Continuous-variable quantum key distribution with composable security and tight error-correction bound towards constrained-device implementations

Panagiotis Papanastasiou [1,3] Carlo Ottaviani [2,3] Stefano Pirandola [2] Poonam Yadav,[2] and Marco Lucamarini [1,3]
[1]*School of Physics, Engineering & Technology, University of York, York YO10 5FT, United Kingdom*
[2]*Department of Computer Science, University of York, York YO10 5GH, United Kingdom*
[3]*York Centre for Quantum Technologies, University of York, York YO10 5FT, United Kingdom*

(Received 8 April 2025; accepted 13 July 2026; published 11 August 2026)

Constrained devices, such as smart sensors, wearable devices, and Internet of Things nodes, are increasingly prevalent in society and rely on secure communications to function properly. These devices often operate autonomously, exchanging sensitive data or commands over short distances, such as within a room, house, or warehouse. In this context, continuous-variable quantum key distribution (CV-QKD) offers the highest secure key rate and the greatest versatility for integration into existing infrastructure. A key challenge in this setting, where devices have limited storage and processing capacity, is obtaining a realistic and tight estimate of the CV-QKD secure key rate within a composable security framework, with error correction (EC) consuming most of the storage and computational power. To address this, we focus on low-density parity-check codes with nonbinary alphabets, which optimize mutual information and are particularly suited for short-distance communications. We develop a security framework to derive finite-size secret keys near the optimal EC leakage limit and model the related memory requirements for the encoding process in one-way error correction. This analysis facilitates the practical deployment of CV-QKD, particularly in constrained devices with limited storage and computational resources.

DOI: 10.1103/2s9n-s2br

I. INTRODUCTION

Quantum key distribution (QKD) [1] allows two parties to establish a common secret key, which can later be used in symmetric cryptographic primitives. Its security relies on the fundamental principles of quantum physics rather than computational complexity conjectures [2–5] and constitutes, along with postquantum cryptography (PQC) [6], the leading candidate for countering quantum threats, such as an eavesdropper equipped with a quantum computer.

Initially, QKD was developed for discrete-variable (DV) systems, which use discrete degrees of freedom, such as the polarization of the electromagnetic field. Later, protocols based on continuous degrees of freedom, such as the quadratures of the electromagnetic field, i.e., continuous-variable (CV) systems [7], emerged, offering high performance in the asymptotic regime and over short distances, as well as compatibility with existing technological infrastructure. Recent studies have advanced both the security [8,9] and the experimental performance [10,11] of CV-QKD, bringing it close to the repeaterless PLOB bound [12] and making it comparable to DV-QKD.

The most common and earliest CV-QKD protocols employ Gaussian modulation of coherent states (GMCS), utilizing homodyne detection [13] or heterodyne detection [14] in direct or reverse reconciliation (RR) [15]. CV-QKD has also been extended to protocols using different frequencies (thermal states), two-way communication [16–20], and network settings [21–32]. Schemes employing postselection [33–35] and discrete modulation [36–41] have also been considered.

Recent demonstrations of continuous-variable quantum key distribution (CV-QKD) [42–46] have used integrated photonic systems [47], paving the way for deploying QKD protocols on constrained devices such as drones, vehicles, Internet of Things (IoT) devices, and wearables. As these devices become ubiquitous in industrial, healthcare, and consumer networks, it is nearly impossible at design time to predict which endpoints will eventually handle mission-critical or privacy-sensitive data [48]. Moreover, untrusted edge nodes often represent the weakest link in a defense-in-depth architecture, potentially creating backdoors into deeper network layers [49].

PQC remains the most mature "plug-and-play" solution for providing quantum-resistant security and is already under benchmarking for constrained-device scenarios [50]. It requires a smooth integration of standardized algorithms into existing stacks [51]. However, it still relies on unproven hardness assumptions and cannot match the information-theoretic security (ITS) of QKD [4,52].

