18) can be used to directly calculate the variance of the trusted noise based on the experimental data.

When the quantum channel transmittance and modulation variance change, the system's SNR remains unchanged due to the addition of the trusted noises, so the mutual information I_{AB} between Alice and Bob remains unchanged. However, the eavesdropped information χ_{BE} will vary and can be estimated by using the substitutions

$$V_A \to V_A', \tag{20}$$

$$T \to T', \tag{21}$$

$$V_{\mathrm{el}} \to V_{\mathrm{el}}' = V_{\mathrm{el}} + V_n. \tag{22}$$

At this stage, the secret key rate can be estimated by Eq. (6) using the modified χ_{BE}.

To design the MET LDPC code, we use the progressive edge growth [45] algorithm to construct the submatrix and then cascade them together. We design a number of RL LDPC codes (with different rates) at certain SNRs by the method in Sec. II, and each RL LDPC code is responsible for reconciling data within a certain SNR range. The values of these initial SNRs are determined by searching for the RL LDPC codes at different SNRs to maximize the secret key rate. Within the reconciliation SNR range of each RL LDPC code, appropriate trusted noises are added to ensure the SNR is always kept at the level of the initial SNR.

The flow chart of the overall postprocessing is shown in Fig. 3. To break the 3-dB limit [46], reverse reconciliation is considered hereafter. After the data sifting, the two parties share correlated Gaussian data (raw keys). Then Alice declares publicly a random sample of her data and Bob uses it to estimate the relevant parameters of the QKD system, including quantum channel transmittance, excess noise, SNR, etc. Based on the estimated parameters, Bob chooses the RL LDPC code with the rate closest to the ideal code rate, calculates the variance of the trusted noise, and then adds the trusted noise to his raw data. It is assumed that the required set of RL LDPC codes with different code rates has been designed in advance according to the fluctuation range of the SNR. Next Bob uses the multidimensional reconciliation method to encode his raw data and sends the side information to Alice through the classical channel. Alice chooses the RL LDPC code with the same rate as that of Bob according to the data length of the received side information and decodes her data. If the decoding is successful, both parties perform private amplification and finally identical secret keys can be shared.

IV. SIMULATION RESULTS

In this section we investigate the performance of our rate-adaptive reconciliation scheme in detail. First, we design the base matrix of RL LDPC codes with a rate of 0.1 [43] and a code length of 20 000 and expand the code length by 50 times to 10^6 by using the quasicyclic construction technique [47]. Note that the quasicyclic construction technology can reduce the complexity of constructing LDPC codes without reducing the error-correction performance. Then four RL LDPC codes with base matrix lengths of 18 900, 19 440, 20 600, and 21 240 are constructed. The code lengths of the four base matrices are determined using Eqs. (3), (5), and (7), where the reconciliation efficiency remains constant. The five RL LDPC codes cover a SNR range from 0.148 to 0.171, i.e., each code covers a SNR range of 0.005. The reasons for the SNR range selection are as follows. For a metropolitan area of 50 km, the typical SNR of a CV QKD system will be around 0.16. Considering the fluctuation range of $\pm 7\%$ of the transmittance for the quantum channel, the corresponding variation range of the SNR is determined.

Figure 4 shows the FER and reconciliation efficiency of the five RL LDPC codes versus the SNR. The simulation parameters are $V_A = 2.941$, $\varepsilon = 0.05$, $\eta = 0.6$, and $V_{\mathrm{el}} = 0.1$. The maximum number of decoding iterations is set to 100 times. Here we have assumed that the variations of the SNR are mainly caused by the fluctuations of the channel transmittance and the modulation variance V_A remains constant. The reason is that the source is controlled by Alice and a servo system is usually employed to stabilize its output, whereas the quantum channel is exposed to an uncontrollable external environment and suffers from disturbance. The SNR changes from 0.148 to 0.171, which corresponds to a transmittance variation from 0.0925 to 0.1069. We use a multidimensional reconciliation method and a layered decoding algorithm. Compared with the flooding log-likelihood ratio (LLR) belief propagation

