entropy

MDPI

Article

Rate-Compatible LDPC Codes for Continuous-Variable Quantum Key Distribution in Wide Range of SNRs Regime

Xiaodong Fan [1], Quanhao Niu [1], Tao Zhao [1] and Banghong Guo [1,2,3,*]

[1] Guangdong Provincial Key Laboratory of Nanophotonic Functional Materials and Devices, Guangdong Provincial Key Laboratory of Quantum Engineering and Quantum Materials, South China Normal University, Guangzhou 510006, China
[2] National Quantum Communication (Guangdong) Co., Ltd., Guangzhou 510700, China
[3] Key Laboratory of Quantum Information, University of Science and Technology of China, Chinese Academy of Sciences, Hefei 230026, China
* Correspondence: guobh@scnu.edu.cn

Abstract: Long block length rate-compatible low-density parity-compatible (LDPC) codes are designed to solve the problems of great variation of quantum channel noise and extremely low signal-to-noise ratio in continuous-variable quantum key distribution (CV-QKD). The existing rate-compatible methods for CV-QKD inevitably cost abundant hardware resources and waste secret key resources. In this paper, we propose a design rule of rate-compatible LDPC codes that can cover all potential SNRs with single check matrix. Based on this long block length LDPC code, we achieve high efficiency continuous-variable quantum key distribution information reconciliation with a reconciliation efficiency of 91.80% and we have higher hardware processing efficiency and lower frame error rate than other schemes. Our proposed LDPC code can obtain a high practical secret key rate and a long transmission distance in an extremely unstable channel.

Keywords: rate compatible; LDPC; continuous-variable quantum key distribution; wide range of SNRs regime

Citation: Fan, X.; Niu, Q.; Zhao, T.; Guo, B. Rate-Compatible LDPC Codes for Continuous-Variable Quantum Key Distribution in Wide Range of SNRs Regime. *Entropy* **2022**, *24*, 1463. https://doi.org/10.3390/e24101463

Academic Editors: Shao-Ming Fei, Ming Li and Shunlong Luo

Received: 21 September 2022
Accepted: 10 October 2022
Published: 13 October 2022

1. Introduction

The cryptosystem based on computational complexity is being challenged by increasingly developed quantum computation. Quantum key distribution (QKD) [1–4], being one-time pad, has been one of the best solutions for its absolute security. QKD enables two remote separated parties named Alice and Bob to extract a symmetrical string of secret keys using a quantum channel.

Currently, there are mainly two types of QKD protocols, called discrete-variable QKD (DV-QKD) [5] and continuous-variable QKD (CV-QKD) [6,7]. In DV-QKD, the information is coded on discrete variables of finite dimensional Hilbert space, such as the polarization or phase of single photon state. In CV-QKD, the information is coded on continuous variables of an infinite-dimensional Hilbert space, including the regular component of coherent state. Compared with the single photon detector used in DV-QKD, homodyne or heterodyne detection techniques, which are used to measure the transmitted quantum states, have already been applied in classical optical communication. Therefore, CV-QKD has great practical advantages for its low cost because of the relatively mature development and being able to transmit in common fiber with classical optical communication. Furthermore, CV-QKD can achieve higher capacity with frequency-multiplexed entanglement source [8].

Due to the imperfection of the quantum channel and potential eavesdropper Eve, the key strings held by Alice and Bob are not consistent, so that a procedure called post-processing is necessary to make them identical. The post-processing of CV-QKD mainly includes four steps: base vector comparison, parameter estimation, information reconciliation and privacy amplification. Information reconciliation is the most important part,

whose performance has a direct correlation to the secret key rate. One of the major factors in information reconciliation is reconciliation efficiency β, which is given by $\beta = R/C$. The R is the rate of key and $C = 0.5log(1 + SNR)$ is the channel compacity. The hardware processing efficiency $\alpha = D_{out}/D_{in}$, where D_{in} represents the data that are input to the hardware device (e.g., Field-programmable Gate Array, FPGA and Graphics Processing Unit, GPU) during information reconciliation and D_{out} represents the output data in unit time [9]. I_{AB} is the mutual information between Alice and Bob. χ_{BE} is the Holevo bound, which is the maximal bound on the information available to the eavesdropper. The factors mentioned above are used to evaluate the performance in a frame, while frame error rate (FER) represents the failure probability of the frames. Ultimately, the practical secret key rate K is given by

$$K = \alpha(1 - FER)(\beta I_{AB} - \chi_{BE}). \tag{1}$$

The parameters mentioned above is related to the error correcting codes, among them low-density parity-compatible (LDPC) code is efficient for CV-QKD [10]. The LDPC code obtained by good degree distribution and reasonable construction method has good error correction performance. The crux of designing a LDPC code is to construct a check matrix which includes check nodes and variable nodes. The degree distribution of check node $\rho(x)$ and variable node $\lambda(x)$ are expressed as:

$$\rho(x) = \sum_{j=2}^{d_c} \rho_i x^{j-1} \tag{2}$$

$$\lambda(x) = \sum_{i=2}^{d_v} \lambda_i x^{i-1}, \tag{3}$$

ρ_i/λ_i is the proportion of the number of edges owned by the check/variable node with degree j/i to the total number of edges in the Tanner graph and d_c/d_v indicates the maximum degree of the check/variable node.

However, quantum is easily influenced in the process of quantum signal preparation and transmission. To realize the free space QKD with satellite [11,12], ship [13], unmanned aerial vehicles [14] or those with orbital angular momentum, we have to take mode distortion, beam wander, weather etc. into account. Therefore, the problems of great variation of quantum channel noise and extremely low signal-to-noise ratio (SNR) have to be solved.

One of the simplest rate-compatible methods for LDPC code is to operate on single-matrix using puncturing, shortening and extending. Furthermore, Gao proposed multi-matrix rate-compatible reconciliation where, in each iteration, multiple matrices produce more useful information to correct errors such that the iteration number falls and the convergence speed increases [15]. However, it inevitably decreases the performance of the original check matrix. Another commonly used way is to construct several check matrices with different code rates to meet the requirements of different SNRs. However, for CV-QKD, the code length has to be longer than 100,000. Base matrices are at least 64,800 long even when we construct the spatially coupled (SC)-LDPC codes or quasi-cyclic (QC)-LDPC codes [16]. As one of the most effective decoding tools of LDPC code, the FPGA has limited hardware resources. To realize high efficiency information reconciliation with FPGA in an extremely unstable channel, it is necessary to construct a single-matrix rate-compatible error correction code. A comparison of the existing works with our proposed LDPC code is shown in Table 1.