Recognizing this trade-off, one can view cryptographic security solutions as spanning a spectrum from classical cryptography through PQC to QKD, with each step requiring greater implementation effort but delivering stronger

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

secrecy. Rather than relying on any single approach, emerging best practices advocate hybrid PQC and QKD schemes that combine the strengths of both methods [53–55]. Therefore, beyond the challenges imposed by their small form factor, it is essential to identify the data postprocessing limitations of such devices to predict the implementation performance of CV-QKD protocols given a specific level of security.

In this study, we focus on the GMCS protocol, which is particularly effective in short- to moderate-loss regimes, and quantify the postprocessing requirements, especially error correction (EC), which can significantly impact the protocol's performance [56–63]. Inspired by the analysis of Ref. [9], which imposes strict bounds on the secret-key length, we quantify the protocol's performance under composable security with finite-size effects.

Furthermore, we adapt the tight one-way EC bound from Ref. [64] to nonbinary low-density parity-check (LDPC) codes [65–67] to compute secret key rates with near-optimal performance. This allows us to estimate the storage requirements of the encoder during EC. Although this represents only half of the complete EC procedure, it provides valuable insight in scenarios where there is a strong asymmetry in computational resources between the CV-QKD transmitter and receiver. Specifically, the transmitter, which performs the encoding, can be lightweight and agile, while the receiver, responsible for decoding, may be bulkier and better suited for computationally intensive tasks.

This scenario arises, for example, in networks of small sensors transmitting to a central processing unit, as seen in smart home sensors, wearable devices, IoT systems [68–73], drones [74], or free-space implementations [75,76]. These devices play a crucial role in modern society, supporting the technological infrastructure by enabling automation, real-time monitoring, and critical decision making. As they become increasingly integrated into daily life, cyber threats targeting them pose significant risks to public safety. We suggest feasible cases where CV-QKD can be used to safeguard against these threats.

In Sec. II A, we introduce the composable secret-key length bound. In Sec. II B, we derive the information leakage in one-way EC for nonbinary LDPC codes. In Sec. II C, we present the final formula for the secret-key rate, and in Sec. III, we introduce one-way EC with nonbinary LDPC codes. Finally, we link these results to an estimate of the storage required for the encoding of the EC procedure to achieve near-optimal performance. Section IV presents our numerical results and Sec. V concludes our work.

II. COMPOSABLE SECURITY RATE WITH TIGHT LEAKAGE BOUND

To provide an accurate and richer description of the security aspects of GMCS protocols, our analysis takes into consideration finite-size effects along with data postprocessing contributions, which are also relevant to assess the implementation and various performance trade-offs. We adopt the security proof of Ref. [9] that contains parameters such as the failure probability of EC, the reconciliation efficiency, or the residual probability of EC, which are connected to the EC leakage needed by the parties to reconcile their raw key.

These parameters can be calculated by applying an EC procedure after running simulations [61,62], predicting accurate results for the performance of the system. However, theoretical works can predict the performance of the EC procedure assuming specific techniques. In particular, Ref. [64] provides bounds for the near-optimal (but achievable) performance of EC based on (binary) LDPC codes assuming finite-size effects.

In this section, we extend the previous analysis to the nonbinary LDPC codes connecting the EC performance and security parameters of the secret-key rate formula from Ref. [9], a useful tool for benchmarking the behavior of the CV-QKD protocols in different settings. As we will discuss in the next section, we can connect this result to the storage requirements for EC encoding.

A. Secret-key length with composable terms

Before identifying the relevant parameters for EC performance connected to the nonbinary LDPC codes, we present and summarize results from Ref. [9] about the secret-key rate in a composable framework.

The secret-key length is upper-bounded by the following expression (see also Ref. [[9], Eq. (36)]):

$$s_n^{\epsilon_{\rm cor}+\epsilon_{\rm h}+\epsilon_{\rm s}} \leqslant nR_\infty - \sqrt{n}\Delta_{\rm aep}^{\epsilon_{\rm s}}(h) + \theta. \tag{1}$$