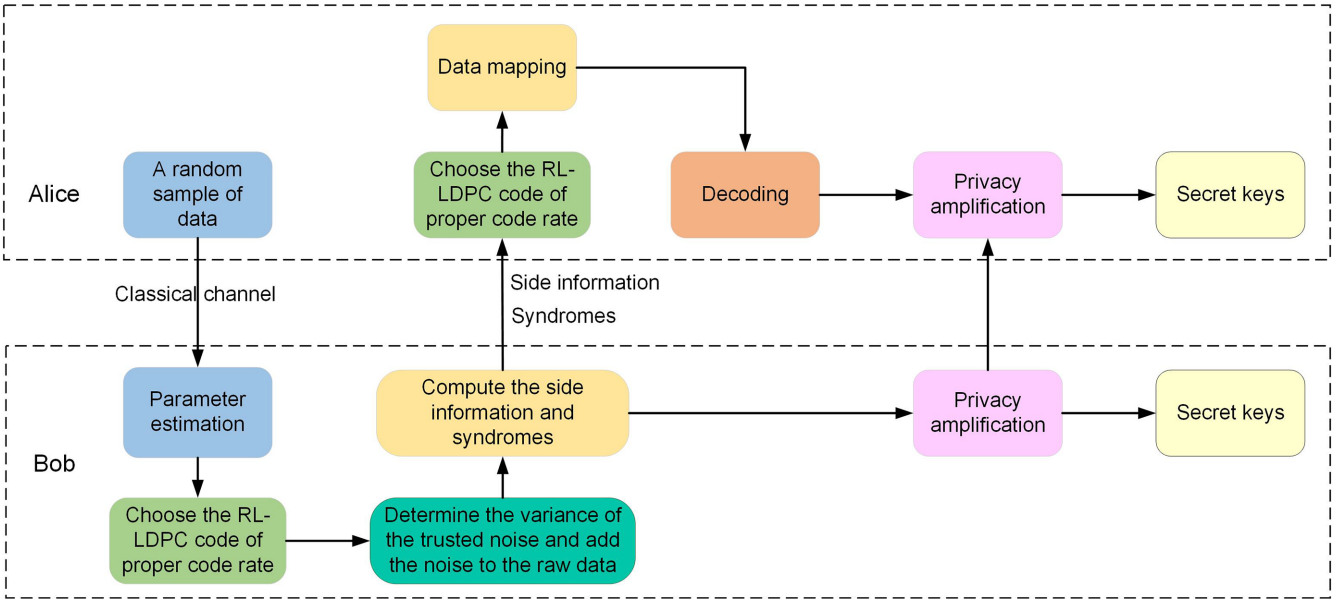

FIG. 3. Flow chart of rate-adaptive reverse reconciliation by combing the RL LDPC code with adding the trusted noise

decoding algorithm, the number of iterations of the layered decoding algorithm is reduced by nearly half without performance degradation and it has lower computational complexity [48]. For each RL LDPC code, the reconciliation efficiency increases when the SNR decreases, whereas the corresponding FER increases due to the difficulty of decoding at lower SNR and vice versa.

By using the results in Fig. 4, we calculate the secret key rate at each SNR and select the SNR points that have the highest secret key rate within the SNR range of each RL LDPC code, as shown in Fig. 5(a). The reconciliation efficiency reaches above 94.4% for excess noises of 0.01, 0.03, and 0.05 and 95.0% for excess noise of 0.08. The selected SNRs act as the benchmark SNR for each RL LDPC code (the minimum SNR). The SNR covering range for a single RL LDPC code is set to be 0.005, and the SNRs higher than the benchmark within this range can be adjusted to the benchmark value by adding trusted noise. From Fig. 5(a), the optimal reconciliation efficiency for the excess noise of 0.08 is higher than that for excess noises of 0.01, 0.03, and 0.05, although the corresponding FER is higher. This is because the secret key rate of the CV QKD is more sensitive to the reconciliation efficiency than the FER at high excess noise levels compared to the case of low excess noise.

