Quantum Information Processing (2024) 23:137
https://doi.org/10.1007/s11128-024-04343-8

Efficient information reconciliation in quantum key distribution systems using informed design of non-binary LDPC codes

Debarnab Mitra[1] · Jayanth Shreekumar[1] · Lev Tauz[1] · Murat Can Sarihan[1] · Chee Wei Wong[1] · Lara Dolecek[1]

Received: 29 November 2023 / Accepted: 7 March 2024 / Published online: 3 April 2024

Abstract

In quantum key distribution (QKD), two users extract a shared secret key using a quantum communication channel in the presence of an eavesdropper. Among QKD protocols, the ones based on energy-time (ET) entanglement of photons have been studied extensively due to their ability to generate high key rates from the arrival times of entangled photons. For the *information reconciliation* (IR) stage of ET-QKD protocols (where the users communicate using a classical channel in order to reconcile differences in their data), a scheme called *multi-level coding* (MLC) was proposed by Zhou et al. in prior work. The MLC scheme splits the raw key symbols into bit layers and utilizes binary low-density parity check (LDPC) codes to encode each layer. Although binary LDPC codes are able to offer low complexity decoding for IR, they have poor error-correcting performance compared to their non-binary counterparts, thus leading to low key rates. Additionally, existing LDPC codes do not fully utilize the properties of the QKD channel to optimize the key rates. In this paper, we mitigate the above issues by proposing a flexible protocol for IR in ET-QKD systems

✉ Debarnab Mitra
debarnabucla@ucla.edu

Jayanth Shreekumar
jayshreekumar98@ucla.edu

Lev Tauz
levtauz@ucla.edu

Murat Can Sarihan
mcansarihan@ucla.edu

Chee Wei Wong
cheewei.wong@ucla.edu

Lara Dolecek
dolecek@ee.ucla.edu

[1] Department of Electrical and Computer Engineering, University of California, Los Angeles, Los Angeles, USA

called non-binary multi-level coding NB-MLC(a) which is parameterized by a positive integer a. The NB-MLC(a) protocol is a generalization of the MLC scheme and utilizes NB-LDPC codes from a Galois field of size 2^a. We show that by using a small value of a, the NB-MLC(a) protocol significantly improves the key rate without much increase in complexity. To further improve the key rates of the NB-MLC(a) protocol, we propose (i) a joint rate and degree distribution optimization (JRDO) algorithm to design the NB-LDPC codes for the protocol and (ii) an interleaved decoding and communication (IDC) scheme to decode the different layers of the NB-MLC(a) protocol. The JRDO algorithm is designed to use the QKD channel information, and we show that it results in a higher key rate than codes used in prior work. Additionally, the IDC scheme improves the key rate compared to the decoding and communication methods utilized previously in literature. Overall, the NB-MLC(a) protocol that uses JRDO-LDPC codes and the IDC scheme results in a significant 40–60% improvement in key rates compared to prior work for ET-QKD systems.

Keywords Quantum key distribution · Information reconciliation · Low-density parity check codes

1 Introduction

Quantum key distribution (QKD) provides a physically secure way to share a secret key between two users, Alice and Bob, over a quantum communication channel in the presence of an eavesdropper Eve [1–7]. Secret keys in QKD systems are established by first performing a *quantum stage* where Alice and Bob exchange quantum states over a quantum channel. The quantum stage is succeeded by a *post-processing stage* that occurs over a classical communication channel. At the end of the two stages, Alice and Bob ideally arrive at identical random sequences (the secret key) which are only known to them. The ultimate goal of a QKD protocol is to achieve a high secret key rate, i.e., to extract a high number of bits in the secret key per generated photon. QKD protocols based on energy-time (ET) entanglement of photons have the potential to achieve this goal due to their high-dimensional nature where multiple bits can be extracted from each generated entangled photon pair [5, 8, 9]. Additionally, ET-QKD protocols also provide unconditional security through non-local Franson and conjugate-Franson interferometry [8] that is critical for secure communications.

At a high level, an ET-QKD protocol consists of three steps [6]: *i) raw key generation ii) information reconciliation (IR)* and *iii) privacy amplification (PA)*. Raw key generation takes place during the quantum stage where Alice and Bob generate raw keys using a quantum communication channel. The use of the quantum channel prevents undetected eavesdropping by Eve. However, due to the transmission noise in the quantum channel as a result of issues such as timing jitters, photon losses, and dark counts, the raw keys at Alice and Bob may disagree in some positions. The raw key may also be partly known to Eve and may not be uniformly random given Eve's knowledge. These shortcomings are overcome in the post-processing stage that consists of the IR and PA steps. In the IR step, Alice and Bob communicate over a classical channel (public and accessible to Eve) to reconcile the differences in the

raw keys to obtain reconciled keys that Eve may have some knowledge about. The IR step is followed by the PA step, where Alice and Bob compress their reconciled key sequences by accounting for Eve's knowledge to amplify the privacy of the key and to achieve uniform randomness. At the end of the above three steps, Alice and Bob end up with a shared secret key known only to them, or they had aborted the protocol [7]. In this paper, we focus on the IR step of the ET-QKD protocol, which has a significant impact on the overall secret key rate of the system.

Error-correcting codes (ECC) [10] are a major mathematical tool used in the IR step [5, 6, 8, 11–16] to overcome the transmission noise in the raw key generation step and ensure that Alice and Bob arrive at an identical sequence of symbols. Any information leaked to Eve during the IR step must be subtracted from the final secret key during privacy amplification [17, 18]. Thus, in order to study the performance of various IR protocols, we define the *IR rate* $\mathcal{R}_{IR}$ of the system (in bits per photon) as

$$\mathcal{R}_{IR} = \mathbb{E}\left[\frac{L_{IR} - \text{leak}_{IR}}{N}\right], \tag{1}$$

where L_{IR} is the length (in bits) of the reconciled key, leak_{IR} is the length (in bits) of the information leaked to Eve during IR, N is the number of entangled photon pairs, and $\mathbb{E}[\ ]$ denotes the expectation operator. A high IR rate results in a high secret key rate in the system, and, in this paper, we provide techniques to improve the IR rate compared to existing schemes.

IR protocols for binary-based QKD systems, where a single bit is exacted from each generated photon, have been extensively researched in the literature. However, very little effort has been placed into optimizing IR protocols for high-dimensional QKD systems (that extract multiple bits from each generated photon) apart from the introduction of a protocol called *multi-level coding* (MLC) [6] in 2013 which has been considered for works such as [5, 19]. In the MLC protocol, the sequence of symbols after the raw key generation step is converted into multiple bit layers and then each bit layer is sequentially reconciled using binary Low-Density Parity-Check (LDPC) codes. Due to the low complexity of decoding binary LDPC codes, the MLC protocol results in a low key generation complexity. However, binary LDPC codes have poor error-correcting performance compared to their non-binary counterparts leading to reduced IR rates. On the other hand, a *fully non-binary (FNB) protocol* defined as an IR protocol that uses a non-binary (NB) LDPC code to directly encode/decode the generated raw key symbols can naturally lead to higher IR rates. However, the symbols in the key generation step can belong to a Galois field of size as large as 2^{10} and it is known that iterative decoding of NB-LDPC codes has a very high complexity (log-linear in the field size [20]) at large field sizes. Hence, an FNB protocol with a large field size is not favorable in QKD applications requiring low complexity, such as in [21, 22]. Apart from the above techniques of IR in ET-QKD systems, various other ECC techniques have been used for IR, however, in the continuous-variable (CV) QKD setting [23]. For example, spatially coupled (SC) LDPC codes [11], irregular repeat accumulate (IRA) and SC-IRA codes [12], polar codes [13, 14], and spinal codes [15] have been used for CV-QKD. However, these techniques involve a different method of IR compared to that used in ET-QKD and hence are not applicable for IR considered in

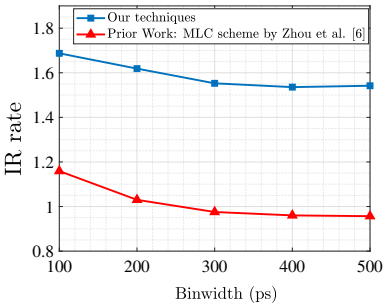

Fig. 1 Improvements in IR rate due to our techniques compared to the MLC scheme of [6]. The red curve utilizes binary LDPC codes and the blue curve utilizes NB-LDPC codes in $\mathbb{GF}(2^5)$. The code length of the LDPC codes used in both curves is 2000. Overall, our techniques result in 40–60% improvement in the IR rates for different values of binwidth. Simulation details about this figure are provided in Fig. 11

this paper. Additionally, the above works focus on channel models such as binary input additive white Gaussian noise (BIAWGN) that do not match the ET-QKD channel [24].

1.1 Contributions

In this paper, we provide techniques to get high IR rates without a large increase in the key generation complexity by optimizing the MLC scheme of [6]. Our techniques involve NB-LDPC code design considering the properties of the ET-QKD channel resulting in higher IR rates compared to conventional LDPC codes. In particular, the contributions of this paper are listed as follows:

1. We provide a flexible protocol for IR in ET-QKD systems called non-binary multi-level coding NB-MLC(a), which is parameterized by an integer $a > 0$. The NB-MLC(a) protocol is a generalization of the MLC protocol of [6]. It splits the raw key symbols into multiple layers with non-binary symbols belonging to $\mathbb{GF}(2^a)$ and utilizes NB-LDPC codes in $\mathbb{GF}(2^a)$ for reconciliation. For $a = 1$, the NB-MLC(a) protocol becomes equivalent to the MLC protocol, and for $a = q$, where q is the number of bits required to represent each raw key symbol, the NB-MLC(a) protocol becomes equivalent to the FNB protocol discussed above. The NB-MLC(a) protocol, thus, offers a natural trade-off between IR rate and complexity depending on the value of a, allowing flexibility in system design. Additionally, we demonstrate that the NB-MLC(a) protocol with a small value of a significantly improves the IR rate without much increase in complexity.
2. The IR rate of the NB-MLC(a) protocol is affected by the NB-LDPC codes used in each layer and the order of decoding and communication among the different layers. In this paper, we provide techniques to optimize these two aspects. In particular, we provide i) a joint code rate and degree distribution optimization (JRDO) framework based on differential evolution [25, 26] to construct NB-LDPC codes for each layer of the NB-MLC(a) protocol and ii) an interleaved decoding and communication (IDC) scheme to decode the different layers of the NB-MLC(a) protocol. The JRDO code design algorithm is tailored to use the ET-QKD channel

information and we demonstrate that it results in a higher IR rate compared to the LDPC codes used in the MLC scheme [6] and that obtained by utilizing degree distributions optimized for conventional channels such as the BIAWGN channel [27]. Additionally, we show that the IDC scheme improves the IR rate compared to the traditional sequential decoding and communication scheme used in [6].

Overall, as demonstrated in Fig. 1, the NB-MLC(a) protocol with a small value of a that utilizes the above proposed techniques results in a significant 40–60% improvement in the IR rate compared to the MLC scheme without much increase in complexity.

1.2 Related work

In this paper, we focus on the entanglement-based ET-QKD protocol and show that it can result in high secret key rates. Along with the above ET-QKD protocol, various other QKD protocols have been proposed for this application that differ in the quantum step of raw key generation compared to the ET-QKD protocol considered in this paper. In [28, 29], twin-field (TF) QKD protocols were proposed for use in practical quantum communication networks. To reduce the experimental complexity and allow free-space realization while maintaining high secret key rates, asynchronous measurement-device independent QKD protocols were proposed in [30, 31]. To mitigate the effect of device imperfections, a phase-matching QKD protocol was proposed in [32]. Quantum digital signature techniques were proposed in [33] to achieve various cryptographic tasks such as integrity, authenticity, and non-repudiation. Along with the above works, various experimental works such as [34–37] demonstrate the feasibility of using QKD and quantum cryptography protocols in real-world applications.

The rest of this paper is organized as follows. In Sect. 2, we provide the preliminaries and the ET-QKD system model. We describe the NB-MLC(a) protocol in Sect. 3. In Sect. 4, we provide the techniques to optimize the NB-MLC(a) protocol that include the JRDO algorithm and the IDC scheme. Finally, we provide simulation results in Sect. 5 to demonstrate the improvements provided by our techniques and conclude the paper in Sect. 6.

2 Preliminaries

In this section, we discuss the general setting for IR in ET-QKD systems, the channel model, relevant performance metrics, and the necessary background about NB-LDPC codes. We then describe our proposed techniques in detail. We use the following notation for the rest of this paper. For a set $\mathcal{S}$, let $|\mathcal{S}|$ denote its cardinality. Let $\lfloor x \rfloor$ and $\lceil x \rceil$ denote the floor and ceil of integer x, respectively. For integers x and y, let $\text{mod}(x, y)$ denote the remainder when x is divided by y. Let $l(\mathbf{B})$ denote the length (in bits) of the sequence of bits $\mathbf{B}$. Let ACK and NACK denote acknowledge and negative acknowledge messages, respectively. For a function $f(x)$, let $f'(x)$ denote the first derivative of $f(x)$. For a vector $\mathbf{v}$ and matrix $\mathbf{m}$, let $\mathbf{v}[k]$ and $\mathbf{m}[k, j]$ denote the kth component of the vector $\mathbf{v}$ and the element at the kth row and jth column of $\mathbf{m}$, respectively. For quantities $\mathrm{C}_i, \mathrm{C}_{i+1}, \ldots, \mathrm{C}_j$ (which could be scalars, vectors, sets,

etc.) where $i < j$ are integers, we define the notation $\mathrm{C}_i^j := \{\mathrm{C}_i, \mathrm{C}_{i+1}, \ldots, \mathrm{C}_j\}$. Additionally, $\mathrm{C}_i^j = \mathrm{D}_i^j$ iff $\mathrm{C}_k = \mathrm{D}_k \ \forall i \leq k \leq j$. All logarithms use base 2 in this paper.