When one considers collective Gaussian attacks assuming that the protocol did not abort, the extracted key is correct, with probability larger than $1-\epsilon_{\rm cor}$, and secret with probability larger than $1-\epsilon_{\rm sec}$. The parameter $\epsilon_{\rm sec}=\epsilon_{\rm s}+\epsilon_{\rm h}$, where $\epsilon_{\rm s}$ is the smoothing parameter and $\epsilon_{\rm h}$ is the probability of failure of privacy amplification (PA). The asymptotic key rate R_∞ in Eq. (1) is given by

$$R_\infty = H(\mathsf{k}) - \chi(\mathsf{k}:E) - {\rm leak}_{\rm ec}, \tag{2}$$

where the secret-key variable k takes values from the alphabet $\mathcal{K}=\{0,1,\ldots,2^{hd}-1\}$ for d-bit digitization of the normalized quadrature results [9,61,62]. The quantity $\Delta_{\rm aep}^{\epsilon_{\rm s}}$, given by the expression

$$\Delta_{\rm aep}^{\epsilon_{\rm s}}(h) \simeq 4\log_2(\sqrt{2^{hd}}+2)\sqrt{\log_2\left(2/\epsilon_{\rm s}^2\right)}, \tag{3}$$

accounts for the penalty due to finite-size effects, while

$$\theta = \log_2\left(2\epsilon_{\rm h}^2\epsilon_{\rm cor}\right) \tag{4}$$

is the penalty paid for nonideal verification and privacy amplification, while n is the block size of the raw key after channel parameter estimation (PE) and ${\rm leak}_{\rm ec}$ is the error-correction (EC) leakage.

The quantity $H(\mathsf{k})$, in Eq. (2), is the Shannon entropy of the key variable k and $\chi(\mathsf{k}:E)$ is Eve's Holevo information. In this description, the dummy variable $h=\{1,2\}$ distinguishes between homodyne ($h=1$) and heterodyne ($h=2$) detection. More specifically, for the heterodyne protocol, the digitized outcomes are in the vectorial form (k_q,k_p), which can be concatenated as

$$\mathsf{k} = k_q 2^d + k_p, \tag{5}$$

without loss of any information because the mapping $(k_q,k_p)\leftrightarrow\mathsf{k}$ is one-to-one, and with Shannon entropies related by the following mathematical expression: $H(\mathsf{k})=$

$2H(k)$ (see Appendix A for details). For error correction, the parties may now decide to use the vectorial form $\{k_q^{(1)}, k_p^{(1)}, k_q^{(2)}, k_p^{(2)}, \ldots, k_q^{(n)}, k_p^{(n)}\}$.

From here, we apply the procedure given in Ref. [[9], Eq. (48)], to write Eq. (2) as

$$R_\infty = hH(k) - \chi(x:E) - \mathrm{leak}_{\mathrm{ec}}, \tag{6}$$

where the variable x is the continuous-variable version of k. Note that for the homodyne protocol, k and k are equivalent forms of the key variable. The steps to obtain the Holevo bound for the reverse-reconciliation (RR) protocol have been detailed in Ref. [9], hence, we give only the final expression for the covariance matrix (CM) for the direct reconciliation (DR) protocol in Appendix B.

We can now calculate the secret-key rate after channel PE, replacing in the Holevo bound the channel parameters, excess noise ξ and transmissivity τ, with their worst-case scenario values $\tau^{\epsilon_{\mathrm{pe}}}$ and $\xi^{\epsilon_{\mathrm{pe}}}$ [77], respectively, given in Ref. [[9], Eqs. (85) and (86)]. These values are dependent on the efficiency η_d, the electronic noise u_{el} of the detection, and the number of signals. We then obtain the expression

$$R_n^{\epsilon_{\mathrm{pe}}} = hH(k) - \mathrm{leak}_{\mathrm{ec}} - \chi(x:E)|_{\tau^{\epsilon_{\mathrm{pe}}},\xi^{\epsilon_{\mathrm{pe}}}}, \tag{7}$$

encapsulating the finite-size dependence of the secret-key rate due to PE, and replace R_∞ in Eq. (1).

B. Theoretical estimation of EC leakage

In the following discussion, we bound the leakage term of the previous formula adopting the approach of Ref. [64] after considering the nonbinary LDPC case. Then we connect it to EC-performance and security parameters such as the reconciliation efficiency, successful verification probability, and EC failure probability.