In this paper, we first obtain degree distribution with discrete density evolution and differential evolution algorithm. Then we use random construction, progressive edge growth (PEG) algorithm and rate compatible methods of extending and puncturing to construct a check matrix with a code length of 64,800. Finally, we extend the above LDPC code with quasi-cyclic extension to a code length of 648,000. The results show that the proposed codes have a reconciliation efficiency of 91.80%, higher hardware processing efficiency and lower FER than other schemes. Therefore, we can obtain a high practical

secret key rate and a longer transmission distance in an extremely unstable channel with wide range of SNRs.

Table 1. Related works comparison in an unstable channel. Transmission distance is 10 km and the number of check matrix changing times N is 3.

Reference	Hardware Resource	Secret Key Rate (bit/pulse)	Abilitiy to Cope with Channel SNR Changing
Single-matrix rate-compatible reconciliation	a, single matrix	0.0021	low
Multimatrix rate-compatible reconciliation [15]	3a, multimatrix	0.0098	low
Multimatrix corresponding to given SNRs [16]	12a, multimatrix	0.0089	low
Our proposed LDPC code	a, single matrix	0.0116	high

The remainder of this paper is organized as follows. In Section 2, we present some preliminaries of LDPC codes and rate-compatibility. In Section 3, we introduce how to construct our rate compatible (RC)-LDPC code. In Section 4, we present the simulation results and comparisons for the proposed scheme and existing schemes. Finally, the conclusions are drawn in Section 5.

2. Preliminaries

In this section, we first briefly introduce the discrete density evolution and differential evolution, which are used to generate degree distribution. Then we introduce the constructions: random construction, PEG algorithm and QC-LDPC extension, with which we can construct the check matrix with the degree distribution ahead. We also introduce the rate compatible methods: puncturing and extending.

2.1. Methods of Obtaining Degree Distribution

2.1.1. Discrete Density Evolution

Compared with continuous density evolution [17] and Gaussian approximation algorithm [18], discrete density evolution [19] has lower complexity and higher accuracy. Therefore, in this paper, we use discrete density evolution to obtain the optimal degree distribution of LDPC codes. The main steps are as follows:

- We firstly define two functions: quantized function Q and probability mass function S.

$$Q(x) = \begin{cases} \left\lfloor \frac{x}{\Delta} + \frac{1}{2} \right\rfloor, & x \geq \frac{\Delta}{2} \\ \left\lceil \frac{x}{\Delta} - \frac{1}{2} \right\rceil, & x \leq -\frac{\Delta}{2}, \\ 0, & else \end{cases} \tag{4}$$

  $\lfloor x \rfloor$ is the largest integer not greater than x; and $\lceil x \rceil$ is the smallest integer not less than x. The value range of decoded message is $[-L, L]$ and evenly divided into $m = 2^q$ intervals; the quantization interval Δ is given by $2L/m$.

$$S(P_a, P_b) = \sum_{(i,j):k\Delta = R(i\Delta, j\Delta)} P_a[i] \cdot P_b[j]. \tag{5}$$

In which two-input operator R is

$$R(a,b) = Q(\tanh^{-1}(tanh\frac{a}{2}tanh\frac{b}{2})), \tag{6}$$

where a and b are quantized messages.

- The check node and variable node updating of discrete density evolution is

$$p_{\overline{u}}^{(l)} = \sum_{i=2}^{d_r} \rho_i S^{i-1}\left(p_{\overline{v}}^{(l-1)}\right) \tag{7}$$

$$p_{\overline{v}}^{(l)}(k) = p_{\overline{v}}^{(0)}(k) \cdot \sum_{i=2}^{d_v} \lambda_i \otimes^{i-1} (p_{\overline{u}}^{(l)}(k)), \tag{8}$$

$\otimes$ is discrete convolution and l is the iteration number. The initial value $p_{\overline{v}}^{(0)}$ is

$$p_{\overline{v}}^{(0)} = \frac{\sigma}{8\pi} exp\left(-\frac{\left(2-\sigma^2 v\right)^2}{8\sigma^2}\right), v^{(0)} \sim N\left(\frac{2}{\sigma^2}, \frac{4}{\sigma^2}\right). \tag{9}$$

- Finally, we calculate the error rate with

$$p_{\overline{e}}^{(l)} = p_{\overline{v}}^{(l)}(0) + \sum_{k=-m/2}^{-1} p_{\overline{v}}^{(l)}(k). \tag{10}$$

End the procedure when the $p_{\overline{e}}^{(l)}) < 0$ or l reaches the maximum number of iterations. Otherwise, we continue to update the check node and variable node.

Discrete density evolution is first proposed to obtain the noise threshold according to the degree distribution ρ_i and λ_i. In our work, we use it to obtain the degree distribution under specific channel noise.

2.1.2. Differential Evolution

Stom first proposed the differential evolution algorithm in 1995 to solve the optimization problem [20]. It uses differential mutation operator and crossover operator to generate new individuals by the way of survival of the fittest. Based on this method, we can obtain the optimal degree distribution under specific channel noise.

- Set channel noise threshold σ, target error probability P_e, maximum number of iterations l_{max}, maximum degree of variable node d_v and the number of terms of degree distribution polynomial n.
- Randomly generate NP vectors $P_{i,G}, i = 1, 2, \ldots, NP$ for the degree distribution of variable node. Use discretized density to evolve each vector and obtain the respective error probability $P_{e_i,G}$. The vector with the lowest error probability is marked as the best vector $P_{best,G}$ and its error probability is marked as $P_{e_{best},G}$.
- For each i, randomly choose four vectors from set of $P_{i,G}$ and the new vector is updated by

$$v_{i,G+1} = P_{best,G} + 0.5\left(P_{r_1,G} - P_{r_2,G}\right) + 0.5\left(P_{r_3,G} - P_{r_4,G}\right). \tag{11}$$

Calculate the corresponding error probability $P_{v_i,G+1}$ for each new vector $v_{i,G+1}$.