2.1 ET-QKD system model

As discussed in Sect. 1, an ET-QKD system consists of the following steps:

1. *Raw key generation:* As shown in Fig. 2, in this step, energy-time entangled photon pairs are first generated by a third party. Alice and Bob receive one photon each out of the pair who then record the arrival times of the received photons. The raw key symbols are derived from the arrival times of the received photons. In this method, the time domain of Alice and Bob (assumed to be synchronized) is divided into non-overlapping frames. Each frame is further divided into 2^q bins of equal size, where q is a positive integer. Thus, each arrival time within a frame can be represented as a symbol in $\mathbb{GF}(2^q)$ based on the bin number the received photon occupies within each frame. Alice and Bob only retain frames in which they both detect a single photon arrival and discard all other frames. The $\mathbb{GF}(2^q)$ symbols corresponding to non-discarded frames are then divided into blocks of N symbols. Let $\mathbf{X} = \{X_1, \ldots, X_N\}$, $X_i \in \mathbb{GF}(2^q)$ and $\mathbf{Y} = \{Y_1, \ldots, Y_N\}$, $Y_i \in \mathbb{GF}(2^q)$ be the sequences of length N recorded by Alice and Bob, respectively. $\mathbf{X}$ and $\mathbf{Y}$ are the raw keys obtained by Alice and Bob, respectively, at the end of the raw key generation step. Due to imperfections in the raw key generation step (e.g., timing jitters, photon losses, dark counts, etc. [9]), the raw key $\mathbf{Y}$ is a noisy version of $\mathbf{X}$. We assume that the sequences $\mathbf{X}$ and $\mathbf{Y}$ are memoryless and each Y_i is the output of the ET-QKD channel characterized by transition law $P_{Y|X}$ and input X_i.
2. *Information reconciliation (IR):* In this step, Alice and Bob communicate over the public channel which is authenticated but accessible to eavesdropper Eve. Based on the public communication and raw key $\mathbf{X}$, Alice generates a sequence of bits $\mathbf{K}$. Similarly, based on the public communication and $\mathbf{Y}$, Bob generates a sequence of bits $\mathbf{K}'$. The goal of the IR step is to make $\mathbf{K}$ equal to $\mathbf{K}'$ but Eve can have some information about $\mathbf{K}$. The sequences $\mathbf{K}$ and $\mathbf{K}'$ are called the *reconciled keys*.
The IR step involves a *verification* procedure $\texttt{verify-key}(\mathbf{B}, \mathbf{B}')$ that Alice and Bob use to check whether some sequence of bits $\mathbf{B}$ and $\mathbf{B}'$ held by Alice and Bob, respectively, match [38]. Here, $\mathbf{B}$ and $\mathbf{B}'$ are substrings of the reconciled keys $\mathbf{K}$ and $\mathbf{K}'$. Using verification, Alice and Bob ensure that $\mathbf{K}$ and $\mathbf{K}'$ are equal with high probability. In this paper, we use the verification procedure mentioned in [39]. To determine whether $\mathbf{B}$ and $\mathbf{B}'$ are equal, Alice and Bob compare the hashes $\mathtt{h}(\mathbf{B})$ and $\mathtt{h}(\mathbf{B}')$, where $\mathtt{h}()$ is a hash function described in [39]. The verification procedure $\texttt{verify-key}(\mathbf{B}, \mathbf{B}')$ is as follows. Bob first sends $h :=\mathtt{h}(\mathbf{B}')$ to Alice. Alice checks if $\mathtt{h}(\mathbf{B})$ is equal to h. If $\mathtt{h}(\mathbf{B}) = h$, Alice sends an $\texttt{ACK}$ message to Bob. Alice and Bob then consider $\mathbf{B}$ and $\mathbf{B}'$ as verified and use them as part of the reconciled keys. If $\mathtt{h}(\mathbf{B}) \neq h$, Alice sends a $\texttt{NACK}$ message to Bob and they both reject the sequences $\mathbf{B}$ and $\mathbf{B}'$.
For a prime p, let $l_p = \lfloor \log p \rfloor$. The hash length of the hash function $\mathtt{h}()$ and the collision probability $\epsilon()$, i.e., the probability that $\mathtt{h}(\mathbf{B}) =\mathtt{h}(\mathbf{B}')$ for some $\mathbf{B} \neq \mathbf{B}'$

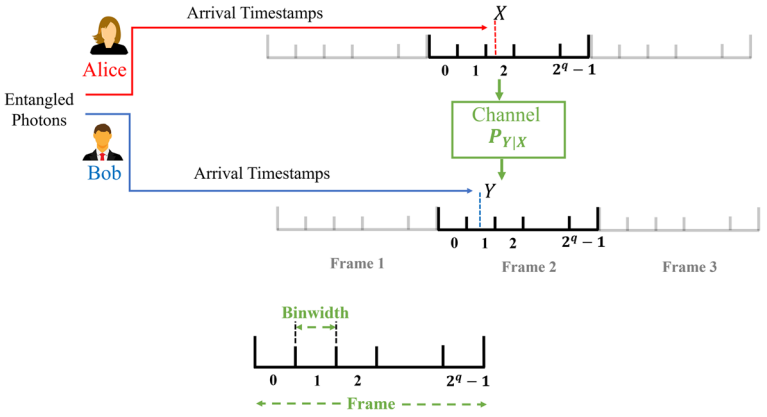

Fig. 2 Raw key generation in the ET-QKD system. The arrival times of photons are discretized to get the raw key symbols. Each frame has 2^q bins and the spacing between frames is called binwidth

are related to p as follows [39]. We have, $l_{\text{ht}} = \lceil \log p \rceil$ bits and

$$\epsilon(l(\mathbf{B})) \leq \frac{\lceil l(\mathbf{B})/l_p \rceil - 1}{p}. \tag{2}$$

The collision probability $\epsilon()$ affects the probability of verification failure ϵ_{ver}, which is the event that Alice and Bob accept reconciled keys $\mathbf{K}$ and $\mathbf{K}'$ that are not the same. The probability of verification failure ϵ_{ver} can be made small by choosing a large p.

We measure the performance of the IR step using the IR rate $\mathcal{R}_{IR}$ described in Eq. (1) where $L_{IR} = l(\mathbf{K}) = l(\mathbf{K}')$. Any information about the reconciled key $\mathbf{K}$ communicated over the public channel during IR (including the hashes during verification) must be included in leak_{IR} and subtracted in the IR rate calculation as per Eq. (1).

3. *Privacy amplification (PA):* This step is applied to the reconciled keys $\mathbf{K}$ and $\mathbf{K}'$ obtained after IR to extract secret keys $\mathbf{S}$ and $\mathbf{S}'$ by Alice and Bob, respectively. Note that if $\mathbf{K} = \mathbf{K}'$, then $\mathbf{S} = \mathbf{S}'$. PA ensures that Eve has no information about $\mathbf{S}$ and that the resulting $\mathbf{S}$ is uniformly distributed given Eve's information. Hence, $\mathbf{S}$ can be safely used as a cryptographic key. The length of $\mathbf{S}$ is determined by the amount of information leaked to Eve during the raw key generation and IR steps. The objective of QKD protocols is to maximize the length of $\mathbf{S}$. In this paper, we focus on the IR step and optimize the IR rate $\mathcal{R}_{IR}$ to achieve the above goal.

Remark 1 The overall secret key rate $\mathcal{R}_{SKR}$ (in bits per photon) of the system can be approximated from the IR rate $\mathcal{R}_{IR}$ as $\mathcal{R}_{SKR} \approx \mathcal{R}_{IR} - \chi_{BE}$ (in bits per photon), where χ_{BE} is Eve's Holevo information [5]. Thus, improving the IR rate $\mathcal{R}_{IR}$ improves the overall secret key rate of the system.

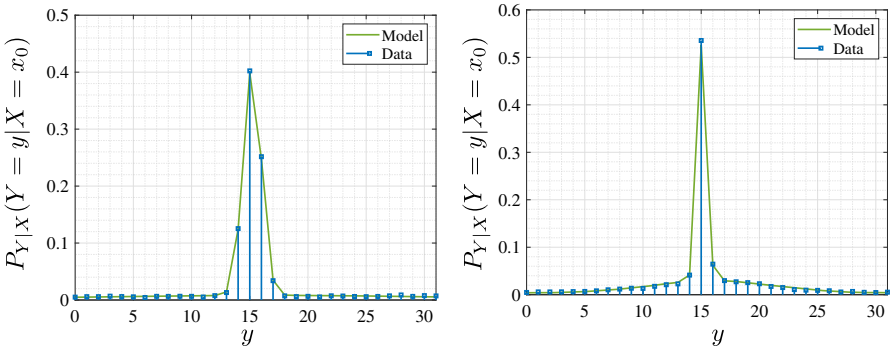

Fig. 3 Empirical transition probabilities obtained from our experimental data and the channel model approximation of Eq. (3). The QKD system has 2^5 bins per frame ($q = 5$). Left panel: Binwidth 100ps, $(\alpha, \sigma_1, \mu_1, \sigma_2, \mu_2, \beta) = (0.013, 1.084, 0.212, 17.175, 1.719, 0.0028)$; Right Panel: Binwidth 700ps, $(\alpha, \sigma_1, \mu_1, \sigma_2, \mu_2, \beta) = (0.052, 0.562, 0.069, 7.286, 0.959, 0.0039)$. Both the plots use $x_0 = 16$

2.2 ET-QKD channel model

As suggested in [24, 40] and also observed from our ET-QKD experiment testbed [9], the ET-QKD channel $P_{Y|X}$ in the raw key generation step is a mixture of a *local* and a *global* channel modeling local and global errors, respectively. Local errors are caused by timing jitters and synchronization errors that result in the two entangled photons falling into different but close enough bins. Global errors are caused due to channel losses and accidental concurrent detection of two non-entangled photons in the same frame. Experimental results show that the local channel is well-fitted by a discretized Gaussian distribution, whereas the global channel is well-fitted by a mixture of a discretized Gaussian and a uniform distribution. Overall, the ET-QKD channel can be approximated using the transition probability

$$P_{Y|X}(Y = y|X = x) = c\left(e^{-\left(\frac{y-x-\mu_1}{\sigma_1}\right)^2} + \alpha e^{-\left(\frac{y-x-\mu_2}{\sigma_2}\right)^2}\right) + \beta,\ x, y \in \mathbb{GF}(2^q), \tag{3}$$

where the parameters α and β, respectively, determine the strengths of the Gaussian component and the uniform component of the global channel in the overall channel transition probability and c is a normalization constant. We observe from our experimental data that μ_1 and μ_2 are both nonzero which makes the ET-QKD channel asymmetric. This asymmetry is due to the misalignment of the center of the bins with the real arrival time of photons [40]. The global component of the channel causes a low SNR in our system resulting in a high operating frame error rate (FER) ($\sim 1 - 10\%$). Finally, note that the distribution $P(X = x)$ is uniform in $\mathbb{GF}(q)$.

Figure 3 provides a comparison of the model in Eq. (3) with that of the empirical transition probabilities obtained from our experimental data. We can see the model closely approximates the data for different choices of q and binwidth. Importantly, the ET-QKD channel is different from conventional channels such as AWGN, BSC, etc.

As such, LDPC codes that have been optimized for these channels are not necessarily the best ones for the ET-QKD channel.

2.3 Non-binary LDPC code preliminaries

A NB-LDPC code over $\mathbb{GF}(2^g)$, where g is a positive integer, is defined by a sparse parity check matrix $\mathbf{H} \in \mathbb{GF}(2^g)^{M\times N}$. The matrix $\mathbf{H}$ has a Tanner graph representation comprising of M check nodes (CNs) and N variable nodes (VNs) corresponding to rows and columns of $\mathbf{H}$. A CN is connected to a VN by an edge if the corresponding entry in $\mathbf{H}$ is nonzero where the edge is additionally labeled by the nonzero entry. The interconnection between VNs and CNs of a code is represented by node-perspective degree distributions $L(x) = \sum_d L_d x^d$ and $P(x) = \sum_d P_d x^d$, where L_d and P_d represent the fraction of VNs and CNs of degree d, respectively. The coding rate R of the code is given by $R = 1 - \frac{L'(1)}{P'(1)}$. The FER performance of the code depends on the degree distributions $L(x)$ and $P(x)$. Degree distribution optimization techniques for LDPC codes based on code thresholds (e.g., [27]) optimize the degree distribution for a fixed code rate R and are not directly applicable to the current ET-QKD problem which needs a joint code rate and FER optimization as we demonstrate in Sect. 2.4. Additionally, the optimized degree distributions are designed for non-QKD channels (e.g., BIAWGN in [27]) and they do not result in large IR rates as we demonstrate in Sect. 5.