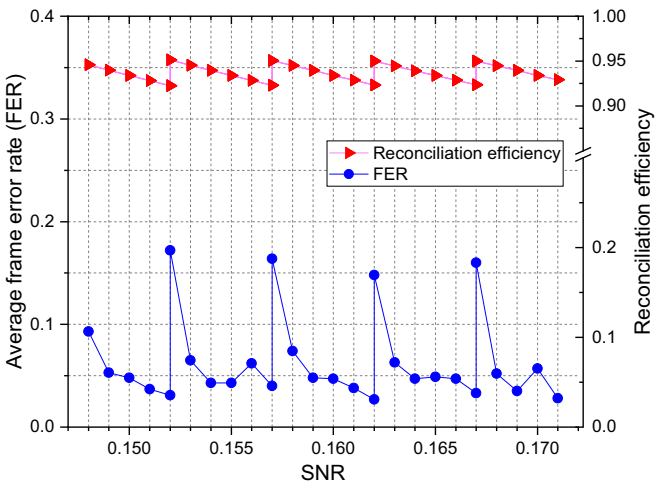

FIG. 4. Performance test of RL LDPC codes with code lengths of 945 000, 972 000, 1 000 000, 1 030 000, and 1 062 000, respectively. The blue circle and red triangle represent the frame error rate and reconciliation efficiency, respectively. The maximum number of decoding iterations is set to 100.

Figure 5(b) shows the secret key rate of the rate-adaptive reconciliation versus the SNR. For comparison, the key rate obtained by the method using only the RL LDPC codes [37] is also depicted. It can be seen that the secret key rate obtained by our scheme is higher, especially when the excess noise is high. By adding the trusted noise, the frame error rate and reconciliation efficiency remain unchanged within the reconciliation range of each RL LDPC code. Although the mutual information between Alice and Bob is reduced due to the addition of noises, from Eq. (7) its product with the reconciliation efficiency, which is exactly the code rate, remains unchanged. At the same time, the addition of trusted noises suppresses the eavesdropped information. Therefore, the overall effect of adding noises improves the secret key rate of the QKD system. For example, by adding trusted noise with a variance of 0.028 15 at the SNR of 0.161 and the excess noise level of 0.08, we can obtain a secret key rate more than 2 times higher than that using only the RL LDPC code. The upper bound on the information that the eavesdropper Eve may steal is 0.096 18 and 0.098 48 bits per pulse for adding trusted noises or not, respectively.

Hardware acceleration can greatly improve the decoding throughput [47,49–52]. By using GPUs, decoding throughputs of 9.17 [47], 30.39 [49], and 64.11 Mbits/s [50] with the maximum number of iterations of 100, 100, and 50 times, respectively, have been reported. We implement the hardware acceleration of our rate-adaptive reconciliation algorithm on