In the case of one-way reconciliation, where low-density parity-check (LDPC) codes [65,66] are used (see, also, Sec. III), the EC leakage term can be upper-bounded by the number of syndrome bits, given by

$$\mathrm{leak}_{\mathrm{ec}} \leqslant n^{-1} \log_2 |\mathcal{M}|, \tag{8}$$

where $\mathcal{M}$ is the alphabet of the syndrome strings. One may calculate the size of the alphabet $|\mathcal{M}|$ via EC simulations, as done in Ref. [60] for asymptotic security analysis and in Refs. [28,61,62] for composable security analysis.

However, in a complete theoretical analysis, one may use the asymptotic bound (Slepian-Wolf coding [81]), stating that

$$\log_2 |\mathcal{M}| - nhH(k|y) \geqslant 0, \tag{9}$$

where $H(k|y)$ is the conditional Shannon entropy of k conditioned on the continuous variable y of the other party. In fact, when considering finite-size effects, one can use a more rigorous bound [64] providing a tighter estimate of the performance of information reconciliation. Such a bound is given by

$$\left|\log_2 |\mathcal{M}| - nhH(k|y) - \sqrt{n}\Delta_{\mathrm{leak}}^{\epsilon_{\mathrm{ec}}}(h)\right| \leqslant \delta(n), \tag{10}$$

where

$$\Delta_{\mathrm{leak}}^{\epsilon_{\mathrm{ec}}}(h) = \sqrt{hV(k|y)}\Phi^{-1}(1-\epsilon_{\mathrm{ec}}), \tag{11}$$

with Φ the cumulative normal distribution and ϵ_{ec} a bound for $P[\hat{k}^{hn} \neq k^{hn}]$, where $\hat{k}^{hn}$, k^{hn} are the strings of the parties after EC. The right-hand side of Eq. (10) is given by

$$\delta(n) = \tfrac{1}{2}\log_2 hn + \mathcal{O}(1), \tag{12}$$

while the conditional entropy and the conditional entropy variance are given by

$$H(k|y) = \mathbb{E}[-\log_2 p(k|y)],\ V(k|y) = \mathrm{Var}[-\log_2 p(k|y)]. \tag{13}$$

We now extend the calculation of Eq. (10) to nonbinary alphabets (i.e., $d > 1$) using the expression of $p(k|y)$ given in Eq. (68) of Ref. [61]. That allows the parties to achieve higher mutual information at short distances because, when d increases, the entropy of the digitized variables approaches that of their continuous-variable counterparts.

To minimize the probability of errors in the final key string, the parties apply the verification step: one party sends the syndrome and a hash of the raw key k^{hn} with collision probability ϵ_{cor}. The other party will compare this with the hash of the guessed string $\hat{k}^{hn}$ and, if they match, then the protocol can continue with probability p_{ec} (we denote this event with $\top$), otherwise they will abort ($\perp$). In this case, the conditional probability of error is bounded by $P[\hat{k}^{hn} \neq k^{hn}|\top] \leqslant \epsilon_{\mathrm{cor}}/p_{\mathrm{ec}}$, which abides with the security definition in Refs. [9,82] about the joint probability,

$$P[\hat{k}^{hn} \neq k^{hn} \wedge \top] = p_{\mathrm{ec}} P[\hat{k}^{hn} \neq k^{hn}|\top] \leqslant \epsilon_{\mathrm{cor}}. \tag{14}$$

To be more specific, in an honest implementation for Eve, the parameter estimation step completely defines and bounds the channel parameters τ and ξ prior to error correction. Based on these estimations, the parties decide the probability of successful verification p_{ec} along with the collision probability ϵ_{cor}. Through them, they can decide the largest probability of failure that their EC LDPC scheme can tolerate given the previous parameters. Therefore, we need to calculate an upper bound for the marginal probability connected to Eq. (11),

$$\begin{aligned} P[\hat{k}^{hn} \neq k^{hn}] &= P[\hat{k}^{hn} \neq k^{hn} \wedge \top] + P[\hat{k}^{hn} \neq k^{hn} \wedge \perp] \\ &\leqslant \epsilon_{\mathrm{cor}} + (1-p_{\mathrm{ec}})P[\hat{k}^{hn} \neq k^{hn}|\perp] \\ &\leqslant \epsilon_{\mathrm{cor}} + (1-p_{\mathrm{ec}}) \\ &= 1 - (p_{\mathrm{ec}} - \epsilon_{\mathrm{cor}}) := \epsilon_{\mathrm{ec}}. \end{aligned} \tag{15}$$