In the IR step, we perform NB-LDPC decoding using side information which is known as the Slepian-Wolf (SW) problem [41]. In the SW problem, we try to decode a sequence of symbols $\mathbf{X}^{sw}$ from syndrome $\mathbf{S}^{sw} = \mathbf{H}\mathbf{X}^{sw}$ and side information $\mathbf{Y}^{sw}$, where $\mathbf{H}$ is the parity check matrix of an NB-LDPC code. The decoder is very similar to the sum-product decoder used in conventional decoding of NB-LDPC codes [10, 42] with minor modifications in the way the channel log-likelihood (LLR) messages are initialized and the CN to VN messages. We describe these quantities briefly here and refer the reader to see [41] and references therein for details about SW-LDPC decoding. The channel LLR message for VN n, denoted by $\mathbf{m}_n^{\text{ch}}$, in a SW-LDPC decoder is

$$\mathbf{m}_n^{\text{ch}}[k] = \log \frac{P(X = 0|Y = \mathbf{Y}^{sw}[n])}{P(X = k|Y = \mathbf{Y}^{sw}[n])},\ k = 0, 1, \ldots, 2^g - 1. \tag{4}$$

Let $\ominus$ and $\oslash$ be the usual operators for subtraction and division, respectively, in $\mathbb{GF}(2^g)$. At iteration ℓ of the sum-product decoder, the message $\mathbf{m}_{m,n}^{(\ell)}$ from CN m to VN n is given by [41]

$$\mathbf{m}_{m,n}^{(\ell)} = \mathcal{A}_{\bar{s}[m]}\widetilde{\mathcal{F}}^{-1}\left(\prod_{n'\in\mathcal{N}(m)\setminus n} \widetilde{\mathcal{F}}\left(W_{\bar{g}[n',m]}\mathbf{m}_{n',m}^{(\ell-1)}\right)\right), \tag{5}$$

where, $\bar{s}_m = \ominus\mathbf{S}^{sw}[m] \oslash \mathbf{H}[n, m]$, $\bar{g}[n', m] = \ominus\mathbf{H}[n', m] \oslash \mathbf{H}[n, m]$, $\mathcal{N}(m)$ is the set of variable nodes in row m of $\mathbf{H}$, and $\mathcal{F}$ and $\mathcal{F}^{-1}$ represent an Fourier-like transform and its inverse as defined in [41]. Additionally, $\mathcal{A}_{\bar{s}[m]}$ and $W_{\bar{g}[n',m]}$ are matrices whose

definitions can be found in [41]. Note that the CN to VN message in the channel coding version of the sum-product LDPC decoder is given by [41]

$$\mathbf{m}_{m,n}^{(\ell)} = \widetilde{\mathcal{F}}^{-1}\left(\prod_{n'\in\mathcal{N}(m)\setminus n} \widetilde{\mathcal{F}}\left(W_{\bar{g}[n',m]}\mathbf{m}_{n',m}^{(\ell-1)}\right)\right).$$

The only difference between the CN to VN message in the SW-LDPC decoder in Eqn (5) and the channel coding version shown above is the matrix $\mathcal{A}_{\bar{s}[m]}$ (in the channel coding version, the matrix $\mathcal{A}_{\bar{s}[m]}$ is the identity matrix). As such, the decoding complexity of the SW-LDPC decoder is the same as the channel coding version of the sum-product decoder and is given by $O(g \log g)$ [42].

Throughout this paper, for all non-binary parity check matrices $\mathbf{H} \in \mathbb{GF}(2^g)^{M\times N}$, each nonzero entry in $\mathbf{H}$ is chosen uniformly at random from the set of nonzero element of $\mathbb{GF}(2^g)$. For a given coding rate R and VN node degree distribution $L(x)$, the CN node degree distribution $P(x)$ is chosen to be a two-element distribution [10] such that it results in rate R. These types of CN degree distributions are called *concentrated* [10] and we show in Sect. 5 that they result in high IR rates. Finally, in the SW-LDPC sum-product decoding used in this paper, the maximum number of decoding iterations is set to Γ.

2.4 Example: fully non-binary (FNB) protocol for IR

In this subsection, we explain the FNB protocol for IR as a demonstrative example. Recall that $\mathbf{X} \in \mathbb{GF}(2^q)^N$ and $\mathbf{Y} \in \mathbb{GF}(2^q)^N$ are the raw keys recorded by Alice and Bob, respectively. In the FNB protocol, the raw keys are directly encoded/decoded using NB-LDPC codes in $\mathbb{GF}(2^q)$. The protocol is as follows. Alice sends Bob $\mathbf{S} = \mathbf{HX}$ over the public channel where $\mathbf{H} \in \mathbb{GF}(2^q)^{M\times N}$ is the parity check matrix of a NB-LDPC code. Bob decodes $\mathbf{X}$ using the received $\mathbf{S}$ and side information $\mathbf{Y}$ following SW-LDPC decoding as explained in Sect. 2.3 to get the decoding output $\widehat{\mathbf{X}}$. After decoding, Alice and Bob proceed with the verification procedure `verify-key`$(\mathbf{X}, \widehat{\mathbf{X}})$. If the verification is successful, Alice and Bob use $\mathbf{K} = \mathbf{X}$ and $\mathbf{K}' = \widehat{\mathbf{X}}$ as the reconciled keys.

The goal of the NB-LDPC code is to make the decoding output $\widehat{\mathbf{X}}$ equal to $\mathbf{X}$ with high probability. Following Eqn (1), the IR rate $\mathcal{R}_{IR}^{FNB}$ for the FNB protocol can be calculated as follows. Let E be the frame error rate during decoding. Then, we have $\mathbb{E}[L_{IR}] = q(1-E)N$. Similarly, we have $\mathbb{E}[\text{leak}_{IR}] = q(1-E)M + l_{\text{ht}}(1-E)$ (recall that l_{ht} is the length of the hash function used during verification). Thus,

$$\mathcal{R}_{IR}^{FNB} = q(1-E)\frac{N-M}{N} - (1-E)\frac{l_{\text{ht}}}{N}. \tag{6}$$

Note that in the above equation, $\frac{N-M}{M}$ is the code rate of $\mathbf{H}$. A unique property of the ET-QKD problem is that the IR rate of the system, as seen in Eq. (6), is closely dependent on both the code rate and the FER. Figure 4 shows the FER and IR rates

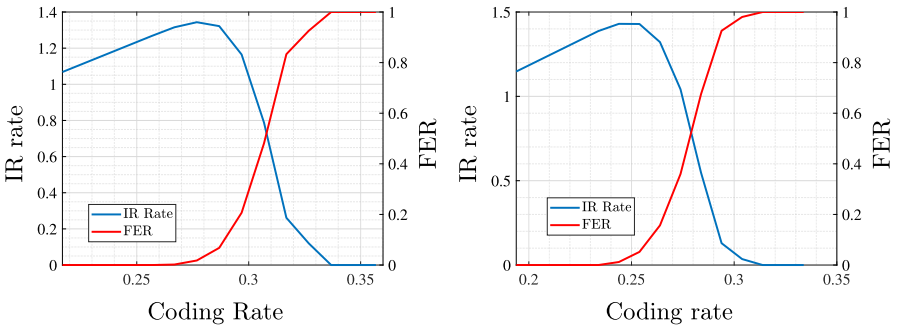

Fig. 4 IR rate and FER versus coding rate for the FNB protocol. Left panel: $q = 5$; Right panel: $q = 6$. Both plots use an ET-QKD system with binwidth 300ps. In the plots, the ET-QKD channel is fixed and the coding rate is varied. IR rate is calculated using Eq. (6). All plots use a VN degree regular LDPC code with a constant VN degree of 3 constructed using the PEG algorithm [43]

obtained by a VN degree regular LDPC code constructed using the PEG algorithm [43] for different values of code rate. From this graph, we see that increasing the code rate can improve the IR rate even at the cost of higher FER. Additionally, as mentioned in Sect. 2.2, the global component of the ET-QKD channel leads to a low SNR in the system. Due to this property, we observe a high FER in the system that results in the maximum IR rate to occur at a relatively large value of FER ($\sim 1 - 10\%$). While the conventional code design approach is to minimize the FER to a very small value for a given code rate, in this paper, we jointly optimize both the code rate and the FER to maximize the IR rate in Section 4.1. In the next section, we explain the NB-MLC(a) protocol for IR.

3 Non-binary multi-level coding protocol

In the FNB protocol described in Sect. 2.4, the symbol size is equal to the number of bins 2^q and the protocol utilizes NB-LDPC codes in $\mathbb{GF}(2^q)$. In this section, we propose the NB-MLC(a) protocol where the symbol size can be varied through an integer parameter a, $1 \leq a \leq q$. The NB-MLC(a) protocol offers a trade-off between IR rate $\mathcal{R}_{IR}$ and decoding complexity through the parameter a allowing flexibility in system design. Let b and r be integers such that $q = ab + r$, where $b = \lfloor \frac{q}{a} \rfloor$ and $r = \mod(q, a)$. Also, let $T = \lceil \frac{q}{a} \rceil$. Let $\alpha_i = a$, $1 \leq i \leq b$ and $\alpha_{b+1} = r$. Thus, $q = \sum_{i=1}^{T} \alpha_i$. Let $u : \mathbb{GF}(2^q) \rightarrow \mathbb{GF}(2^{\alpha_1}) \times \mathbb{GF}(2^{\alpha_2}) \ldots \times \mathbb{GF}(2^{\alpha_T})$ be an bijective mapping such that for $x \in \mathbb{GF}(2^q)$, $u(x) = (u_1(x), u_2(x), \ldots, u_T(x))$ where $u_i(x) \in \mathbb{GF}(2^{\alpha_i})$, $1 \leq i \leq T$. At the beginning of the NB-MLC(a) protocol, Alice and Bob initialize their reconciled keys $\mathbf{K}$ and $\mathbf{K}'$ to empty bit sequences.

Each symbol X in $\mathbf{X}$ received by Alice is an element of $\mathbb{GF}(2^q)$. Using the injective mapping $u()$, Alice maps X into T symbols $(X_1, X_2, \ldots X_T)$, where $X_i = u_i(X)$, $1 \leq i \leq T$. Using the above conversion, Alice splits the sequence $\mathbf{X}$ into T layers $(\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_T)$, where $\mathbf{X}_i \in \mathbb{GF}(2^{\alpha_i})^N$, $1 \leq i \leq T$. For each layer i, Alice uses a NB-LDPC code $\mathbf{H}_i \in \mathbb{GF}(2^{\alpha_i})^{m_i \times N}$, $1 \leq i \leq T$. Alice then generates

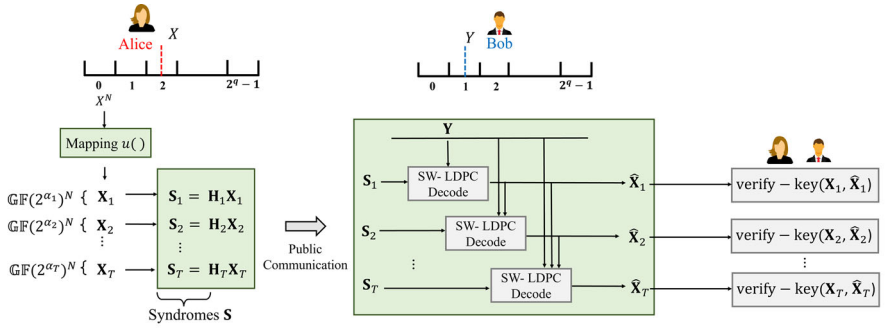

Fig. 5 Illustration of the NB-MLC(a) protocol

a message $\mathbf{S} = \{\mathbf{S}_1, \ldots, \mathbf{S}_T\}$ where $\mathbf{S}_i = \mathbf{H}_i\mathbf{X}_i$, $1 \le i \le T$, are the corresponding syndromes for each layer. Alice then sends $\mathbf{S}$ to Bob.

Bob sequentially decodes every layer $\mathbf{X}_i$, $1 \le i \le T$, using $\mathbf{S}$, $\mathbf{Y}$ and $\mathbf{H}_i$, $i \le i \le T$. Let $\widehat{\mathbf{X}}_1^{i-1} := \{\widehat{\mathbf{X}}_1, \widehat{\mathbf{X}}_2, \ldots, \widehat{\mathbf{X}}_{i-1}\}$ be the decoding output of layers $1, 2, \ldots i-1$. Since $\mathbf{X}_i$, $1 \le i \le T$ are correlated, Bob uses the decoding output $\widehat{\mathbf{X}}_i$ of layer i for decoding the subsequent layers $i+1, \ldots, T$. As such, Bob decodes layer i using the syndrome $\mathbf{S}_i$ and side information $\{\mathbf{Y}, \widehat{\mathbf{X}}_1^{i-1}\}$ to get $\widehat{\mathbf{X}}_i$ following SW-LDPC decoding described in Sect. 2.3. After decoding layer i to get $\widehat{\mathbf{X}}_i$, Alice and Bob perform a verification procedure verify-key($\mathbf{X}_i, \widehat{\mathbf{X}}_i$). For each layer i, if verification procedure verify-key($\mathbf{X}_i, \widehat{\mathbf{X}}_i$) is successful, Alice and Bob append $\mathbf{X}_i$ and $\widehat{\mathbf{X}}_i$ to the reconciled keys $\mathbf{K}$ and $\mathbf{K}'$, respectively. An illustration of the NB-MLC(a) protocol is provided in Fig. 5.

In the SW-LDPC decoding procedure carried out by Bob above, the ith layer has an equivalent channel with input $\mathbf{X}_i$, output $\{\mathbf{Y}, \mathbf{X}_1^{i-1}\}$ and channel transition law $\gamma_{\text{seq}}^i := P(Y = y, X_1^{i-1} = x_1^{i-1} | X_i = x_i)$. The transition law γ_{seq}^i is used in the channel LLR initialization of the SW-LDPC decoder as per Eq. (4) and can be derived from the ET-QKD channel $P_{Y|X}(Y = y|X = x)$ as follows:

$$\gamma_{\text{seq}}^i := P(Y = y, X_1^{i-1} = x_1^{i-1} | X_i = x_i) = \frac{\sum_{x \in A_1(x_1, x_2, \ldots, x_i)} P_{Y|X}(Y = y|X = x)}{|A_2(x_i)|}, \tag{7}$$

where, $A_1(x_1, x_2, \ldots, x_i) = \{x \in \mathbb{GF}(2^q) \mid u_j(x) = x_j, 1 \le j \le i\}$ and $A_2(x_i) = \{x \in \mathbb{GF}(2^q) \mid u_i(x) = x_i\}$. Additionally, note that $P(X_i = x_i)$ is uniform in $\mathbb{GF}(2^{\alpha_i})$.