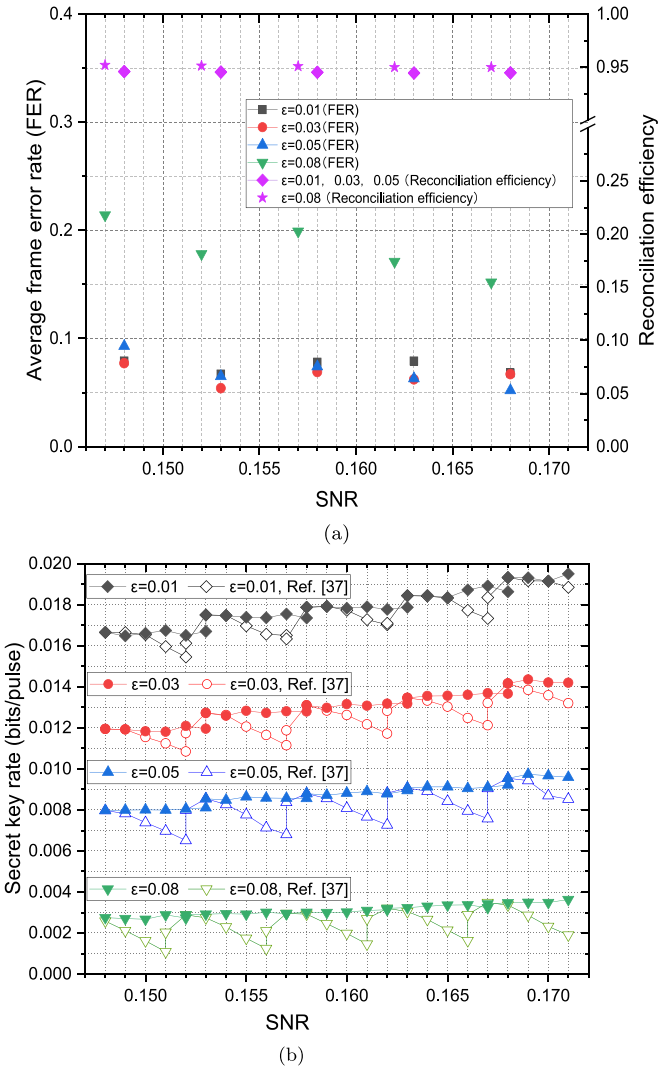

FIG. 5. (a) Reconciliation efficiency and FER of the five RL LDPC codes versus the SNR under different excess noises. (b) Secret key rate using our rate-adaptive reconciliation scheme or RL LDPC codes only. The simulation parameters are set as follows: The modulation variances V_A are 2.935, 2.938, 2.941, and 2.946, which correspond to the excess noise levels of 0.01, 0.03, 0.05, and 0.08, respectively; the detection efficiency η is 0.6; and the electric noise is 0.1.

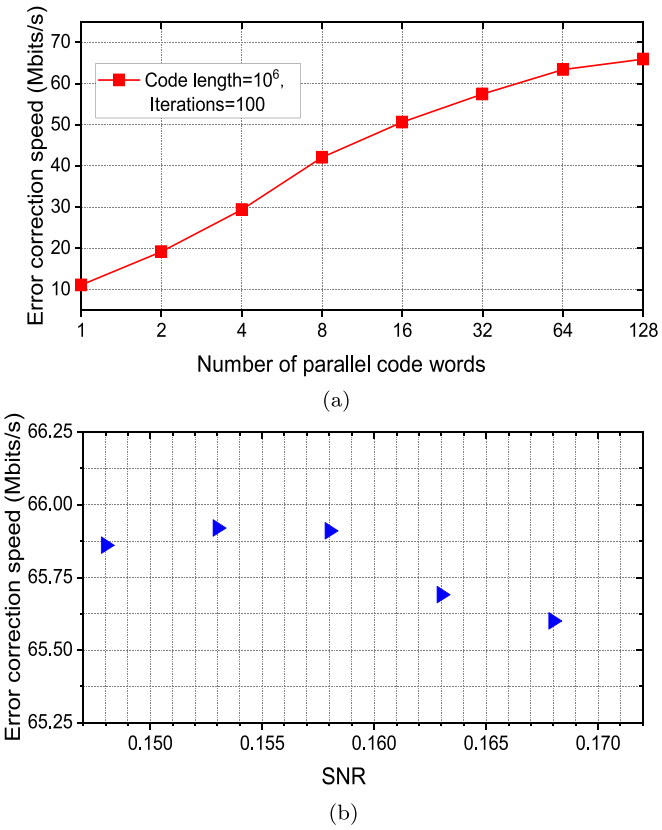

FIG. 6. (a) Error-correction speed versus the different number of parallel code words. The code rate is 0.1, the code length is 10^6, and the number of iterations is 100. (b) Error-correction speed of RL LDPC codes with five different code lengths (1 060 000, 1 030 000, 1 000 000, 970 000, and 945 000). The number of parallel code words is 128 and the number of iterations is 100.