- For each i, compare $P_{e_i,G}$ with $P_{v_i,G+1}$ and let $P_{i,G+1} = v_{i,G+1}$ if $P_{e_i,G} > P_{v_i,G+1}$. The vector with the lowest error probability is marked as the best vector $P_{best,G+1}$ and its error probability is marked as $P_{e_{best},G+1}$.

- If the error probability corresponding to the best vector $P_{e_{best},G+1} > P_e$, update the vectors again and return to step (4). If $P_{e_{best},G+1} \leq P_e$, the $P_{best,G+1}$ is the ideal vector that we want.

2.2. Constructions

In this work, we use random construction, the PEG algorithm and QC extension for their good results in various situations.

2.2.1. Random Construction

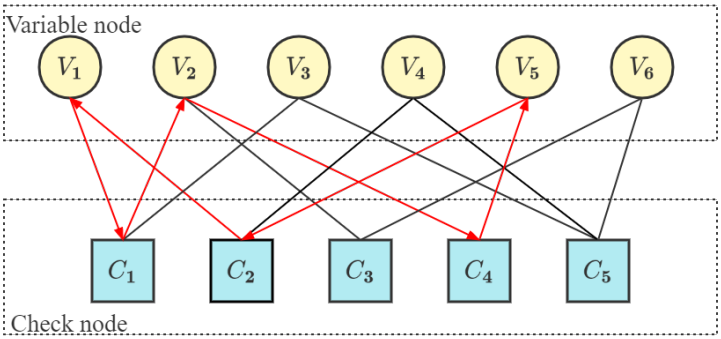
Various random constructions have been proposed based on the same core thought, that is, place non-zero elements in random unfilled positions in the check matrix without violating any set constraint. There are two constraint rules: one is that line l_i contains X_i "1" and column c_i contains Y_i "1" according to the degree distribution of check nodes and variable nodes; the other one is the number of elements "1" at the same position in any two rows or columns is less than or equal to 1. It means that the shortest girth has to be longer than 4.

2.2.2. Progressive-Edge-Growth Algorithm

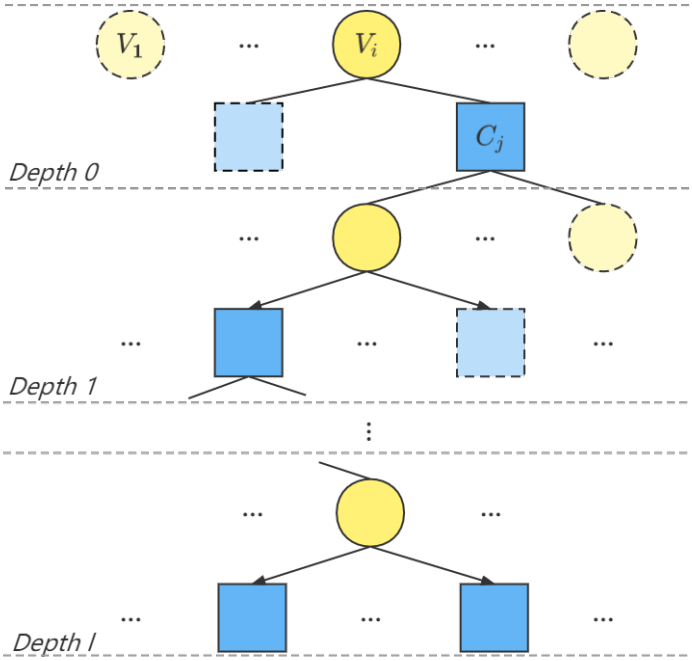
Before introducing the PEG algorithm, we first introduce a common representation of LDPC codes—the Tanner diagram and several concepts. As shown in Figure 1a, V_i is a variable node, C_j is a check node and the line between them is called an edge. If two nodes are connected with each other, we say these two nodes are adjacent to each other. The girth is defined as the minimum number of lines that comes from a node and back to this node, whose intermediate node is only passed once. As shown in Figure 1a, the shortest girth is 6 and one of them is $V_1 \to C_1 \to V_2 \to C_4 \to V_5 \to C_2 \to V_1$, for instance.

For the PEG algorithm, new edges are added to make the loop girth in the Tanner diagram corresponding to the check matrix as large as possible. As shown in Figure 1b, the steps are as follows:

- Determine the number of check node, variable node and the degree distribution of variable node.
- Randomly choose a variable node V_i and find the check node C_j with the least number of connected edges in the Tanner graph. Then connect the variable node V_i and the check node C_j with an edge and take it as the first edge of the variable node V_i.
- Take the variable node V_i as the root node and expand the current Tanner diagram. When the expansion depth is l, the set of check nodes adjacent to V_i is recorded as $N^l_{V_i}$. The $\overline{N^l_{V_i}}$ is the complement set of $N^l_{V_i}$, where the complete set V_c is the set of all variable nodes. Expand the Tanner graph with the root node and the depth of l. When $\overline{N^l_{V_i}} \neq \varnothing$, $\overline{N^{l+1}_{V_i}} = \varnothing$ and the number of nodes contained in $N^l_{V_i}$ stops increasing but is still less than the number of matrix rows l, connect the check node C_j with the least number of connected edges to the variable node V_i.
- Repeat step (2) to add edges to the selected variable nodes until all of them are added.
- Repeat steps (1) to (3) to add edge for all other variable nodes.



(a)



(b)

Figure 1. (**a**) Tanner graph; (**b**) PEG algorithm.