We now calculate the IR rate $\mathcal{R}_{IR}^{\text{NB-MLC}(a)}$ for the NB-MLC(a) protocol. Let E_i be the frame error rate encountered while decoding the ith layer using the decoded outputs of the previous layers. We discuss the effect of error propagation on E_i and ways to mitigate it in Sect. 3.1. For the IR rate calculation in Eq. (1), we have $\mathbf{E}[L_{IR}] = \sum_{i=i}^{T} \alpha_i(1 - E_i)N$ and $\mathbf{E}[\text{leak}_{IR}] = \sum_{i=i}^{T} \alpha_i(1 - E_i)m_i + \sum_{i=i}^{T}(1 - E_i)l_{\text{ht}}$. Thus,[1]

[1] Note that the IR rate calculation in Eqn (8) takes into account the information leakage due to the verification procedure verify-key($\mathbf{X}_i, \widehat{\mathbf{X}}_i$).

$$\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)} = \sum_{i=i}^{T} \alpha_i (1 - E_i) \frac{N - m_i}{N} - \sum_{i=1}^{T} (1 - E_i) \frac{l_{\text{ht}}}{N}. \tag{8}$$

In the above equation, the IR rate for the $\texttt{NB-MLC}(a)$ protocol is the sum of the IR rates of each layer which is a result of using the verification procedure $\texttt{verify-key}(\mathbf{X}_i, \widehat{\mathbf{X}}_i)$ on each layer individually. Due to using the verification procedure on each layer, a decoding success in one layer can contribute to the overall reconciled key $\mathbf{K}$ even if other layers have decoding failures, thus helping to improve the IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$. Additionally, we conjecture that the IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$ is non-monotonic in a due to the following reasons: i) Increasing the value of a makes the $\texttt{NB-MLC}(a)$ protocol use NB-LDPC codes from a larger Galois field. These are typically stronger codes with better FER performance resulting in better IR rates per layer; ii) However, using a smaller a results in more layers. Thus, due to the sum IR rate property described above, a higher number of layers as a result of smaller a positively affects the overall IR rate. Due to the above effects in i) and ii), the overall IR rate is non-monotonic. We demonstrate the non-monotonic behavior of $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$ in Sect. 5. Note that the decoding complexity of the $\texttt{NB-MLC}(a)$ protocol is the sum of the decoding complexities of each of the layers in the protocol. As such, the complexity can be written as $O(\sum_{i=1}^{T} \alpha_i \log \alpha_i)$ which can be shown to monotonically increase with a. Finally, note that in the $\texttt{NB-MLC}(a)$ protocol described above, setting $a = 1$ gives us the binary MLC scheme of [6] and $a = q$ provides the FNB protocol described in Sect. 2.4.

The $\texttt{NB-MLC}(a)$ protocol involves the verification of each layer separately. As such, the probability of verification failure $\epsilon_{\text{ver}}^{\texttt{NB-MLC}(a)}$ of the $\texttt{NB-MLC}(a)$ protocol can be calculated as

$$\epsilon_{\text{ver}}^{\texttt{NB-MLC}(a)} \leq (1 - (1 - \epsilon(a))^T)$$

which is the probability of at least one collision in the verification of all layers, where an upper bound on the function $\epsilon()$ is provided in Eq. (2). The value of $\epsilon_{\text{ver}}^{\texttt{NB-MLC}(a)}$ can forced to be small by choosing a large prime p. Next, we discuss the effect of error propagation in the $\texttt{NB-MLC}(a)$ protocol and how it can be eliminated using interactive communication between Alice and Bob.

3.1 Interactive communication to mitigate error propagation

In the $\texttt{NB-MLC}(a)$ protocol described above, the decoding output $\widehat{\mathbf{X}}_i$ of layer i is used in the decoding of the subsequent layers. This process results in error propagation where a decoding error in $\widehat{\mathbf{X}}_i$ results in decoding errors in the subsequent layers, increasing the FERs $E_{i+1}, \ldots, E_T$ and decreasing the overall IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$. However, the effect of error propagation can be eliminated by using interactive communication (IC) between Alice and Bob [6]. In the interactive communication protocol, after decoding layer i, if the verification procedure $\texttt{verify-key}(\mathbf{X}_i, \widehat{\mathbf{X}}_i)$ fails, then Alice directly sends $\mathbf{X}_i$ to Bob which Bob uses to decode the subsequent layers instead of $\widehat{\mathbf{X}}_i$. Since

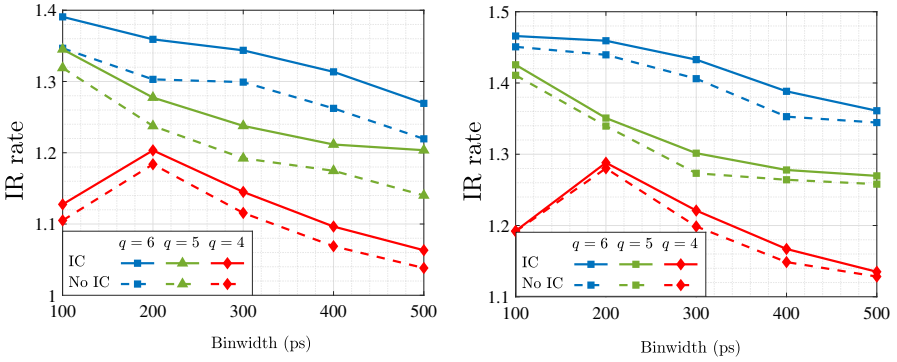

Fig. 6 Comparison of the IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$ with and without interactive communication (IC) for different values of q. Left panel: $a = 1$; Right panel: $a = 2$. All plots use a VN degree regular LDPC code with a constant VN degree 3 constructed using the PEG algorithm [43]. Plots corresponding to IC do not have error propagation while plots corresponding to No IC have error propagation during decoding the layers of the $\texttt{NB-MLC}(a)$ protocol. The channel $P_{Y|X}$ for different binwidths is derived empirically from our experimental data

now Bob uses the true $\mathbf{X}_i$ instead of $\widehat{\mathbf{X}}_i$ for decoding the subsequent layers, it gets a more accurate channel LLR initialization in Eq. (4), resulting in improved FERs $E_{i+1}, \ldots, E_T$ and overall IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$. Note that when decoding fails for layer i, since the corresponding $\mathbf{X}_i$ and $\widehat{\mathbf{X}}_i$ are not added to the reconciled keys $\mathbf{K}$ and $\mathbf{K}'$, revealing $\mathbf{X}_i$ does not add anything to leak_{IR}. Hence, the IR rate for the $\texttt{NB-MLC}(a)$ protocol with interactive communication is still given by Eq. (8) (where the FERs E_i, $1 \leq i \leq T$, are calculated considering the interactive communication protocol described above). The average communication cost due to interactive communication $\texttt{CommCost-IC}(a)$ is given by $\texttt{CommCost-IC}(a) = \sum_{i=1}^{T-1} \alpha_i E_i N$. Note that the FERs E_i encountered at the point of maximum IR rate are typically less than 10% and hence the additional communication cost due to interactive communication is small compared to the communication cost involved in sending the syndromes $\mathbf{S}$. Figure 6 demonstrates the improvement in IR rates for different values of q when the $\texttt{NB-MLC}(a)$ protocol utilizes interactive communication to prevent error propagation.

We call the decoding and the interactive communication protocol mentioned in this section as *sequential decoding and communication (SDC)* due to its sequential nature. Next, we discuss the design choices present in the $\texttt{NB-MLC}(a)$ protocol which can be optimized to result in high IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$.

3.2 Design choices in the NB-MLC(a) protocol

For a given a, the IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$ provided in Eq. (8) depends on three key design choices (marked in green in Fig. 5):

1. NB-LDPC code design which involves the NB parity check matrix $\mathbf{H}_i$ and the code rate $R_i = \frac{N-m_i}{N}$ used in each layer i. The parity check matrix $\mathbf{H}_i$ and the rate R_i affect the FER E_i and hence the IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$. In Sect. 4.1, we provide the

JRDO algorithm to jointly design $\mathbf{H}_i$ and R_i for each layer i of the NB-MLC(a) protocol.

2. The order of operations of decoding and interactive communication of the different layers. In the NB-MLC(a) protocol described above, the order of decoding operations and communication is sequential in the sense that Bob first completes the decoding of layer i and then performs interactive communication before proceeding to decode the subsequent layers. To further improve the IR rate under interactive communication, in Sect. 4.2, we provide an interleaved decoding and communication (IDC) protocol where Bob starts decoding another layer before completing the decoding of the existing layer.
3. The mapping $u(x) = (u_1(x), u_2(x), \ldots, u_T(x))$ used to map the raw key symbols $\mathbf{X}$ into symbols of different layers in the NB-MLC(a) protocol. The mapping function $u(x)$ affects the channel transition probability γ^i_{seq} of each layer provided in Eq. (7) thus affecting the frame error rate E_i and hence the IR rate of layer i. In Sect. 4.3, we show that binary mapping is a good choice of mapping for the ET-QKD channels we have encountered in our testbed and it results in high IR rates.

4 Optimizing the NB-MLC(a) protocol

In this section, we provide the techniques to optimize the NB-MLC(a) protocol. We start with providing the JRDO algorithm based on differential evolution to jointly optimize the code rate and degree distribution of the NB-LDPC codes to be used in the NB-MLC(a) protocol.

4.1 Joint rate and degree distribution optimization (JRDO)

In this section, we describe the algorithm to design parity check matrices $\mathbf{H}_i$ and coding rate R_i, $1 \leq i \leq T$ for use in the ith layer of the NB-MLC(a) protocol that has a channel transition probability γ^i_{seq} provided in Eq. (7). The mapping that determines the channel transition probability γ^i_{seq} is $u()$. Note that the construction method is the same for all layers; hence, we drop index i. As mentioned in Sect. 2.3, the FER performance of the code (and hence the IR rate $\mathcal{R}_{IR}^{\text{NB-MLC}(a)}$) depends on the VN node degree distribution $L(x)$ and coding rate R of $\mathbf{H}$. In this section, we construct $\mathbf{H}$ using the PEG algorithm [43] with VN node degree distribution $L(x)$, code length N, and coding rate R that are optimized by the JRDO framework. We call such parity check matrices $\mathbf{H}^{\text{PEG}}(L(x), R)$.

The JRDO algorithm utilizes differential evolution (DE) [25, 26] to find $L(x)$ and R. DE is a popular and effective population-based evolutionary algorithm that can be used for the maximization (or minimization) of any function $f()$. The algorithm iteratively improves a candidate solution (that maximizes $f()$) using an evolutionary process and can explore large design spaces with low complexity. Note that other optimization algorithms such as genetic algorithms [44], evolution strategies [45, 46], and simulated annealing [47] have also been used in many applications for the

minimization of the function $f()$. However, DE offers parallelizability to cope with computation-intensive functions $f()$, is easy to use with few hyperparameters, and has good convergence properties [25]. Hence, DE has been extensively used in coding theory literature to design good irregular LDPC codes for the erasure channel [48], AWGN channel [27], Rayleigh fading channel [49], etc. In these works, the goal is to design degree distributions that have low FER. This goal is achieved by using DE where the function $f()$ is generally set to a low complexity predictor of the FER performance of the code such as the threshold obtained by density evolution [27]. However, the goal for us in this paper is to maximize the IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$ and not merely to minimize the FER. Additionally, the techniques for optimizing the degree distributions using code thresholds work for a fixed code rate and we have not found any work, relevant for this setting, that jointly optimizes the code rate along with maximizing the threshold. This observation is primarily because, in prior work, the performance of the system was not a direct function of the threshold and the code rate. However, in our case, the performance of the system in terms of the IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$ depends directly on the code rate R and degree distribution and hence must be optimized jointly. In the JRDO algorithm, we perform this joint optimization of the code rate and the degree distribution by maximizing the function $f_{\text{JRDO}}(L(x), R)$ described as

$$f_{\text{JRDO}}(L(x), R) = (1 - E)R, \tag{9}$$

where E is the FER obtained by the parity check matrix $\mathbf{H}^{\text{PEG}}(L(x), R)$ on a channel with transition probability γ_{seq}. The function $f_{\text{JRDO}}(L(x), R)$ is proportional to the IR rate (without the verification cost penalty[2]) of the corresponding layer of the $\texttt{NB-MLC}(a)$ protocol whose parity check matrix is getting designed. Note that to be able to optimize $f_{\text{JRDO}}(L(x), R)$ feasibly using DE, the cost of computing $f_{\text{JRDO}}(L(x), R)$ must be low (since the DE algorithm evaluates $f_{\text{JRDO}}(L(x), R)$ a certain fixed number of times in every iteration). However, since the FER E of the code at the point of maximum in IR rate is high ($\sim 1 - -10\%$), $f_{\text{JRDO}}(L(x), R)$ can again be easily estimated using MC simulations with a small number of MC experiments (e.g., 200–300). The overall JRDO algorithm is provided in Algorithm 1.