the GPU (NVIDIA GeForce RTX 3090). In order to maintain a low frame error rate and high reconciliation efficiency, we set the maximum number of iterations to 100 without early termination. First, we construct a base matrix of the RL LDPC code with a length of 400 and extend its code length to 424. Then we use the quasicyclic construction technique to expand the code length by 2500 times to 1 060 000. Next we cut the code length of the base matrix (424) to 412, 400, 388, and 378 without changing the degree distribution. We expand the four base matrices by the same expansion factor (2500), and each submatrix of the quasicyclic expansion has the same shift amount. In this way, the whole information of the four RL LDPC codes is involved in the parity-check matrix of the RL LDPC code with the code length (base matrix) of 424, which saves the memory space for the matrix information on the GPU and reduces the complexity of programming.

To implement the iterative decoding, a layered decoding algorithm is adopted and the unrelated submatrices are merged [50] to improve the thread utilization and the decoding speed. Figure 6(a) shows the decoding throughput as a function of the number of parallel code words under the code length of 10^6.

The decoding throughput on the GPU is given by

$$K = \frac{a \times b}{T}, \tag{23}$$

where a is the code length of the RL LDPC code, b is the number of parallel code words, and T is the total time consumed by the decoding process. Here T includes the time consumed by GPU initialization, data transfer from CPU to GPU, LLR initialization, iterative decoding, decoding decision, and data transfer from GPU to CPU. It can be seen from Fig. 6(a) that the error-correction speed increases linearly with the number of parallel code words at first and gradually tends to saturation. In the following, we choose the number of parallel code words to be 128.

Figure 6(b) shows the error-correction speed of five RL LDPC codes versus the SNR by using the GPU platform. The maximum number of iterations is set to 100. In the entire

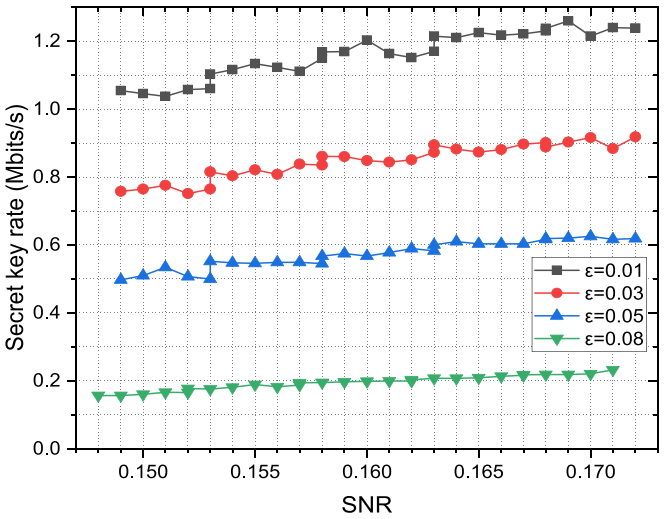

FIG. 7. Actual secret key rate obtained by considering the GPU decoding throughput under different excess noises. The other parameters are the same as those in Fig. 5.

adaptive range of the SNR, the decoding throughput is robust to the SNR of the QKD system and can reach above 65.5 Mbits/s.

When considering the actual decoding throughput, the obtainable secret key rate per second can be calculated by the formula

$$K_1 = a(1 - R_{FE})\left(\frac{n}{N}\right)(\beta I_{AB} - \chi_{BE}), \tag{24}$$

where a is the real-time decoding throughput of the information reconciliation and we have assumed that the clock rate of the system is no less than the decoding throughput.