2.2.3. QC-LDPC Extension

QC-LDPC extension is uniquely determined by the dimension and shift times of the circulant matrix. Its quasi-cyclic characteristics make the process of coding and decoding more efficient. Compared with randomly constructed LDPC codes, QC-LDPC codes have lower error level and are more convenient for storage and hardware implementation. We multiply the corresponding positions of the base matrix H_b and the coefficient matrix H_c and we define this operation as $\odot$, the expression is expressed as follows:

$$H_b \odot H_c = \begin{bmatrix} B_{1,1} & \cdots & B_{1,i} \\ \vdots & \ddots & \vdots \\ B_{j,1} & \cdots & B_{j,i} \end{bmatrix} \odot \begin{bmatrix} C_{1,1} & \cdots & C_{1,i} \\ \vdots & \ddots & \vdots \\ C_{j,1} & \cdots & C_{j,i} \end{bmatrix} = \begin{bmatrix} B_{1,1}C_{1,1} & \cdots & B_{1,i}C_{1,i} \\ \vdots & \ddots & \vdots \\ B_{j,1}C_{j,1} & \cdots & B_{j,i}C_{j,i} \end{bmatrix}. \quad (12)$$

Take lifting size of 3 as an example, the elements of the base matrix are 0 and 1, and the elements of the coefficient matrix are 1, 2 and 3. Then the matrix elements are replaced by

the cyclic permutation matrices (CPMs). We replace 0 with zero matrices, 1 with $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, 2 with $\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$ and 3 with $\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$.

2.3. Methods of Rate-Compatible

Puncturing is a method that makes the code rate change from low to high. As shown in Figure 2a, the submatrix A are information bits and submatrix B and C are check bits. The initial code rate is $R = L_0/(L_0 + L_1 + L_2)$. By deleting the submatrix C, we can obtain a code rate increasing to $R = L_0/(L_0 + L_1)$.

On the contrary, extending as shown in Figure 2b enables the code rate to change from high to low. We first construct a check matrix A with the high bit rate of $(N_0 - M_0)/N_0$. Moreover, by adding the submatrix A_n, we extend the matrix to make it compatible for the low rate. The code rate is expressed as:

$$R_i = \frac{\sum_0^n N_i - \sum_0^n M_i}{N}. \tag{13}$$

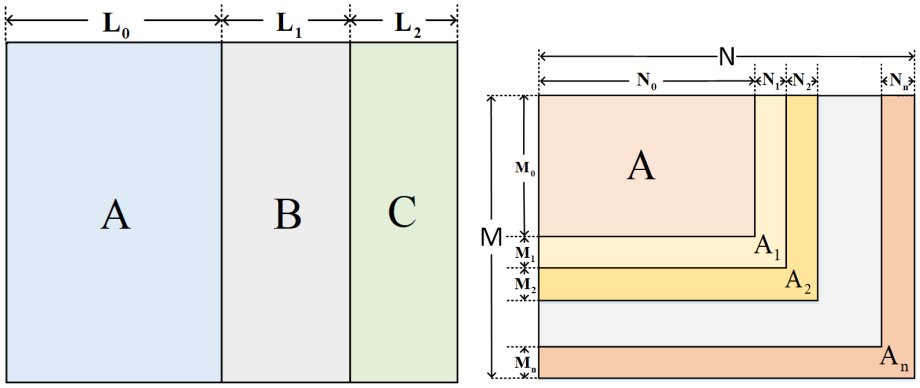


Figure 2. The rate-compatible method: (**a**) puncturing; (**b**) extending.

3. Proposed Check Matrix for RC-LDPC Codes with Wide Range of SNRs Regime

From the Equation (1) we can see that high hardware processing efficiency and reconciliation efficiency result in a good performance of final secret key rate for a given SNR. Proper degree distribution and reasonable construction method lead to good error correction performance.

3.1. Obtaining Degree Distribution

We first obtain the initial optimal degree distribution using discretize density evolution and differential evolution refer to Sections 2.1.1 and 2.1.2. Maximum degree of variable node and the number of terms of degree distribution polynomial are set as 10 and 4, respectively.

From the initial optimal degree distribution, we find that the pairs of degree distribution are distributed nearby λ_3 and λ_7 except of λ_2 and λ_{10}. Therefore, we calculate the average number of λ_3 and λ_7 at rate from 0.3 to 1, i.e., SNR from 0.1 to 3 (the degree distribution is appropriate to the SNR larger than 3 but the maximum rate 1 corresponds to the SNR of 3). The initial values are average number $\overline{\lambda_3}$ and $\overline{\lambda_7}$, and maximum degree of variable node and the number of terms of degree distribution polynomial are still set as 10 and 4. The difference is that the degree distribution of the variable distribution is set on the

λ_2, λ_3, λ_7 and λ_{10} instead of a random distribution. Then we repeat the above operations to obtain the optimal degree distribution in these conditions.

Through the above operations, we obtain the degree, the maximum degree of the variable node, and the number of terms of the degree distribution polynomial. Ultimately, we calculate the optimal degree distribution for proposing our LDPC code with Algorithm 1.

Algorithm 1 Obtaining the ultimate variable degree distribution with density evolution and differential evolution

Input: Target error probability P_e, maximum number of iterations l_{max}, population size $NP = 50$, the number of terms of variable node degree distribution polynomial $l = 5$, the highest power of variable node degree distribution, $\lambda_3 = 0.0047$, $\lambda_7 = 0.5072$

Output: Error rate $P_{e_{best}}$, vector P_{best}

for i= 1 to NP **do**
  refer to Section 2.1.1 generate vector P_i with λ_2, λ_3, λ_7, λ_8 and λ_{10}, $\lambda_2 + \lambda_8 + \lambda_{10} = 0.4881$;
  calculate the error probability P_{e_i};
  if $P_{e_{best}} > P_{e_i}$ **then**
    $P_{e_{best}} \leftarrow P_{e_i}$; $P_{best} \leftarrow P_i$;
  end if
end for
for $j = 1$ to l_{max} **do**
  randomly choose four numbers r_1, r_2, r_3 ,r_4 from 1 to NP;
  v_j = P_{best} + 0.5(P_{r_1} - P_{r_2} + P_{r_3} - P_{r_4});
  calculate the error probability P_{e_j};
  if $P_{e_{best}} < P_e$ **then**
    output v_j;
  end if
  if $P_{e_{best}} > P_{e_j}$ **then**
    $P_{e_{best}} \leftarrow P_{e_j}$;
  end if
end for

Table 2 is the result of Algorithm 1, whose input signal $X \sim (0, 1)$ and additive white Gaussian noise $Z \sim (0, \sigma^2)$ are random variables that obey Gaussian distribution and independent of each other. The channel noise $SNR = 1/\sigma^2$ and σ represents the maximum allowed value of noise for the additive white Gaussian channel. For $\rho(x) = \lambda(x) = 1$, the check node degree distribution is definite with the constraint condition $r = 1 - \int_0^1 \rho(x)dx / \int_0^1 \lambda(x)dx$. The degree distribution in our scheme especially decreases the difficulty of constructing the check matrix.