The algorithm starts with initializing a population Π of degree distribution and rate pairs of size N_p. The first entry $L_1(x)$ in the population is initialized to a regular distribution with VN degree d_v (line 3) and the rate R_1 is such that it results in the maximum value of $f_{\text{JRDO}}(L_1, R_1)$ (line 4), where $\mathcal{R}_{search} = \{R_{\max}, R_{max} - R_{step}, R_{max} - 2R_{step}, \ldots, R_{\min}\}$. The remaining entries of the population are initialized randomly as shown in lines 5–7. Note that the rates are initialized from a small interval around R_1 (e.g., $\Delta_1 = 0.1$) to ensure that the algorithm starts with good enough rates. Now, at every iteration of the JRDO algorithm, each population entry undergoes mutation and cross over (lines 9–12) to result in pairs (L_j^c, R_j^c), $1 \leq j \leq N_p$, where the procedures `DiffMutation()` and `CrossOver()` have conventional meanings as per [25]. Each population entry (L_j, R_j) is then replaced with the corresponding (L_j^c, R_j^c) if the function evaluation $f_{\text{JRDO}}(L_j^c, R_j^c) > f_{\text{JRDO}}(L_j, R_j)$. After the completion of the

[2] We do not subtract the information leakage due to verification in Eq. (9) to allow the design to be independent of the chosen verification parameters.

Algorithm 1 JRDO: Joint Rate and degree Distribution Optimization

1: **Inputs:**, $N_p, d_v, d_v^{\max}, R_{max}, R_{step}, R_{min}, \Delta_1, \Delta_2$
2: Initialize population $\Pi = \{(L_1, R_1), \ldots, (L_{N_p}, R_{N_p})\}$:
3: $L_1(x) =$ regular distribution with VN degree d_v
4: $R_1 = \underset{R \in \mathcal{R}_{search}}{\text{argmax}} \; f_{\text{JRDO}}(L_1, R)$
5: **for** $j = 1 : N_p$ **do**
6: $L_j =$ random degree distribution with no degree 1 VNs and maximum VN degree $d_v^{\max}$
7: $R_j =$ random rate in the range $[R_1 - \Delta_1, R_1 + \Delta_1]$
8: **for** max number of iterations **do**
9: **for** $j = 1 : N_p$ **do**
10: $(L_j^m, R_j^m) =$`DiffMutation`(j, Π)
11: $(L_j^c, R_j^c) =$ `CrossOver`$\big((L_j^m, R_j^m), (L_j, R_j)\big)$
12: Evaluate $f_{\text{JRDO}}(L_j^c, R_j^c)$ using Monte-Carlo simulations
13: **for** $j = 1 : N_p$ **do**
14: **if** $f_{\text{JRDO}}(L_j^c, R_j^c) > f_{\text{JRDO}}(L_j, R_j)$ **then**
15: Update population: $(L_j, R_j) \leftarrow (L_j^c, R_j^c)$
16: $(L^f, R^f) \leftarrow$ entry in Π with largest $f_{\text{JRDO}}()$
17: $\mathcal{R}_{search}^f = \{R^f - \Delta_2, R^f - \Delta_2 + R_{step}, R^f - \Delta_2 + 2R_{step}, \ldots, R^f + \Delta_2\}$
18: $R^o = \underset{R \in \mathcal{R}_{search}^f}{\text{argmax}} \; f_{\text{JRDO}}(L^f, R)$
19: **Output:** (L^f, R^o)

maximum number of iterations of differential evolution, we perform a final rate search (lines 16–18) around the population entry (L^f, R^f) to allow for further improvements in the function value. Finally, the algorithm outputs (L^f, R^o). We demonstrate the improvements in the IR rate $\mathcal{R}_{IR}^{\text{NB-MLC}(a)}$ due to the JRDO algorithm in Sect. 5. In the next subsection, we provide the interleaved decoding and communication (IDC) protocol to further improve the IR rate under interactive communication.

4.2 Interleaved decoding and communication

In the NB-MLC(a) protocol described in Sect. 3 using interactive communication, the order of operations followed by Alice and Bob is the following: Starting from the first layer, (i) Bob decodes layer i to get $\widehat{\mathbf{X}}_i$; (ii) Alice and Bob perform the verification procedure `verify-key`$(\mathbf{X}_i, \widehat{\mathbf{X}}_i)$ (iii) Alice sends $\mathbf{X}_i$ to Bob if the verification procedure fails; (iv) Bob decodes layer $i+1$. The process is then continued for all layers. We call the above *sequential decoding and communication* (SDC) since the order of operations is sequential where Bob completes the decoding of layer i and performs interactive communication to get the correct $\mathbf{X}_i$ before starting to decode the next layer. In this case, sending the correct $\mathbf{X}_i$ to Bob via interactive communication in the event of a decoding failure of layer i helps Bob get more reliable channel LLR initialization (Eq. 4) for decoding layers $i+1, \ldots, T$ using the channels $\gamma_{\text{seq}}^{i+1}, \ldots, \gamma_{\text{seq}}^{T}$ mentioned in Eq. (7). More reliable channel LLR initialization improves the FERs $E_{i+1}, \ldots, E_T$ of layers $i+1, \ldots, T$, thus, improving the overall IR rate $\mathcal{R}_{IR}^{\text{NB-MLC}(a)}$.

Now, consider the following transition probability:

$$\gamma_{\text{int}}^{i} := P(Y = y, X_1^{i-1} = x_1^{i-1}, X_{i+1}^{T} = x_{i+1}^{T} | X_i = x_i) = \frac{\sum_{x \in A_1'(x_1, x_2, \ldots, x_T)} P_{Y|X}(Y = y | X = x)}{|A_2(x_i)|}, \tag{10}$$

where, $A_1'(x_1, x_2, \ldots, x_T) = \{x \in \mathbb{GF}(2^q) \mid u_j(x) = x_j, 1 \leq j \leq T\}$ and $A_2(x_i) = \{x \in \mathbb{GF}(2^q) \mid u_i(x) = x_i\}$. The above transition probability can provide a more accurate channel LLR for $\mathbf{X}_i$ compared to the transition probability γ_{seq}^{i} mentioned in Eq. (7), provided Bob has reliable values of $\{\mathbf{X}_1, \ldots, \mathbf{X}_{i-1}, \mathbf{X}_{i+1}, \ldots, \mathbf{X}_T\}$. Thus, similar to the SDC protocol, the IR rate can also be improved if Bob can utilize the correct $\mathbf{X}_{i+1}, \mathbf{X}_{i+2}, \ldots, \mathbf{X}_T$ sent by Alice after interactive communication of layers $i+1, i+2, \ldots, T$ for decoding the previous layers $i, i-1, \ldots, 1$. We now describe the IDC protocol that achieves the above goal. The protocol provides an alternative way for Bob and Alice to get the reconciled keys $\mathbf{K}$ and $\mathbf{K}'$ using $\mathbf{X}, \mathbf{Y}, \mathbf{H}_i, 1 \leq i \leq T$, and syndromes $\mathbf{S} = \{\mathbf{S}_1, \ldots, \mathbf{S}_T\}$ compared to the SDC procedure mentioned in Sect. 3. The overall IDC protocol is provided in Algorithm 2 below. Recall that the maximum number of LDPC decoder iterations for each layer is Γ.

Algorithm 2 IDC: Interleaved Decoding and Communication

1: **for** $i = 1 : T$ **do**
2: Bob:
3: Initialize channel LLR $\mathbf{m}_{\text{seq}}^{\text{ch,i}}$ for the LDPC decoder of layer i using $\gamma_{\text{seq}}^{i}, \overline{\mathbf{X}}_1, \overline{\mathbf{X}}_2, \ldots, \overline{\mathbf{X}}_{i-1}, \mathbf{Y}$
4: $\overline{\mathbf{X}}_i$ = LDPC decode using $\mathbf{H}_i, \mathbf{S}_i$, channel LLR $\mathbf{m}_{\text{seq}}^{\text{ch,i}}$ for Γ_1 iterations
5: **for** $i = T : 1$ **do**
6: Bob:
7: Update channel LLR for the LDPC decoder of layer i to $\mathbf{m}_{\text{int}}^{\text{ch,i}}$ using $\gamma_{\text{int}}^{i}, \overline{\mathbf{X}}_1, \ldots, \overline{\mathbf{X}}_{i-1}, \widehat{\mathbf{X}}_{i+1}, \widehat{\mathbf{X}}_{i+2}, \ldots, \widehat{\mathbf{X}}_T, \mathbf{Y}$
8: $\widehat{\mathbf{X}}_i$ = continue LDPC decoding of step 4 using updated channel LLR $\mathbf{m}_{\text{int}}^{\text{ch,i}}$ for remaining $\Gamma - \Gamma_1$ iterations
9: Alice and Bob:
10: `verify-key`$(\mathbf{X}_i, \widehat{\mathbf{X}}_i)$
11: If verification in the previous step is unsuccessful, Alice sends $\mathbf{X}_i$ to Bob and Bob updates $\widehat{\mathbf{X}}_i \leftarrow \mathbf{X}_i$
12: $\mathbf{K}$ = concatenation of $\mathbf{X}_i$ of all layers where verification is sucessfull
13: $\mathbf{K}'$ = concatenation of $\widehat{\mathbf{X}}_i$ of all layers where verification is sucessfull
14: **Output:** Reconciled keys $\mathbf{K}$ and $\mathbf{K}'$ at Alice and Bob, respectively

In the above IDC protocol, Bob first decodes layers $1, 2, \ldots, T$ sequentially (lines 2–4), but performs maximum Γ_1 decoding iterations out of the Γ iterations allowed for each layer. The decoded output of the different layers at the end of Γ_1 iterations is denoted by $\overline{\mathbf{X}}_1, \ldots, \overline{\mathbf{X}}_T$ (line 4). Here, the outputs $\overline{\mathbf{X}}_1, \ldots, \overline{\mathbf{X}}_{i-1}$ are used in the channel LLR initialization[3] of layer i using transition probability γ_{seq}^{i} (line 3). After

[3] This method of channel LLR initialization can result in error propagation during decoding the different layers. However, the interleaved interactive communication ensures that this error propagation does not reduce the IR rates.

performing Γ_1 decoding iterations for every layer, Bob continues the decoding of the different layers in reverse order (lines 6–12) for $\Gamma - \Gamma_1$ more decoding iterations using the updated channel LLRs $\mathbf{m}_{\text{int}}^{\text{ch,i}}$. For each layer i, it finds the updated[4] channel LLR $\mathbf{m}_{\text{int}}^{\text{ch,i}}$ using the transition probability γ_{int}^{i}. To find the updated channel LLR, it uses the decoded outputs $\overline{\mathbf{X}}_1, \ldots, \overline{\mathbf{X}}_{i-1}$ of the layers $1, \ldots, i-1$ (obtained after Γ_1 iterations). It also uses $\widehat{\mathbf{X}}_{i+1}, \ldots, \widehat{\mathbf{X}}_T$ which are the decoded outputs of layers $i+1, \ldots, T$ after continuing the decoding of each layer for $\Gamma - \Gamma_1$ more decoding iterations with the updated channel LLR messages (line 8). Additionally, after obtaining $\widehat{\mathbf{X}}_i$ for each layer, Alice and Bob perform the verification procedure `verify-key`$(\mathbf{X}_i, \widehat{\mathbf{X}}_i)$ (line 10). If the verification is unsuccessful, Alice sends the correct $\mathbf{X}_i$ to Bob and Bob updates $\widehat{\mathbf{X}}_i$ with $\mathbf{X}_i$ (line 11). Thus, the $\widehat{\mathbf{X}}_{i+1}, \ldots, \widehat{\mathbf{X}}_T$ that Bob uses in line 7 to get the updated channel LLRs γ_{int}^{i} are always accurate due to the interactive communication step in line 11.

In the IDC protocol, since Bob uses the transition probability γ_{int}^{i} with the correct $\mathbf{X}_{i+1}, \mathbf{X}_{i+2}, \ldots, \mathbf{X}_T$ (due to interactive communication) to get the updated channel LLRs (in line 7), it can improve the FER of layer i for the same rate or allow a higher rate for the same FER allowing to improve the IR rate. Note that since in the initial decoding phase (lines 2–4), the unverified decoded outputs $\overline{\mathbf{X}}_1, \ldots, \overline{\mathbf{X}}_{i-1}$ are used in the decoding of the next layers as well as in calculating the updated channel LLRs γ_{int}^{i}, there is an effect of error propagation in the system. However, with appropriately chosen code rates R_i, $1 \leq i \leq T$, and the value of Γ_1 (which is the number of decoding iterations in the first phase), the effect of error propagation can be made small and the IDC protocol can improve[5] the IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$. Note that the IR rate $\mathcal{R}_{IR}^{\texttt{NB-MLC}(a)}$ using the IDC protocol is also provided by Eq. (8).

We now describe how to choose appropriate rates R_i^{IDC}, $1 \leq i \leq T$, for use in the different layers of the `NB-MLC`(a) protocol with IDC. Let R_i^o, $1 \leq i \leq T$, be the rates provided by the JRDO algorithm. Note that the rates R_i^o, $1 \leq i \leq T$, in the JRDO algorithm are designed for the SDC case described in Sect. 3. For the case of the IDC protocol, the rates used have to be modified compared to R_i^o, $1 \leq i \leq T$, to result in the largest IR rate[6]. To find the rates, we perform a heuristic search in a small interval around R_i^o, $1 \leq i \leq T$, as provided in Algorithm 3 using the function

$$f_{\text{IDC}}(R_1, \ldots, R_T) = \sum_{i=1}^{T} \alpha_i (1 - E_i) R_i,$$

where E_i is the FER encountered in layer i of the `NB-MLC`(a) protocol with IDC. We demonstrate the improvements in IR rate provided by the IDC protocol in Sect. 5. In the next subsection, we discuss the choice of the mapping function $u()$.