Figure 7 shows the actual secret key rate as a function of the SNR obtained by considering the GPU decoding throughput. For the fixed excess noise, the key rate increases with the SNR, similar to the phenomenon of Fig. 5(b). When the excess noises are 0.01, 0.03, 0.05, and 0.08, the secret key rate per second can reach above 1.04 Mbits/s, 750 kbits/s, 500 kbits/s, and 150 kbits/s, respectively.

V. CONCLUSION

The rapid development of CV QKD technology puts forward higher requirements for the information reconciliation. Adaptive information reconciliation plays a key role in practical CV QKD systems: It ensures that the QKD systems can obtain a high and stable secret key rate in realistic time-varying channel environments. In this paper, we proposed a high-efficiency rate-adaptive scheme by combining RL LDPC codes and adding trusted noise. We established the model of adding trusted noise and systematically analyzed the reconciliation efficiency and frame error rate of the rate-adaptive scheme and their effect on the secret key rate under different excess noise levels. The simulation results showed that our scheme has superior performance: The reconciliation efficiency remains above 94.4% (95%) and the frame error rate keeps lower than 9.3% (21.4%) for the excess noise levels of 0.01, 0.03, and 0.05 (0.08) within a SNR variation range of 15%. Furthermore, we implemented hardware acceleration of the decoding algorithm on a GPU platform. By optimizing the storage of RL LDPC codes and merging unrelated submatrices, we improved the thread utilization and reduced the complexity of programming. A decoding throughput of over 65.5 Mbits/s was achieved in the entire SNR range of adaptive reconciliation. The proposed rate-adaptive information reconciliation scheme and hardware acceleration implementation can be applied to practical CV QKD systems in real application scenarios and may find useful applications in other quantum communication fields.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China (Grants No. 62175138 and No. 62205188), Shanxi 1331KSC, Fundamental Research Program of Shanxi Province (Grant No. 202203021222232), Innovation Program for Quantum Science and Technology (Grant No. 2021ZD0300703), and Scientific and Technological Innovation Programs of Higher Education Institutions in Shanxi Province (Grant No. 2021L258).

[1] F. Xu, X. Ma, Q. Zhang, H.-K. Lo, and J.-W. Pan, Rev. Mod. Phys. **92**, 025002 (2020).
[2] S. Pirandola, U. L. Andersen, L. Banchi, M. Berta, D. Bunandar, R. Colbeck, D. Englund, T. Gehring, C. Lupo, C. Ottaviani, J. L. Pereira, M. Razavi, J. S. Shaari, M. Tomamichel, V. C. Usenko, G. Vallone, P. Villoresi, and P. Wallden, Adv. Opt. Photon. **12**, 1012 (2020).
[3] C. Portmann and R. Renner, Rev. Mod. Phys. **94**, 025008 (2022).
[4] M. Lucamarini, Z. L. Yuan, J. F. Dynes, and A. J. Shields, Nature (London) **557**, 400 (2018).
[5] X. Zhong, J. Hu, M. Curty, L. Qian, and H.-K. Lo, Phys. Rev. Lett. **123**, 100506 (2019).
[6] Y.-A. Chen, Q. Zhang, T.-Y. Chen, W.-Q. Cai, S.-K. Liao, J. Zhang, K. Chen, J. Yin, J.-G. Ren, Z. Chen, S.-L. Han, Q. Yu, K. Liang, F. Zhou, X. Yuan, M.-S. Zhao, T.-Y. Wang, X. Jiang, L. Zhang, W.-Y. Liu *et al.*, Nature (London) **589**, 214 (2021).
[7] S. Wang, Z.-Q. Yin, D.-Y. He, W. Chen, R.-Q. Wang, P. Ye, Y. Zhou, G.-J. Fan-Yuan, F.-X. Wang, W. Chen, Y.-G. Zhu, P. V. Morozov, A. V. Divochiy, Z. Zhou, G.-C. Guo, and Z.-F. Han, Nat. Photon. **16**, 154 (2022).
[8] Y. Liu, W.-J. Zhang, C. Jiang, J.-P. Chen, C. Zhang, W.-X. Pan, D. Ma, H. Dong, J.-M. Xiong, C.-J. Zhang, H. Li, R.-C. Wang, J. Wu, T.-Y. Chen, L. You, X.-B. Wang, Q. Zhang, and J.-W. Pan, Phys. Rev. Lett. **130**, 210801 (2023).
[9] L. Zhou, J. Lin, Y. Jing, and Z. Yuan, Nat. Commun. **14**, 928 (2023).
[10] W. Li, L. Zhang, Y. Lu, Z.-P. Li, C. Jiang, Y. Liu, J. Huang, H. Li, Z. Wang, X.-B. Wang, Q. Zhang, L. You, F. Xu, and J.-W. Pan, Phys. Rev. Lett. **130**, 250802 (2023).
[11] T. C. Ralph, Phys. Rev. A **61**, 010303(R) (1999).
[12] P. Jouguet, S. K. Jacques, A. Leverrier, P. Grangier, and E. Diamanti, Nat. Photon. **7**, 378 (2013).
[13] E. Diamanti and A. Leverrier, Entropy **17**, 6072 (2015).