Therefore, from Eq. (10), we have that

$$\log_2 |\mathcal{M}| \leqslant nhH(k|y) + \sqrt{n}\Delta_{\mathrm{leak}}^{\epsilon_{\mathrm{ec}}}(h) + \delta(n), \tag{16}$$

to be used in Eq. (8) and subsequently in Eq. (7). We then obtain

$$\begin{aligned} R_n^{\epsilon_{\mathrm{pe}}+\epsilon_{\mathrm{ec}}} &= hH(k) - hH(k|y) - \sqrt{n^{-1}}\Delta_{\mathrm{leak}}^{\epsilon_{\mathrm{ec}}}(h) - \frac{\delta(n)}{n} \\ &\quad - \chi(x:E)|_{\tau^{\epsilon_{\mathrm{pe}}},\xi^{\epsilon_{\mathrm{pe}}}} \end{aligned} \tag{17}$$

$$= hI(k:y) - \chi(x:E)_{\tau^{\epsilon_{\mathrm{pe}}},\xi^{\epsilon_{\mathrm{pe}}}} - \sqrt{n^{-1}}\Delta_{\mathrm{leak}}^{\epsilon_{\mathrm{ec}}}(h), \tag{18}$$

where we omitted terms $\mathcal{O}(\frac{\log_2 n}{n})$ in our calculations, and $I(k:y) = H(k) - H(k|y)$ is the mutual information between k and y. This is the rate that includes finite-size effects connected only to PE and the performance of EC.

We can also group the mutual information and $\Delta_{\text{leak}}^{\epsilon_{\text{ec}}}$ term in Eq. (18), to define the quantities

$$\zeta_{\text{leak}} hI(k:y) := hI(k:y) - \Delta_{\text{leak}}^{\epsilon_{\text{ec}}}(h)/\sqrt{n} \tag{19}$$

and

$$\zeta_{\text{digit}} := hI(k:y)/I(x:y), \tag{20}$$

where the mutual information $I(x:y)$ refers to the Gaussian variables as described in Ref. [[9], Eq. (83)]. Using Eqs. (19) and (20) into Eq. (18), we can rewrite the rate as follows:

$$R_n^{\epsilon_{\text{pe}}+\epsilon_{\text{ec}}} = \zeta I(x:y) - \chi(x:E)|_{\tau^{\epsilon_{\text{pe}}}, \xi^{\epsilon_{\text{pe}}}}, \tag{21}$$

where

$$\zeta = \zeta_{\text{digit}} \zeta_{\text{leak}}. \tag{22}$$

C. Secret-key rate with tight estimation of EC performance

Before we make the connection with the storage requirements for EC encoding, we present the main result of our previous discussion (see, also, Appendix B).

Replacing $R_n^{\epsilon_{\text{pe}}}$ of Eq. (7) with $R_n^{\epsilon_{\text{pe}}+\epsilon_{\text{ec}}}$ from either Eq. (18) or (21), we obtain

$$s_n^{\epsilon}/n \leqslant r_n^{\epsilon} := \; R_n^{\epsilon_{\text{pe}}+\epsilon_{\text{ec}}} - \frac{\Delta_{\text{aep}}^{\epsilon_s}(h)}{\sqrt{n}} + \frac{\theta}{n} - \mathcal{O}[\log_2(n)/n]. \tag{23}$$

This gives the highest number of bits per signal that can be extracted with security parameter ϵ and a tight estimate of EC leakage, and it can be used to compute the final (expected) secret-key rate,

$$R := p_{\text{ec}}(n/N) r_n^{\epsilon}, \tag{24}$$

where N is the total number of signals in the block. Here, the choice of p_{ec} entails a trade-off: a higher target p_{ec} requires higher leakage and, as a result, a smaller key.