Table 2. Variable nodes degree distribution pairs for the code rate from 0.3 to 1.0.

Rate	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0** [1]
λ_2	0.0001	0.0001	0.0007	0.0001	0.0002	0.0002	0.0004	0.0005
λ_3				0.0047				
λ_7				0.5072				
λ_8	0.1382	0.1268	0.1044	0.0761	0.0480	0.0367	0.0281	0.0089
λ_{10}	0.3498	0.3612	0.3830	0.4119	0.4399	0.4512	0.4596	0.4787
σ	1.3868	1.1547	1.0000	0.8771	0.7809	0.7001	0.6337	0.5774
SNR	0.52	0.75	1.00	1.30	1.64	2.04	2.49	3.00
C	0.3072	0.4037	0.5000	0.6008	0.7003	0.8020	0.9016	1.0000

[1] The practical rate at 1 is close to but lower than 1.

In order to maximize the use of limited key resources, we still need to fully consider the condition of rate lower than 0.1. Obviously, the secret key rate is low for the low mutual

information I_{AB}. Therefore, in order to simplify our work, the degree distribution pairs we choose for the rate lower than 0.1 are directly refer to Appendix A [21,22].

3.2. Constructing Check Matrix for RC-LDPC Code

With the degree distribution we obtained above, we construct a single matrix RC-LDPC code simultaneously with the random construction, the PEG algorithm, and QC-LDPC codes mentioned in Section 2. The structure of the check matrix is shown in Figure 3 and combined with parts A, B and C.

The part A is a shared part for the rate from 0.1 to 1, which is constructed with $\overline{\lambda_3}$ and $\overline{\lambda_7}$. This structure has the advantage of reducing computational complexity and saving the storage resources. Previous work showed that the PEG algorithm has better performance at SNR$\sim$3 [23], while random construction exhibits better performance at SNR$\sim$1 [24]. Therefore, the construction that we use to construct the sub-matrix A is the PEG algorithm.

The part B is constructed with rest of degree distribution to realize the rate-compatible method of puncturing. In order to further improve the performance of our LDPC code, we construct the check matrix with the thought of puncturing. More specifically, we divide submatrix B_n into two part and construct one part when the R decreases every 0.05. For rate from 0.3 to 0.1, this number is 0.1. We use PEG algorithm to construct B_1 to B_5 and random construction to construct extra part. Moreover, the structure of part B is a lower triangular matrix, which can be directly encoded.

Multi-edge-type (MET)-LDPC codes are employed with low SNRs due to their good error-correcting performances, more amenable decoding complexity and also being able to be rate-compatible at low rates [25]. Based on the check matrix above, we construct part C with degree distribution of the MET-LDPC codes from Appendix A for the rate from 0.01 to 0.1.

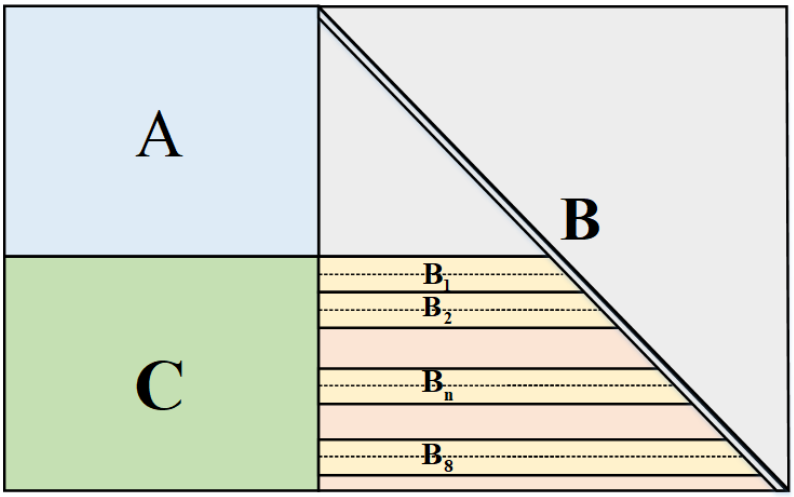


Figure 3. The check matrix for RC-LDPC codes with wide range of SNR.

4. Simulation Experiment

In this section, we summarize the implementation results of the proposed LDPC codes over an unstable channel. Our purpose is to construct a RC-LDPC code with single matrix that can be adapt to the SNR from 0.01 to 15. We show the performance of reconciliation efficiency β, hardware processing efficiency α and FER, which are influenced by the change of SNR. Furthermore, the decoding algorithm is a modified Min-Sum algorithm.

The reconciliation efficiency comes from $\beta = R/C$. Referring to the construction mentioned in Section 3.2, we change the check matrix when R reduces to a certain extent. When R is from 0.3 to 1, C decreases 0.1 to an integer multiple of 0.1. When R is from 0.01 to 0.3, C decreases 0.05 to an integer multiple of 0.05. In Figure 4, assuming that the channel noise is uniformly distributed, the LDPC code we proposed has an average reconciliation efficiency β of 91.80%, and for higher rates from 0.3 to 1 this number is 96.13%. Because

the data with rate lower than 0.3 only have a little contribution to reconciliation efficiency, the practical reconciliation efficiency is close to 96.13%. Compared with the existing scheme, the proposed LDPC code has a relatively high reconciliation efficiency.