[14] F. Laudenbach, C. Pacher, C.-H. F. Fung, A. Poppe, M. Peev, B. Schrenk, M. Hentschel, P. Walther, and H. Hübel, Adv. Quantum Technol. **1**, 1800011 (2018).
[15] Y. Zhang, Z. Chen, S. Pirandola, X. Wang, C. Zhou, B. Chu, Y. Zhao, B. Xu, S. Yu, and H. Guo, Phys. Rev. Lett. **125**, 010502 (2020).
[16] B. Qi, H. Gunther, P. G. Evans, B. P. Williams, R. M. Camacho, and N. A. Peters, Phys. Rev. Appl. **13**, 054065 (2020).
[17] Y. Tian, P. Wang, J. Liu, S. Du, W. Liu, Z. Lu, X. Wang, and Y. Li, Optica **9**, 492 (2022).
[18] H. Wang, Y. Li, Y. Pi, Y. Pan, Y. Shao, L. Ma, Y. Zhang, J. Yang, T. Zhang, W. Huang, and B. Xu, Commun. Phys. **5**, 162 (2022).
[19] N. Jain, H.-M. Chin, H. Mani, C. Lupo, D. S. Nikolic, A. Kordts, S. Pirandola, T. B. Pedersen, M. Kolb, B. Ömer, C. Pacher, T. Gehring, and U. L. Andersen, Nat. Commun. **13**, 4740 (2022).
[20] Y. Tian, Y. Zhang, S. Liu, P. Wang, Z. Lu, X. Wang, and Y. Li, Opt. Lett. **48**, 2953 (2023).
[21] M. Zhang, P. Huang, P. Wang, S. Wei, and G. Zeng, Opt. Lett. **48**, 1184 (2023).
[22] S. Du, P. Wang, J. Liu, Y. Tian, and Y. Li, Photon. Res. **11**, 463 (2023).
[23] Y. Xu, T. Wang, H. Zhao, P. Huang, and G. Zeng, Photon. Res. **11**, 1449 (2023).
[24] Z. Chen, X. Wang, S. Yu, Z. Li, and H. Guo, npj Quantum Inf. **9**, 28 (2023).
[25] S. S. Liu, Z. G. Lu, P. Wang, Y. Tian, X. Y. Wang, and Y. M. Li, npj Quantum Inf. **9**, 92 (2023).
[26] S. S. Yang, Z. L. Yan, H. Z. Yang, Q. Lu, Z. G. Lu, L. Y. Cheng, X. Y. Miao, and Y. M. Li, EPJ Quantum Technol. **10**, 40 (2023).
[27] G. Van Assche, J. Cardinal, and N. J. Cerf, IEEE Trans. Inf. Theory **50**, 394 (2004).
[28] Z. Bai, S. Yang, and Y. Li, Jpn. J. Appl. Phys. **56**, 044401 (2017).
[29] A. Leverrier, R. Alléaume, J. Boutros, G. Zémor, and P. Grangier, Phys. Rev. A **77**, 042325 (2008).
[30] R. G. Gallager, IEEE Trans. Inf. Theory **8**, 21 (1962).
[31] T. Richardson and R. Urbanke, *Modern Coding Theory* (Cambridge University Press, New York, 2008).
[32] Y.-F. Feng, Y.-J. Wang, R. Qiu, K. Zhang, H. Ge, Z. Shan, and X.-Q. Jiang, Phys. Rev. A **103**, 032603 (2021).