[4] Note that since γ_{seq}^{T} and γ_{int}^{T} are the same transition probabilities, there is no channel LLR update for the last layer. The algorithm directly decodes the last layer with channel LLRs initialized in step 3 for Γ iterations.

[5] We have observed that choosing Γ_1 to be 5-10 iterations less than Γ improves the IR rate compared to the SDC protocol.

[6] Note that we use the same degree distributions in the IDC protocol as those provided by JRDO to reduce the complexity of degree distribution design.

Algorithm 3 Rate Search for IDC

1: **for** $i = 1 : T$ **do**
2: $\mathcal{R}_{search}^{IDC} = \{R_i^o - \Delta_3, R_i^o - \Delta_3 + R_{step}, R_i^o - \Delta_3 + 2R_{step}, \ldots, R_i^o + \Delta_3\}$
3: $R_i^{\text{IDC}} = \underset{R \in \mathcal{R}_{search}^{IDC}}{\text{argmax}} f_{\text{IDC}}(R_1^{\text{IDC}}, \ldots, R_{i-1}^{\text{IDC}}, R, R_{i+1}^o, \ldots, R_T^o)$
4: **Output:** $R_i^{\text{IDC}}, 1 \le i \le T$

4.3 Mappings

In this section, we discuss the choice of the mapping function $u()$ that results in high IR rates. The mapping function $u : \mathbb{GF}(2^q) \to \mathbb{GF}(2^{\alpha_1}) \times \ldots \times \mathbb{GF}(2^{\alpha_T})$ can be equivalently represented as a mapping $u_b : \mathbb{GF}(2^q) \to \mathbb{GF}(2)^q$ that converts a symbol $x \in \mathbb{GF}(2^q)$ into a binary string x^b of length $q = \sum_{i=1}^{T} \alpha_i$. Let $x_1^b || x_2^b || \ldots || x_T^b$ be the partition of the binary string x^b, such that $l(x_i^b) = \alpha_i, 1 \le i \le T$. Then, x_i^b is the binary (base 2) representation of $u_i(x)$. Thus, in the rest of the paper, we directly discuss the choices for the mapping function $u_b(x)$ that leads to a reasonably good IR rate.

Binary mapping is the simplest mapping to consider. It is the function $u_b : \mathbb{GF}(2^q) \to \mathbb{GF}(2)^q$ such that for $x \in \mathbb{GF}(2^q)$, $x = \sum_{i=1}^{q} u_b(x)[i] 2^{i-1}$, where $u_b(x)[i]$ is the ith bit in the bit string $u_b(x)$. Another commonly used mapping is the gray mapping [50] where two successive symbols in $\mathbb{GF}(2^q)$ differ only in 1 bit in their mapped bit strings. Binary and gray mappings are easy to construct. However, it is not clear if they are good choices of mapping to get high IR rates $\mathcal{R}_{IR}^{\text{NB-MLC}(a)}$ for our particular channels. Due to the large search space of mappings, it is computationally expensive to find the optimal mappings using a brute-force search. Here, we use a heuristic approach using the simulated annealing (SA) algorithm [47] to see whether we can improve the IR rates compared to binary or gray mapping.

In the SA algorithm, we start with the binary mapping as the initial choice of u_b and then modify u_b if the modification leads to a better mapping. Specifically, we swap the output of u_b for two distinct input values $x, y \in \mathbb{GF}(2^q)$ if the operation leads to a higher value of the function $f_{\text{SA}}(u_b)$ defined as

$$f_{\text{SA}}(u_b) = \sum_{i=1}^{T} \max_{R_i \in \mathcal{R}_{search}} (\alpha_i (1 - E_i) R_i), \tag{11}$$

where, $\mathcal{R}_{search} = \{R_{\max}, R_{max} - R_{step}, R_{max} - 2R_{step}, \ldots, R_{\min}\}$, α_i and T are constants in the NB-MLC(a) protocol, and E_i is the FER obtained on layer i of the NB-MLC(a) protocol with SDC by a VN degree regular NB-LDPC code constructed using the PEG algorithm [43] with constant VN degree d_v, code length N, and coding rate R_i. The function $f_{\text{SA}}(u_b)$ follows similarly as Eq. (9) and approximates the maximum IR rate $\mathcal{R}_{IR}^{\text{NB-MLC}(a)}$ (see Eqn (8)) achieved by a VN degree regular PEG NB-LDPC code, where the rate R_i is found using a grid search in the set $\mathcal{R}_{search}$. The detailed SA algorithm is provided in Algorithm 4.

Algorithm 4 Simulated Annealing (SA) for Mapping

1: u_b = Binary mapping
2: $f_{\text{SA}}^{\max} = f_{\text{SA}}(u_b)$
3: $u_b^{\max} = u_b$
4: **for** T = max number of SA iterations to 1 **do**
5: u'_b = mapping obtained from u_p by swapping the output for two distinct input values $x, y \in \mathbb{GF}(2^q)$
6: $D_{\text{f}} = f_{\text{SA}}(u'_b) - f_{\text{SA}}(u_b)$
7: **if** $D_{\text{f}} \geq f_{\text{th}}$ **then**
8: $u_b \leftarrow u'_b$
9: **if** $f_{\text{SA}}(u'_b) > f_{\text{SA}}^{\max}$ **then**
10: $u_b^{\max} = u'_b$
11: $f_{\text{SA}}^{\max} = f_{\text{SA}}(u'_b)$
12: **else if** $e^{\frac{D_{\text{f}} - f_{\text{th}}}{T \times \lambda}} > \text{rand}(0, 1)$ **then**
13: $u_b \leftarrow u'_b$
14: **Output:** $u_b^{\max}$

In the SA algorithm, we start with the binary mapping as the current mapping. Then in each iteration, we modify the current mapping u_b to obtain a new mapping[7] u'_b in line 5. Now, if the difference D_{f} in the $f_{SA}()$ values of u'_b and u_b is greater than threshold f_{th} (line 7), we update the current mapping to u'_b. If the difference D_{f} is less than f_{th}, we update the current mapping to u'_b only a fraction of the times based on the condition in line 12 to allow the algorithm to break out of a local maximum.

A comparison of the IR rates obtained by mappings output by the SA search algorithm with that of binary and gray mapping is provided in Fig. 7. From the figure, we first see that in all cases, the binary and gray mappings have very close IR rates. Additionally, we see that the IR rates produced by the SA search algorithm are very close compared to binary and gray mappings. Thus, binary and gray mappings are good choices for mappings for use in the NB-MLC(a) protocol.

5 Simulation results

In this section, we demonstrate the performance of the NB-MLC(a) protocol and the optimization techniques introduced in Sect. 4. We compare the performance with the MLC scheme of [6] as well as with LDPC codes designed for the BIAWGN [27] channel. The various parameters used in our simulations are summarized in Table 1. For the verify-key() procedure, we use the parameters, $p = 2^{32} - 5$, $l_p = \lfloor \log p \rfloor = 31$, $l_{\text{ht}} = \lceil \log p \rceil = 32$ bits. For $\mathcal{R}_{search}$ used in Sect. 4.1, we use $R_{max} = H(X|Y) + 0.1$, where $H()$ denotes the entropy function[8], $R_{min} = 0.01$ and $R_{step} = 0.01$. Similarly, we use $d_v = 3$ in Sect. 4.1. For the JRDO algorithm in Algorithm 1, we optimize degree distribution $L(x) = \sum_{d=2}^{d_v^{\max}} L_d$ with $d_v^{\max} = 5$ and $L_1 = 0$. For the

[7] Duing the algorithm we ensure that at each iteration, we generate a mapping u'_b that has not been encountered before.

[8] The chosen value to R_{max} ensures a high enough starting rate for the search in line 4 of Algorithm 1.

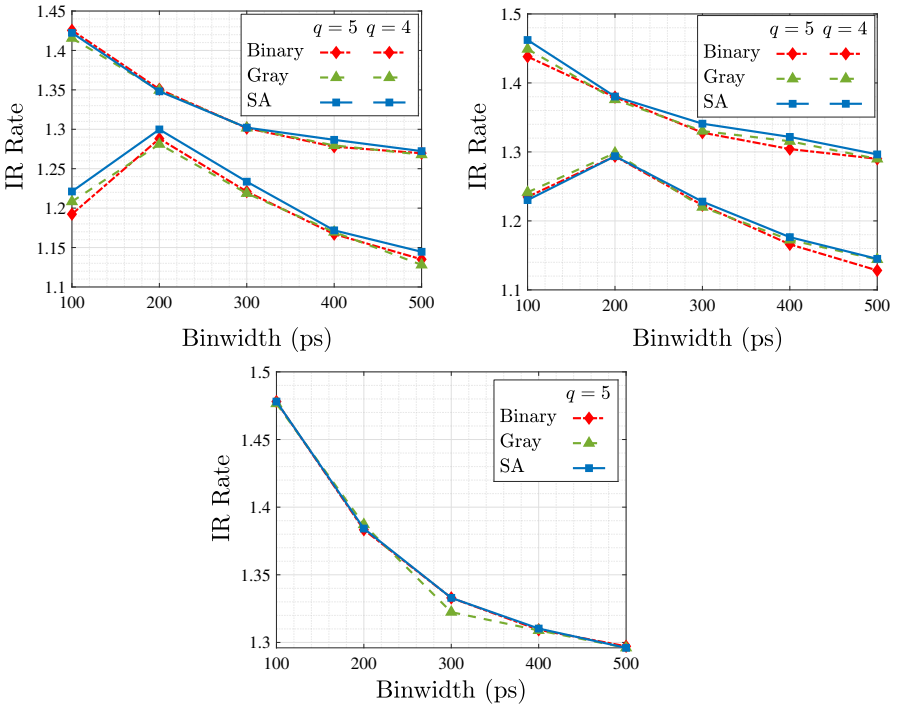

Fig. 7 Comparison of IR rates due to different mapping functions $u_b()$ for the NB-MLC(a) protocol described in Sect. 3 with SDC. Left panel: $a = 2$; Right panel: $a = 3$; Bottom panel: $a = 4$. All plots use a VN degree regular LDPC code with a constant VN degree 3 constructed using the PEG algorithm [43]. The channel $P_{Y|X}$ for different binwidths is derived empirically from our experimental data

rate initialization (line 6 in Algorithm 1), we use $\Delta_1 = 0.1$, Additionally, for $\mathcal{R}^{f}_{search}$ (line 16 of Algorithm 1), we use $\Delta_2 = 0.05$ and $R_{step} = 0.01$. For codes that are not designed using the JRDO algorithm, to calculate the corresponding $\mathcal{R}^{\text{NB-MLC}(a)}_{IR}$, we choose the rates R_i, $1 \leq i \leq T$, that maximize $f_{\text{SA}}()$ defined in Eq. (11). For SW-LDPC decoding, we use the maximum number of decoding iterations $\Gamma = 50$ and $\Gamma_1 = 35$ for the IDC protocol[9] (Algorithm 2). For the rate search in the IDC protocol (Algorithm 3), we use $\Delta_2 = 0.05$ and $R_{step} = 0.01$. Finally, in the ET-QKD system, we use $N = 2000$ in our simulations. For all simulations, we show trends when the channel transition probability $P_{Y|X}$ is provided by the parameterized channel model in Eq. (3) as well as on actual experimental data [9] where $P_{Y|X}$ is derived empirically from the data. For simulations considering the channel transition law $P_{Y|X}$ provided by Eq. (3), we choose a default set of values for parameters $(\alpha, \sigma_1, \mu_1, \sigma_2, \mu_2, \beta)$ that are close to the ones that fit our experimental data for binwidth 100ps (as provided in Fig. 3).

In Fig. 8, we study the effect of the NB-MLC bit size a on the IR rate $\mathcal{R}^{\text{NB-MLC}(a)}_{IR}$. The left panel corresponds to the parameterized channel model in Eq. (3) while the

[9] We found from our simulations that among $\{35, 40, 45\}$, $\Gamma_1 = 35$ results in the largest IR rate $\mathcal{R}^{\text{NB-MLC}(a)}_{IR}$.