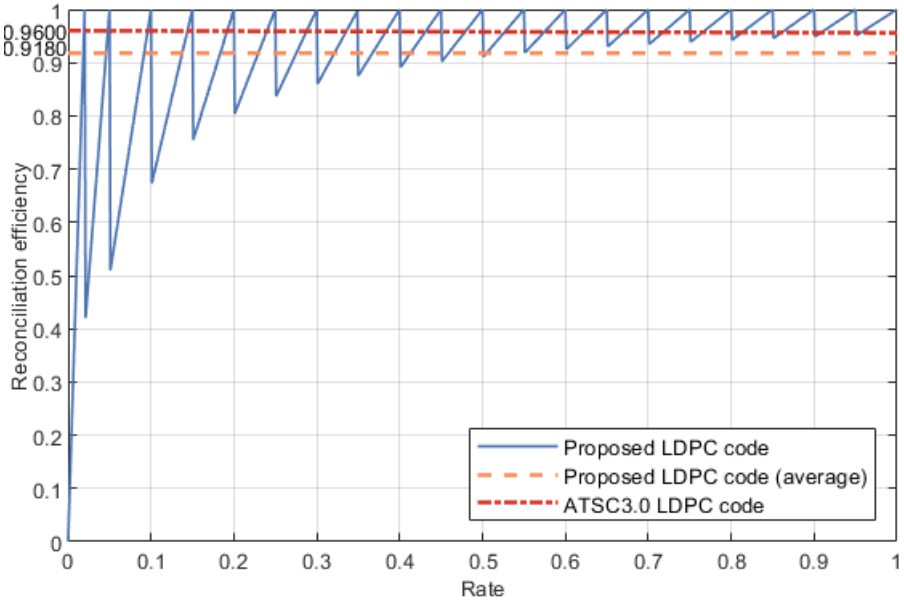


Figure 4. The reconciliation efficiency for different code rate.

From Equation (1), the secret key rate is also related to the hardware processing efficiency α, which is equal to the ratio of D_{out} and D_{in}. More specifically, supposing the times used to load check matrix, load data and decode data are t_{lm}, t_{ld} and t_{dd}, separately. The number of times that check matrix has to be reloaded is n and the number of data blocks that have to be processed is m. Suppose the secret key rate that optical system can provide is M, the number of data blocks m is M/L. The hardware processing efficiency α is

$$\alpha = \frac{1}{nt_{lm} + m(t_{ld} + t_{dd})} \tag{14}$$

Because of the finite-size effects, the block length in the procedure of privacy amplification is at least 10^7, which also takes up abundant hardware resource [26,27], so that not all the check matrices can be stored in advance. The reconciliation efficiency will be reduced quickly even if the SNR changes in a very small range. Therefore, other schemes have to reload the appropriate check matrix and then load and decode data when the rate is higher than the channel capacity. With our proposed LPDC code, we save the time of reloading the check matrix. For the block length of 648,000, the times used to load data and decode data we tested with the FPGA Arria 10 are 13.0 ms and 211.2 ms. Furthermore, the average time we used to load check matrix of ATSC 3.0 LDPC codes is 11.1ms. From the Figure 5, we can see that our work keeps a high hardware processing efficiency α with the number of check matrix changing times n increases. Meanwhile, difference of hardware reconciliation efficiency between our proposed LDPC code and ATSC 3.0 LDPC code also increases.

Frame error rate is the rate that a data block failed to be decoded. It is mainly caused by two reasons: the defect of error correcting code and decoding algorithm; the unadaptable check matrix led by the changing of SNR. The FER caused by the defect of error correcting code and decoding algorithm can be reduced to 3.25×10^{-3}, which is far lower than the FER led by the latter reason [28]. Therefore, we only take the latter reason into account. It can be seen from Figure 6 that with the number of check matrix changes increases, our proposed LDPC code has a lower FER than the other scheme.

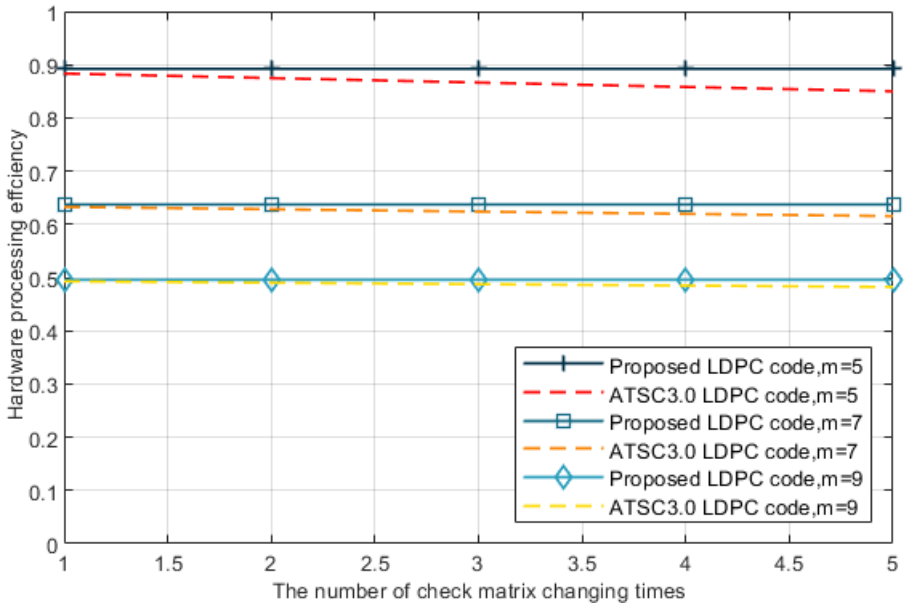


Figure 5. The hardware processing efficiency α influenced by the number of check matrix changing times n.