[33] S. Jeong, H. Jung, and J. Ha, npj Quantum Inf. **8**, 6 (2022).
[34] X. Wang, Y. Zhang, S. Yu, B. Xu, Z. Li, and H. Guo, Quantum Inf. Comput. **17**, 1123 (2017).
[35] C. Zhou, X. Wang, Y. Zhang, Z. Zhang, S. Yu, and H. Guo, Phys. Rev. Appl. **12**, 054013 (2019).
[36] S. Kreinberg, I. Koltchanov, and A. Richter, *Conference on Lasers and Electro-Optics, Washington, DC, 2020*, OSA Technical Digest (Optica, Washington, DC, 2020), paper ATh1I.6.
[37] C. Zhou, X. Wang, Z. Zhang, S. Yu, Z. Chen, and H. Guo, Sci. China Phys. Mech. Astron. **64**, 260311 (2021).
[38] Z. Cao, X. Chen, G. Chai, K. Liang, and Y. Yuan, Phys. Rev. Appl. **19**, 044023 (2023).
[39] S. I. Park, Y. Wu, H. M. Kim, N. Hur, and J. Kim, IEEE Trans. Broadcast. **60**, 239 (2014).
[40] S.-Y. Chung, T. J. Richardson, and R. L. Urbanke, IEEE Trans. Inf. Theory **47**, 657 (2001).
[41] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, IEEE Trans. Inf. Theory **47**, 585 (2001).
[42] S.-Y. Chung, G. D. Forney, T. S. Richardson, and R. Urbanke, IEEE Commun Lett. **5**, 58 (2001).
[43] H. Mani, T. Gehring, P. Grabenweger, B. Ömer, C. Pacher, and U. L. Andersen, Phys. Rev. A **103**, 062419 (2021).
[44] A. Leverrier, F. Grosshans, and P. Grangier, Phys. Rev. A **81**, 062343 (2010).
[45] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, IEEE Trans. Inf. Theory **51**, 386 (2005).
[46] C. Silberhorn, T. C. Ralph, N. Lütkenhaus, and G. Leuchs, Phys. Rev. Lett. **89**, 167901 (2002).
[47] M. Milicevic, C. Feng, L. M. Zhang, and P. G. Gulak, npj Quantum Inf. **4**, 21 (2018).
[48] D. E. Hocevar, *Proceedings of the IEEE Workshop on Signal Processing Systems: Design and Implementation, Austin, 2004* (IEEE, Piscataway, 2004), pp. 107–112.
[49] X. Wang, Y. Zhang, S. Yu, and H. Guo, Sci. Rep. **8**, 10543 (2018).
[50] Y. Li, X. Zhang, Y. Li, B. Xu, L. Ma, J. Yang, and W. Huang, Sci. Rep. **10**, 14561 (2020).
[51] S. Yang, Z. Lu, and Y. Li, J. Lightw. Technol. **38**, 3935 (2020).
[52] S.-S. Yang, J.-Q. Liu, Z.-G. Lu, Z.-L. Bai, X.-Y. Wang, and Y.-M. Li, IEEE Access **9**, 47687 (2021).