Table 1 Parameters used in simulation

Parameter	Value
p	$2^{32}-5$
l_p	31
l_{ht}	32 bits
R_{max}	$H(X\|Y+0.1)$
R_{min}, R_{step}	0.01
d_v	3
$d_v^{\max}$	5
$L(x)$	$\sum_{d=2}^{d_v^{\max}} L_d$
Δ_1	0.1
Δ_2	0.05
Γ	50
Γ_1	35
N	2000
$(\alpha, \sigma_1, \mu_1, \sigma_2, \mu_2, \beta)$	(0.013, 1.084, 0.212, 17.175, 1.719, 0.0028)

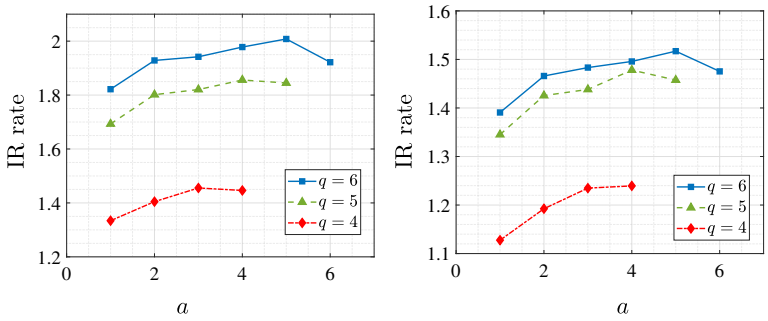

Fig. 8 IR rate for different q as the NB-MLC(a) protocol parameter a is varied. Left Panel: $P_{Y|X}$ is given by Eq. (3) with $(\alpha, \sigma_1, \mu_1, \sigma_2, \mu_2, \beta) = (0.005, 1.1, 0.2, 17, 1.5, 0.0025)$. Right Panel: $P_{Y|X}$ derived empirically from our experimental data with binwidth 100 ps. In both the figures, NB-MLC(a) protocol utilizes binary mapping for $u_b()$ and SDC. All plots use a VN degree regular LDPC code with a constant VN degree 3 constructed using the PEG algorithm [43]

right panel corresponds to our experimental data. From the figure, we can see that for all values of q, the IR rate is non-monotonic in a and has a maximum when a is strictly between 1 and q. As explained in Sect. 3, the IR rate is non-monotonic in a due to the following two effects: i) Increasing a makes the NB-MLC(a) protocol utilize NB-LDPC codes from a larger Galois field which are stronger resulting in improved FER performance and better IR rates per layer. ii) More number of layers due to a smaller a, however, has a positive effect on the IR rate due to the sum IR rate formula in Eq. (8). The combined effect of (i) and (ii) makes the IR rate non-monotonic. Note that, as described in Sect. 3, increasing the value of a increases the complexity of the NB-MLC(a) protocol monotonically. Thus, based on Fig. 8, the NB-MLC(a)

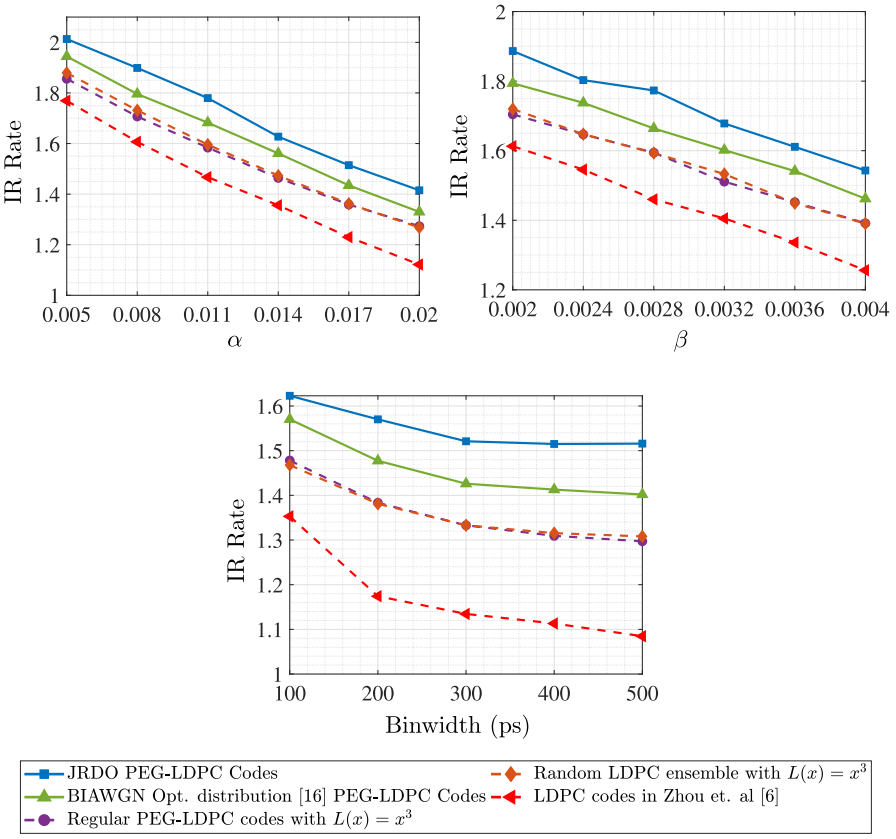

Fig. 9 IR Rate for different NB-LDPC code constructions. Left and right panels have $P_{Y|X}$ given by Eq. (3). Left panel: IR rate vs α for $(\sigma_1, \mu_1, \sigma_2, \mu_2, \beta) = (1.1, 0.2, 17, 1.5, 0.0025)$; Right Panel: IR rate vs β for $(\alpha, \sigma_1, \mu_1, \sigma_2, \mu_2) = (0.005, 1.1, 0.2, 17, 1.5)$; Bottom panel: IR rate vs binwidth where $P_{Y|X}$ is derived empirically from our experimental data for different binwidths. In all figures, the NB−MLC(a) protocol uses $q = 5$, $a = 4$, binary mapping for $u_b()$ and SDC

protocol with a small value of a (3 or 4) provides the best trade-off between IR rate and complexity. Additionally, note that the points $a = 1$ in the different curves in the figure correspond to the MLC scheme of [6]. We can clearly see that by using $a = 3$ or 4, there is a large improvement in IR rates compared to using $a = 1$.

In Fig. 9, we demonstrate the performance of the JRDO algorithm. In the figure, we compare the IR rate of JRDO-LDPC codes with the IR rates obtained by other code constructions used in prior work. The left and right panels correspond to the parameterized channel model in Eq. (3), where we vary the channel parameters α and β, respectively, while keeping the rest of the parameters fixed. The bottom panel corresponds to our experimental data. The red curves correspond to NB-LDPC codes used in the MLC scheme [6]. As per [6], these LDPC codes are randomly constructed such that each VN has a constant degree of 3. Note that there is no limitation on the CN degree distribution in [6]. The orange curves correspond to LDPC codes chosen

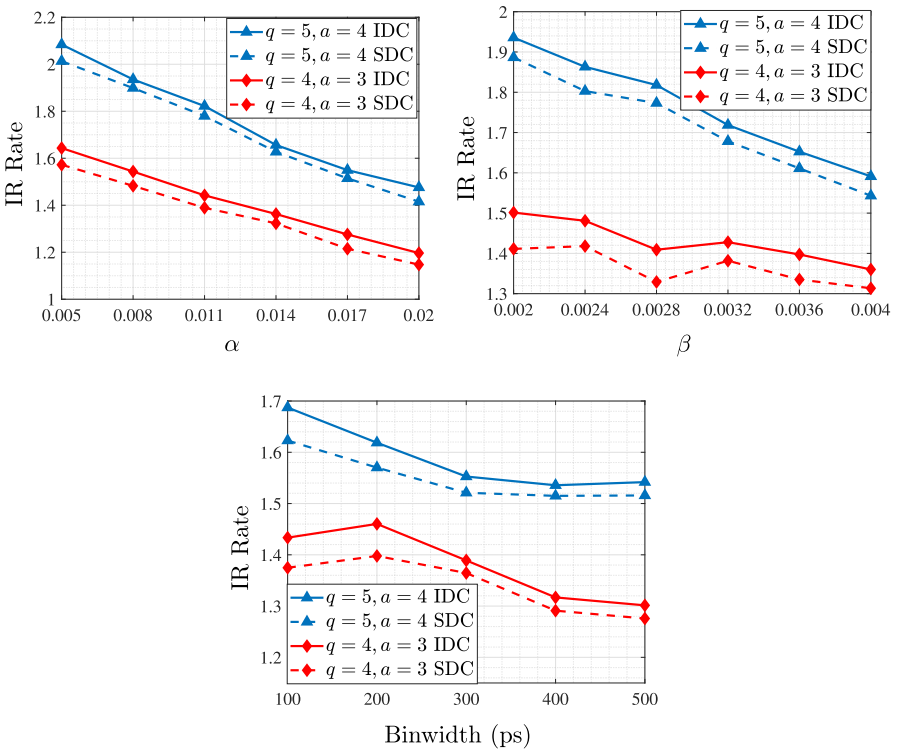

Fig. 10 IR rate comparison for the IDC and SDC protocol. Left and Right panels have $P_{Y|X}$ given by Eq. (3). Left panel: IR rate vs α for $(\sigma_1, \mu_1, \sigma_2, \mu_2, \beta) = (1.1, 0.2, 17, 1.5, 0.0025)$; Right Panel: IR rate vs β for $(\alpha, \sigma_1, \mu_1, \sigma_2, \mu_2) = (0.005, 1.1, 0.2, 17, 1.5)$; Bottom panel: IR rate versus binwidth where $P_{Y|X}$ is derived empirically from our experimental data for different binwidths. In all figures, the NB-MLC(a) protocol uses binary mapping for $u_b()$ and JRDO PEG-LDPC codes

from a random LDPC ensemble [10] with regular VN degree distribution $L(x) = x^3$ (similar to [6]) but with a two-element CN degree distribution (that is chosen to result in the required coding rate). Note that these type of CN degree distributions are called *concentrated* [10]. The purple curves correspond to NB-LDPC code constructed using the PEG algorithm [43] with regular VN degree distribution $L(x) = x^3$. The PEG algorithm is known to result in *concentrated* CN degree distributions [43] similar to the ones used in the orange curve. The green curves correspond to NB-LDPC codes constructed using the PEG algorithm using the degree distribution provided in [27, Table I] with a maximum VN degree 5. Note that this degree distribution is optimized for the BIAWGN channel. Finally, the blue curves correspond to NB-LDPC codes constructed using the PEG algorithm with degree distributions and rates obtained using the JRDO algorithm. From the three plots in Fig. 9, we make the following observations. The IR rates for the red curves are worse compared to the orange and purple curves. This trend suggests that it is better to use a concentrated CN degree distribution. Note that the IR rates for the orange and purple curves are very close. The IR rates for the green curves (BIAWGN optimized degree distribution) are better

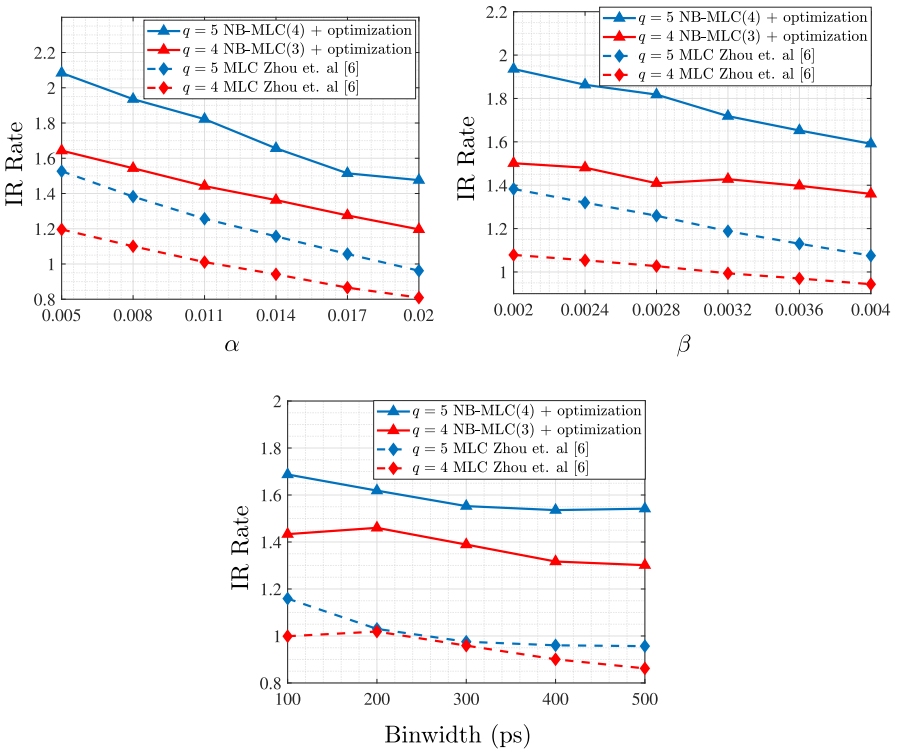

Fig. 11 IR rate comparison of our techniques combined (solid curves) vs. the MLC scheme of [6] (dotted curves). The solid curves are the result of utilizing the NB-MLC(a) protocol with JRDO PEG-LDPC codes and the IDC protocol. All curves use binary mapping. Left and Right panels have $P_{Y|X}$ given by Eq. (3). Left panel: IR rate vs α for $(\sigma_1, \mu_1, \sigma_2, \mu_2, \beta) = (1.1, 0.2, 17, 1.5, 0.0025)$; Right Panel: IR rate vs β for $(\alpha, \sigma_1, \mu_1, \sigma_2, \mu_2) = (0.005, 1.1, 0.2, 17, 1.5)$; Bottom panel: IR rate vs binwidth where $P_{Y|X}$ is derived empirically from our experimental data for different binwidths. The curves corresponding to $q = 5$ in the right panel are presented in Fig. 1

compared to the purple curves (VN degree 3 regular LDPC codes). This trend suggests that it is better to use irregular LDPC codes compared to regular LDPC codes to get improved IR rates. Finally, we observe that the blue curves that correspond to JRDO-LDPC codes have better IR rates compared to the green curve and result in the largest IR rates among all codes. The reason JRDO-LDPC codes have higher IR rates compared to other codes is because they are optimized for the ET-QKD channel.

In Fig. 10, we compare the performance of the interleaved decoding and communication (IDC) and sequential decoding and communication (SDC) protocols. Note that the SDC protocol was utilized in [6]. Similar to Fig. 9, the left and right panels correspond to the parameterized channel model with varying α and β, respectively, and the bottom panel corresponds to our experimental data. We compare the performance for NB-MLC(a) protocol parameters $q = 4, a = 3$ (blue curves) and $q = 5, a = 4$ (red curves). The solid curves correspond to IDC, while the dotted curves correspond to SDC. From the figure, we can clearly see that for different choices of protocol parameters and channel conditions, the IDC protocol always results in a greater IR rate

compared to the SDC protocol. As explained in Sect. 4.2, the IDC protocol improves the IR rate since it strategically utilizes the channels γ_{int}^i, $1 \leq i \leq T$, during the decoding of each layer of the NB-MLC(a) protocol which provides more reliable information about the reconciled keys $\mathbf{X}_i$ compared to the channels γ_{seq}^i, $1 \leq i \leq T$, used in SDC.