III. LDPC CODES FOR NONBINARY VARIABLES AND STORAGE REQUIREMENTS

Here, we describe the EC encoding procedure with LDPC codes connecting it to its storage requirements. We further use the bound of the previous section to quantify numerically the leakage and the storage needed for a specific secret-key rate performance. We consider an EC scheme, where the encoder obtains a string k^{hn} and computes the syndrome $\mathbf{H}k^{hn} = s^r$, where $\mathbf{H}$ is a $r \times hn$ parity-check matrix. In general, every element of the matrix belongs to the Galois field $k \in \mathcal{GF} = \{0, 1, \ldots, 2^d - 1\}$. More specifically, the matrix is a representation of a Tanner graph [61], with hn message nodes and r parity-check nodes. When a message $i = 1, \ldots, hn$ is included in a parity-check $j = 1, \ldots, r$, there is an edge between the corresponding message node and parity-check node, while the entry H_{ji} of the associated parity-check matrix in this intersection is chosen randomly from $k = 1, \ldots, 2^d - 1$.

One then may calculate the corresponding code rate,

$$R_{\text{code}} = \frac{hn - r}{hn} = 1 - R_{\text{synd}}, \tag{25}$$

with R_{synd} being the syndrome rate. The code is usually designed by assuming that each message participates in d_{v} (column weight) checks and every check contains d_c (row weight) messages. Then the number of edges must follow $hnd_{\text{v}} = rd_{\text{c}}$ (for sparse matrices, i.e., $d_{\text{v}} < d_{\text{c}} \ll r < hn$). Therefore, by replacing $r := hn\frac{d_{\text{v}}}{d_{\text{c}}}$ in Eq. (25), we obtain another equivalent expression,

$$R_{\text{code}} = 1 - \frac{d_{\text{v}}}{d_{\text{c}}}, \tag{26}$$

where we usually set $d_{\text{v}} = 2$ as it gives the best performance in decoding [61]. This means that the rates for regular LDPCs are given by

$$R_{\text{code}} = 0.333, 0.5, 0.6, 0.666, 0.714, 0.75, \ldots, 0.777, 0.8 \ldots.$$

On the other hand, irregular LDPC codes can be designed where the column and row weights are not constant. Through the probability distributions of the weights, one defines their means $\bar{d}_{\text{c}}$ and $\bar{d}_{\text{v}}$, respectively. Therefore, we have

$$R_{\text{code}} = 1 - \frac{\bar{d}_{\text{v}}}{\bar{d}_{\text{c}}}, \tag{27}$$

which allows for more flexible values than before. Note that Eq. (26) allows us to generalize the notion of R_{code} to the irregular code case. However, the performance of these codes is not particularly stable, i.e., different structures with the same R_{code} can behave very differently at different signal-to-noise ratios (SNRs) or in terms of probability of successful EC. Then, from Eq. (8), we have that

$$\text{leak}_{\text{ec}} \leqslant n^{-1} \log_2 |\mathcal{M}| = n^{-1} dr = hdR_{\text{synd}}. \tag{28}$$

One then may define a tight approximation for the optimal R_{synd} by Eq. (16), where

$$R_{\text{synd}}^* = H(k|y)/d + \Delta_{\text{leak}}^{\epsilon_{\text{ec}}}(h)/(dh\sqrt{n}) + \delta(n)/(dhn). \tag{29}$$

Then, one may find values for the R_{code} that perform closely to the previous tight bound in terms of leakage (the corresponding structure of the codes needs to be optimized to achieve a certain probability of successful EC for a specific SNR). These are given through Eq. (25) by

$$R_{\text{code}}^* = 1 - R_{\text{synd}}^*, \tag{30}$$

where, by $n \to \infty$, we arrive at the asymptotic expression,

$$R_{\text{code}}^*|_{n\to\infty} = 1 - d^{-1} H(k|y).$$

Then, we can calculate the required memory to store the parity-check matrix $\mathbf{H}$ as

$$M_{\text{code}} := hn \times \log_2 |\mathcal{M}| = hndr = n^2 h^2 d R_{\text{synd}}. \tag{31}$$

For example, for a protocol using homodyne detection, with a block size of the raw key equal to $n = 10^5$ and $dR_{\text{synd}}^* = 4 \times 0.667$, we obtain the parity-check matrix storage to be around 3.34 GB, while for a protocol using heterodyne detection, it will be four times larger.