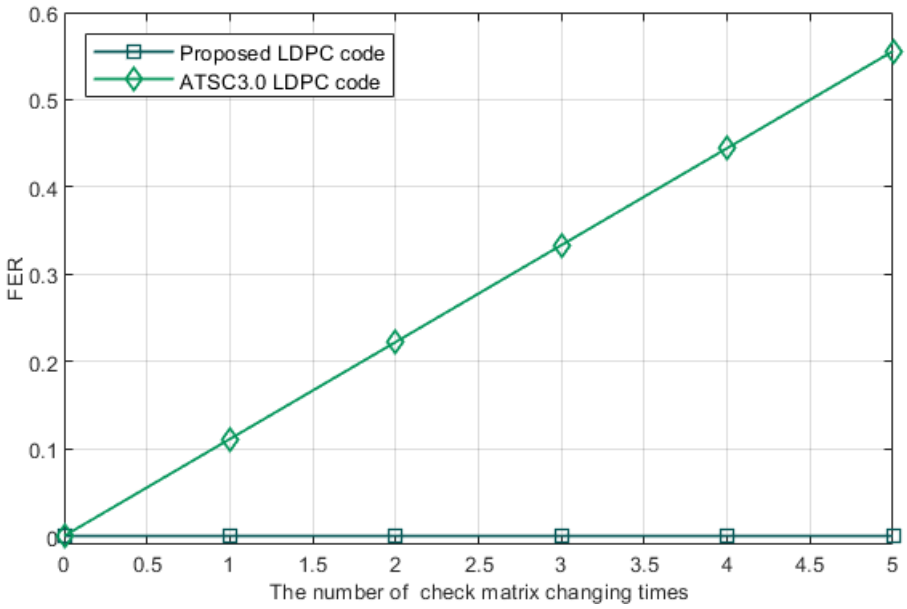


Figure 6. FER influenced by the number of check matrix changing times N. The number of data blocks that have to be processed is nine.

Given the excess noise, efficiency of receiver's detector and electronic noise at Bob's side, we can calculate the practical secret key rate [29]. Figure 7 is the comparison of the practical secret key rate of the proposed LDPC code and ATSC 3.0 LDPC codes. As can be seen in the graph, our scheme has a better performance with same number of check matrix changes N and has a lower performance reduction when the N increases. This comes from the fact that combined action of reconciliation efficiency β, hardware processing efficiency α and FER.

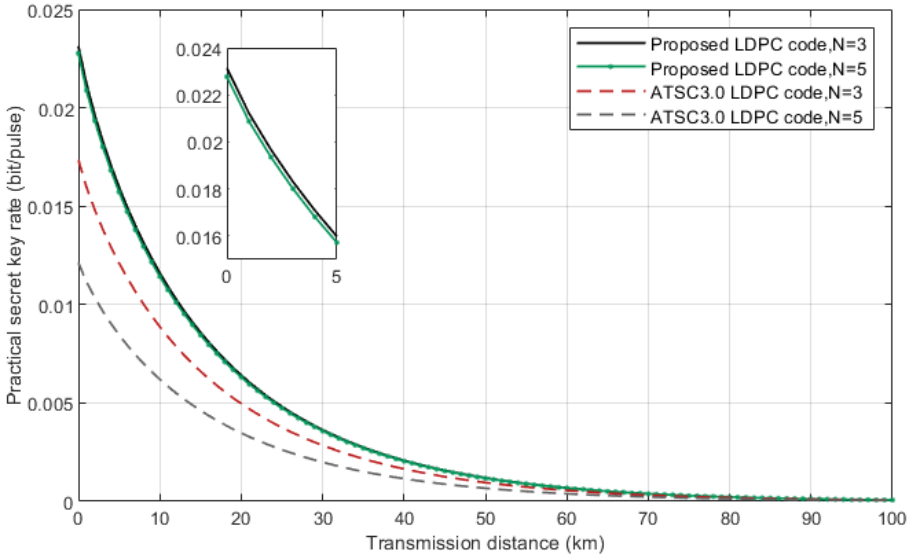


Figure 7. Practical secret key rate with reconciliation efficiency of 91.80% for our proposed LDPC code and 96.00% for ATSC 3.0 LDPC code. The extra parameters $\varepsilon = 0.01$, $\eta = 0.64$ and $V_{el} = 0.1$.

5. Conclusions

In this study, we design a rule of proposing a RC-LDPC code with single matrix for SNRs between 0.01 and 15 to solve the problems of great variation of quantum channel noise and extremely low SNR. First, we use the discretized density evolution algorithm and differential evolution to acquire good node degree distribution pairs of LDPC codes. Then, with construction methods including PEG algorithm, random construction, quasi-cyclic extension and rate-compatible methods including extending and puncturing, we proposed a convenient and efficient construction method for designing a RC-LDPC code. Considering the number of check matrix changing times led by the change of SNR, the result shows that we have a reconciliation efficiency of 91.80%, higher hardware processing efficiency and lower FER. It has a good performance especially in an extremely unstable channel.

Author Contributions: Conceptualization, X.F.; methodology , X.F., Q.N. and T.Z.; software, X.F.; validation, X.F.; formal analysis, X.F.; investigation, X.F.; resources, B.G.; data curation, X.F.; writing—original draft preparation, X.F.; writing—review and editing, X.F., Q.N. and T.Z.; visualization, X.F.; supervision, B.G.; funding acquisition, B.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Key-Area Research and Development Program of Guangdong Province (Grant No. 2018B030325002).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We thank Peng-Cheng Wang, Bo-Wen Dong and Jie Jia for their helpful discussions.

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

Appendix A

Table A1. Degree distribution pairs of code rate from 0.01 to 0.1.

Rate	Degree Distribution	σ	SNR	C
0.1	$v(r,x) = 0.0775r_1x_1^2x_2^{20} + 0.0475r_1x_1^3x_2^{22} + 0.875r_1x_3$ $\mu(x) = 0.0025x_1^{11} + 0.0225x_1^{12} + 0.03x_2^2x_3 + 0.845x_2^3x_3$	2.541	0.15	0.0488
0.05	$v(r,x) = 0.04r_1x_1^2x_2^{34} + 0.03r_1x_1^3x_2^{34} + 0.93r_1x_3$ $\mu(x) = 0.01x_1^8 + 0.01x_1^9 + 0.41x_2^2x_3 + 0.52x_2^3x_3$	5.91	0.03	0.0213
0.02	$v(r,x) = 0.0225r_1x_1^2x_2^{34} + 0.0175r_1x_1^3x_2^{34} + 0.96r_1x_3$ $\mu(x) = 0.010625x_1^3 + 0.009375x_1^7 + 0.6x_2^2x_3 + 0.36x_2^3x_3$	2.541	0.15	0.1008