In Fig. 11, we combine all the techniques introduced in this paper and demonstrate the overall improvement in the IR rate compared to the MLC scheme of [6]. The solid curves correspond to our techniques and utilize the NB-MLC(a) protocol with JRDO PEG-LDPC codes and the IDC protocol. The values of a in the NB-MLC(a) protocol are chosen (as per the discussion in Fig. 8) to improve the IR rate without much increase in complexity. The dotted curves correspond to the MLC scheme of [6] that utilizes randomly constructed LDPC codes with regular VN degree distribution $L(x) = x^3$ and the SDC protocol. From the curves, we can clearly see a significant improvement in the IR rates using our techniques compared to the MLC scheme. Overall, our techniques result in around 40–60% improvement in IR rates on actual experimental data (right panel) demonstrating their efficacy.

6 Conclusion

In this paper, we considered the problem of IR in ET-QKD systems and proposed a protocol for IR called NB-MLC(a). The NB-MLC(a) protocol offers flexibility in system design in terms of IR rate and complexity via the parameter a. Additionally, using a small value of a (3 or 4), the NB-MLC(a) protocol results in a significant improvement in the IR rate compared to prior work without a large increase in complexity. To further improve the IR rate performance of the NB-MLC(a) protocol, we proposed the JRDO algorithm to design NB-LDPC codes for each layer and the IDC scheme to decode the different layers of the NB-MLC(a) protocol. Overall, NB-MLC(a) protocol that uses NB-LDPC codes designed by the JRDO algorithm and the IDC scheme results in a significant 40–60% improvement in IR rate compared to prior work. The techniques proposed in this work can be additionally combined with the adaptive modulation techniques of [51] to further improve the IR rates. It is an exciting direction of future research to tailor the NB-MLC(a) protocol to use adaptive modulation.

Acknowledgements All authors were supported by NSF grant CCF-CIF no. 2312872 and NSF grant QuIC-TAQS no. 2137984. D. Mitra was supported by UCLA Dissertation Year Fellowship. L. Tauz was supported by the NSF-UCLA AIF-Q: Quantum Science and Engineering PhD Fellowship.

Author Contributions All authors contributed to the conception of the research and its analysis. D.M. and J.S. created new software used in this work. M.C.S. carried out the physical experiments and provided the experimental data used in this work. D.M. wrote the main manuscript text. All authors reviewed the manuscript.

Data Availability The code and data used for this work are available under reasonable request to the corresponding author.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Bennett, C.H., Brassard, G.: Quantum cryptography: public key distribution and coin tossing. Theoret. Comput. Sci. **560**, 7–11 (2014)
2. Lo, H.-K., Curty, M., Tamaki, K.: Secure quantum key distribution. Nat. Photonics **8**(8), 595–604 (2014)
3. Zhuang, Q., Zhang, Z., Dove, J., Wong, F.N., Shapiro, J.H.: Floodlight quantum key distribution: a practical route to gigabit-per-second secret-key rates. Phys. Rev. A **94**(1), 012322 (2016)
4. Zhang, Z., Zhuang, Q., Wong, F.N., Shapiro, J.H.: Floodlight quantum key distribution: demonstrating a framework for high-rate secure communication. Phys. Rev. A **95**(1), 012332 (2017)
5. Zhong, T., Zhou, H., Horansky, R.D., Lee, C., Verma, V.B., Lita, A.E., Restelli, A., Bienfang, J.C., Mirin, R.P., Gerrits, T., et al.: Photon-efficient quantum key distribution using time-energy entanglement with high-dimensional encoding. New J. Phys. **17**(2), 022002 (2015)
6. Zhou, H., Wang, L., Wornell, G.: Layered schemes for large-alphabet secret key distribution. In 2013 Information Theory and Applications Workshop (ITA). IEEE, pp. 1–10 (2013)
7. Dolecek, L., Soljanin, E.: Qkd based on time-entangled photons and its key-rate promise. IEEE BITS Inf. Theory Mag. **2**(3), 39–48 (2022)
8. Zhang, Z., Mower, J., Englund, D., Wong, F.N., Shapiro, J.H.: Unconditional security of time-energy entanglement quantum key distribution using dual-basis interferometry. Phys. Rev. Lett. **112**(12), 120506 (2014)
9. Chang, K.-C., Cheng, X., Sarihan, M.C., Vinod, A.K., Lee, Y.S., Zhong, T., Gong, Y.-X., Xie, Z., Shapiro, J.H., Wong, F.N., et al.: 648 Hilbert-space dimensionality in a biphoton frequency comb: entanglement of formation and Schmidt mode decomposition. NPJ Quantum Inf. **7**(1), 48 (2021)
10. Richardson, T., Urbanke, R.: Modern Coding Theory. Cambridge University Press, Cambridge (2008)
11. Jiang, X.-Q., Yang, S., Huang, P., Zeng, G.: High-speed reconciliation for CVQKD based on spatially coupled LDPC codes. IEEE Photonics J. **10**(4), 1–10 (2018)
12. Johnson, S.J., Chandrasetty, V.A., Lance, A.M.: Repeat-accumulate codes for reconciliation in continuous variable quantum key distribution. In 2016 Australian Communications Theory Workshop (AusCTW). IEEE, pp. 18–23 (2016)
13. Zhang, M., Dou, Y., Huang, Y., Jiang, X.-Q., Feng, Y.: Improved information reconciliation with systematic polar codes for continuous variable quantum key distribution. Quantum Inf. Process. **20**, 1–16 (2021)
14. Zhang, M., Hai, H., Feng, Y., Jiang, X.-Q.: Rate-adaptive reconciliation with polar coding for continuous-variable quantum key distribution. Quantum Inf. Process. **20**, 1–17 (2021)
15. 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. **19**(10), 350 (2020)
16. Li, Q., Wen, X., Mao, H., Wen, X.: An improved multidimensional reconciliation algorithm for continuous-variable quantum key distribution. Quantum Inf. Process. **18**, 1–20 (2019)
17. Brádler, K., Mirhosseini, M., Fickler, R., Broadbent, A., Boyd, R.: Finite-key security analysis for multilevel quantum key distribution. New J. Phys. **18**(7), 073030 (2016)
18. Bennett, C.H., Brassard, G., Crépeau, C., Maurer, U.M.: Generalized privacy amplification. IEEE Trans. Inf. Theory **41**(6), 1915–1923 (1995)
19. Lee, C., Bunandar, D., Zhang, Z., Steinbrecher, G.R., Dixon, P.B., Wong, F.N., Shapiro, J.H., Hamilton, S.A., Englund, D.: Large-alphabet encoding for higher-rate quantum key distribution. Opt. Express **27**(13), 17539–17549 (2019)

20. Davey, M.C., MacKay, D.J.: Low density parity check codes over gf (q). In 1998 Information Theory Workshop (Cat. No. 98EX131). IEEE, pp. 70–71 (1998)
21. Martinez-Mateo, J., Elkouss, D., Martin, V.: Key reconciliation for high performance quantum key distribution. Sci. Rep. **3**(1), 1576 (2013)
22. Wehner, S., Elkouss, D., Hanson, R.: Quantum internet: a vision for the road ahead. Science **362**(6412), 9288 (2018)
23. 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. **11**(4), 045023 (2009)
24. Yang, S., Sarihan, M.C., Chang, K.-C., Wong, C.W., Dolecek, L.: Efficient information reconciliation for energy-time entanglement quantum key distribution. In 2019 53rd Asilomar Conference on Signals, Systems, and Computers. IEEE, pp. 1364–1368 (2019)
25. Storn, R., Price, K.: Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**, 341–359 (1997)
26. Price, K., Storn, R.M., Lampinen, J.A.: Differential Evolution: A Practical Approach to Global Optimization. Springer, Berlin (2006)
27. Richardson, T.J., Shokrollahi, M.A., Urbanke, R.L.: Design of capacity-approaching irregular low-density parity-check codes. IEEE Trans. Inf. Theory **47**(2), 619–637 (2001)
28. Lucamarini, M., Yuan, Z.L., Dynes, J.F., Shields, A.J.: Overcoming the rate-distance limit of quantum key distribution without quantum repeaters. Nature **557**(7705), 400–403 (2018)
29. Xie, Y.-M., Weng, C.-X., Lu, Y.-S., Fu, Y., Wang, Y., Yin, H.-L., Chen, Z.-B.: Scalable high-rate twin-field quantum key distribution networks without constraint of probability and intensity. Phys. Rev. A **107**(4), 042603 (2023)
30. Xie, Y.-M., Lu, Y.-S., Weng, C.-X., Cao, X.-Y., Jia, Z.-Y., Bao, Y., Wang, Y., Fu, Y., Yin, H.-L., Chen, Z.-B.: Breaking the rate-loss bound of quantum key distribution with asynchronous two-photon interference. PRX Quantum **3**(2), 020315 (2022)
31. Zeng, P., Zhou, H., Wu, W., Ma, X.: Mode-pairing quantum key distribution. Nat. Commun. **13**(1), 3903 (2022)
32. Shao, S.-F., Cao, X.-Y., Xie, Y.-M., Gu, J., Liu, W.-B., Fu, Y., Yin, H.-L., Chen, Z.-B.: Phase-matching quantum key distribution without intensity modulation. Phys. Rev. Appl. **20**(2), 024046 (2023)
33. Yin, H.-L., Fu, Y., Li, C.-L., Weng, C.-X., Li, B.-H., Gu, J., Lu, Y.-S., Huang, S., Chen, Z.-B.: Experimental quantum secure network with digital signatures and encryption. Natl. Sci. Rev. **10**(4), 228 (2023)
34. Zhou, L., Lin, J., Xie, Y.-M., Lu, Y.-S., Jing, Y., Yin, H.-L., Yuan, Z.: Experimental quantum communication overcomes the rate-loss limit without global phase tracking. Phys. Rev. Lett. **130**(25), 250801 (2023)
35. Li, W., Zhang, L., Lu, Y., Li, Z.-P., Jiang, C., Liu, Y., Huang, J., Li, H., Wang, Z., Wang, X.-B., et al.: Twin-field quantum key distribution without phase locking. Phys. Rev. Lett. **130**(25), 250802 (2023)
36. Gu, J., Cao, X.-Y., Fu, Y., He, Z.-W., Yin, Z.-J., Yin, H.-L., Chen, Z.-B.: Experimental measurement-device-independent type quantum key distribution with flawed and correlated sources. Sci. Bull. **67**(21), 2167–2175 (2022)
37. Lu, F.-Y., Wang, Z.-H., Zapatero, V., Chen, J.-L., Wang, S., Yin, Z.-Q., Curty, M., He, D.-Y., Wang, R., Chen, W., et al.: Experimental demonstration of fully passive quantum key distribution. Phys. Rev. Lett. **131**(11), 110802 (2023)
38. Trushechkin, A.S., Kiktenko, E.O., Kronberg, D.A., Fedorov, A.K.: Security of the decoy state method for quantum key distribution. Phys. Usp. **64**(1), 88 (2021)
39. Fedorov, A., Kiktenko, E., Trushechkin, A.: Symmetric blind information reconciliation and hash-function-based verification for quantum key distribution. Lobachevskii J. Math. **39**, 992–996 (2018)
40. Yang, S.: Application-Driven Coding Techniques: From Cloud Storage to Quantum Communications. University of California, Los Angeles (2021)
41. Dupraz, E., Savin, V., Kieffer, M.: Density evolution for the design of non-binary low density parity check codes for Slepian-wolf coding. IEEE Trans. Commun. **63**(1), 25–36 (2014)
42. Declercq, D., Fossorier, M.: Decoding algorithms for nonbinary LDPC codes over gf(q). IEEE Trans. Commun. **55**(4), 633–643 (2007)
43. Hu, X.-Y., Eleftheriou, E., Arnold, D.-M.: Regular and irregular progressive edge-growth tanner graphs. IEEE Trans. Inf. Theory **51**(1), 386–398 (2005)
44. Goldberg, D.E.: Genetic algorithms in search optimization and machine learning. (1988)

45. Rechenberg, I.: Evolutionsstrategie: Optimierung technischer systeme nach prinzipien der biologischen evolution (1973)
46. Schwefel, H.-P.: Evolution and optimum seeking. In Sixth-generation Computer Technology Series (1995). https://api.semanticscholar.org/CorpusID:43418053
47. Glover, F.W., Kochenberger, G.A.: Handbook of Metaheuristics. Springer, Berlin (2006)
48. Shokrollahi, A., Storn, R.: Design of efficient erasure codes with differential evolution. In 2000 IEEE International Symposium on Information Theory (Cat. No. 00CH37060). IEEE, p. 5 (2000)
49. Hou, J., Siegel, P.H., Milstein, L.B.: Performance analysis and code optimization of low density parity-check codes on Rayleigh fading channels. IEEE J. Sel. Areas Commun. **19**(5), 924–934 (2001)
50. Caire, G., Taricco, G., Biglieri, E.: Bit-interleaved coded modulation. IEEE Trans. Inf. Theory **44**(3), 927–946 (1998)
51. Karimi, E., Soljanin, E., Whiting, P.: Increasing the raw key rate in energy-time entanglement based quantum key distribution. In 2020 54th Asilomar Conference on Signals, Systems, and Computers. IEEE, pp. 433–438 (2020)