References

1. Gisin, N.; Ribordy, G.; Tittel, W.; Zbinden, H. Quantum cryptography. *Rev. Mod. Phys.* **2001**, *74*, 145–195. [CrossRef]
2. Scarani, V.; Bechmann-Pasquinucci, H.; Cerf, N.J.; Dušek, M.; Lütkenhaus, N.; Peev, M. The security of practical quantum key distribution. *Rev. Mod. Phys.* **2009**, *81*, 1301. [CrossRef]
3. Zhu, M.; Hu, M.; Guo, B. Free-Space QKD with Modulating Retroreflectors Based on the B92 Protocol. *Entropy* **2022**, *24*, 204. [CrossRef]
4. Hua X, Hu M, Guo B. Multi-User Measurement-Device-Independent Quantum Key Distribution Based on GHZ Entangled State. *Entropy* **2022**, *24*, 841. [CrossRef] [PubMed]
5. Alia, O.; Tessinari, R.S.; Bahrani, S.; Bradley, T.D.; Sakr, H.; Harrington, K.; Hayes, J.; ; Chen, Y.; Petropoulos, P.; Richardson, D.; et al. DV-QKD Coexistence With 1.6 Tbps Classical Channels Over Hollow Core Fibre. *J. Light. Technol.* **2022**, *40*, 5522–5529. [CrossRef]
6. Weedbrook, C.; Pirola, S.; García-Patrón, R.; Cerf, N.J.; Ralph, T.C.; Shapiro, J.H.; Lloyd, S. Gaussian quantum information. *Rev. Mod. Phys.* **2011**, *84*, 621–669. [CrossRef]
7. Jain, N.; Chin, H.M.; Mani, H.; Lupo, C.; Nikolic, D.S.; Kordts, A.; Pirola, S.; Pedersen, T.B.; Kolb, M.; Omer, B.; et al. Practical continuous-variable quantum key distribution with composable security. *Nat. Commun.* **2022**, *13*, 4740. [CrossRef]
8. Kovalenko, O.; Ra, Y.S.; Cai, Y.; Usenko, V.C.; Fabre, C.; Treps, N.; Filip, R. Frequency-multiplexed entanglement for continuous-variable quantum key distribution. *Photonics Res.* **2021**, *9*, 2351–2359. [CrossRef]
9. Yang, S.; Lu, Z.G.; Li, Y. High-Speed Post-Processing in Continuous-Variable Quantum Key Distribution Based on FPGA Implementation. *J. Light. Technol.* **2020**, *38*, 3935. [CrossRef]
10. Mink, A.; Nakassis, A. LDPC for QKD Reconciliation. *arXiv* **2012**, arXiv:1205.4977.
11. Chen, Y.A.; Zhang, Q.; Chen, T.Y.; Cai, W.Q.; Liao, S.K.; Zhang, J.; Chen, K.; Yin, J.; Ren, J.G.; Chen, Z.; et al. An integrated space-to-ground quantum communication network over 4600 kilometres. *Nature* **2021**, *589*, 214–219. [CrossRef]
12. Yin, J.; Cao, Y.; Li, Y.H.; Liao, S.K.; Zhang, L.; Ren, J.G.; Pan, J.W. Satellite-based entanglement distribution over 1200 kilometers. *Science* **2017**, *356*, 1140–1144. [CrossRef]
13. Zhao, W.; Shi, R.; Ruan, X.; Guo, Y.; Mao, Y.; Feng, Y. Monte Carlo-based security analysis for multi-mode continuous-variable quantum key distribution over underwater channel. *Quantum Inf. Process.* **2022**, *21*, 186. [CrossRef]
14. Liu, H.Y.; Tian, X.H.; Gu, C.; Fan, P.; Ni, X.; Yang, R.; Zhang, J.N.; Hu, M.; Guo, J.; Zhu, S.N.; et al. Drone-based entanglement distribution towards mobile quantum networks. *Natl. Sci. Rev.* **2020**, *5*, 921–928. [CrossRef]
15. Gao, C.H.; Guo, Y.; Jiang, D.; Liu, J.; Chen, L.J. Multimatrix rate-compatible reconciliation for quantum key distribution. *Phys. Rev. A* **2020**, *102*, 022604. [CrossRef]
16. 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]
17. Richardson, T.J. The capacity of low-density parity-check codes under message-passing decoding. *IEEE Trans. Inf. Theory* **2001**, *47*, 599. [CrossRef]
18. Chung, S.Y.; Richardson, T.J.; Urbanke, R.L. Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation. *Inf. Theory IEEE Trans.* **2001**, *47*, 657–670. [CrossRef]
19. Chung, S.Y. On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit. *IEEE Commun. Lett.* **2002**, *5*, 58–60. [CrossRef]
20. Storn, R.; Price, K. Differential Evolution—A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
21. Wang, X.; Zhang, Y.C.; Li, Z.; Xu, B.; Yu, S.; Guo, H. Efficient rate-adaptive reconciliation for continuous-variable quantum key distribution. *arXiv* **2017**, arXiv:1703.04916.
22. 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]
23. Bai, Z.; Wang, X.; Yang, S.; Li, Y. High-efficiency Gaussian key reconciliation in continuous variable quantum key distribution. *Sci. China Phys. Mech. Astron.* **2016**, 59, 614201. [CrossRef]

24. Bai, Z.; Yang, S.; Li, Y. High-efficiency reconciliation for continuous variable quantum key distribution. *Jpn. J. Appl. Phys.* **2017**, *56*, 044401. [CrossRef]
25. Jeong, S.; Jung, H.; Ha, J. Rate-compatible multi-edge type low-density parity-check code ensembles for continuous-variable quantum key distribution systems. *NPJ Quantum Inf.* **2022**, *8*, 6. [CrossRef]
26. Zhang, C.M.; Li, M.; Huang, J.Z.; Li, H.W.; Li, F.Y.; Wang, C.; Yin, Z.Q.; Chen, W.; Han, Z.F.; Sripimanwat, K.; et al. Fast implementation of length-adaptive privacy amplification in quantum key distribution. *Chin. Phys. B* **2014**, *23*, 090310. [CrossRef]
27. Yan, B.; Li, Q.; Mao, H.; Chen, N. An efficient hybrid hash based privacy amplification algorithm for quantum key distribution. *Quantum Inf. Process.* **2022**, *21*, 130. [CrossRef]
28. Shi, J.J.; Li, B.P.; Huang, D. Reconciliation for CV-QKD using globally-coupled LDPC codes. *Chin. Phys. B* **2020**, *29*, 040301. [CrossRef]
29. 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]