In the compressed row storage (CRS) format [[63], Sec. 6.3], one will need $\bar{d}_{\text{v}} \times hn \times d$ bits for storing the nonzero elements of the parity-check matrix, $\bar{d}_{\text{v}} \times hn \times \lceil \log_2(hn) \rceil$ bits for storing the column indices, and

TABLE I. We have chosen block sizes 1–4 $\times 10^5$ and found the associated R^*_{code} and M^*_{sparse} from Fig. 3. Then with the associated parameters, we have created parity-check matrices in the CRS format in Python with $R_{\text{code}} \approx R^*_{\text{code}}$, using regular nonbinary LDPC codes. We finally calculated the actual storage needed for these cases and listed the results under the M_{sparse} column. These points have been depicted in red in the bottom panel of Fig. 3 following the predicted performance.

$N(10^5)$	$n(10^5)$	R^*_{code}	M^*_{sparse}(MB)	R_{code}	M_{sparse}(MB)
1	0.676	0.78	0.389	0.777	0.464
2	1.6	0.78	0.95	0.777	1.1
4	3.2	0.7949	2	0.8	2.3

$(\frac{\log_2|\mathcal{M}|}{d}+1)\times\lceil\log_2(\bar{d}_v hn)\rceil$ bits for the row pointers. Gathering all these terms, one may calculate the sparse matrix representation storage as

$$M_{\text{sparse}} = \bar{d}_v hnd + \bar{d}_v\, hn\lceil\log_2(hn)\rceil + (hnR_{\text{synd}}+1)\lceil\log_2(\bar{d}_v hn)\rceil. \quad (32)$$

For a protocol using homodyne detection with $\bar{d}_v = 2$, $nR_{\text{synd}} = 0.667\times 10^5$, $d = 4$, and $n = 10^5$, we have that the sparse matrix storage is approximately 0.67 MB, while for a protocol using heterodyne detection, it will be two times larger.

Note here that Eqs. (31) and (32) calculate the practical storage associated with the parity-check matrix of a code with rate $R_{\text{code}} = 1 - R_{\text{synd}}$. Theoretically, one may compute tight bounds for these quantities through Eq. (10) and obtain an approximate prediction for them. Thus, we may write

$$M^*_{\text{code}} = (hn)^2 dR^*_{\text{synd}}, \quad (33)$$

$$M^*_{\text{sparse}} = \bar{d}_v hn(d+\lceil\log_2(hn)\rceil) + (hnR^*_{\text{synd}}+1)\lceil\log_2(\bar{d}_v hn)\rceil. \quad (34)$$

The most accurate estimation for these quantities is to simulate the results, i.e., create parity-check matrices for different block sizes (see Table I) with similar parameters and store them in CRS format. We have done this using the EC encoding script developed in the simulation library for the GMCS protocol with heterodyne detection [83]. This may give different results due to a particular choice of the type of variables in the script or other software parameters used to describe the parity-check matrix in the specific format that may add an overhead, which is not included in Eq. (34). Although this formula cannot give very accurate results, it is quite simple and can provide key insights for the encoding function. Notice that Eq. (34) through Eq. (29) associates the secret-key rate performance [see Eq. (24)] to the required storage when both are calculated using the same parameter values.

IV. RESULTS

In this section, we connect the preceding theoretical results to the practical implications for the protocol operations and especially the data postprocessing part. We focus on the protocol with homodyne detection. In Fig. 1, we plot the secret-key rate of Eq. (24) against the loss in dB, where we have optimized over V and the PE ratio $1-(n/N)$. We also plot the changes in terms of ζ or leakage and the associated

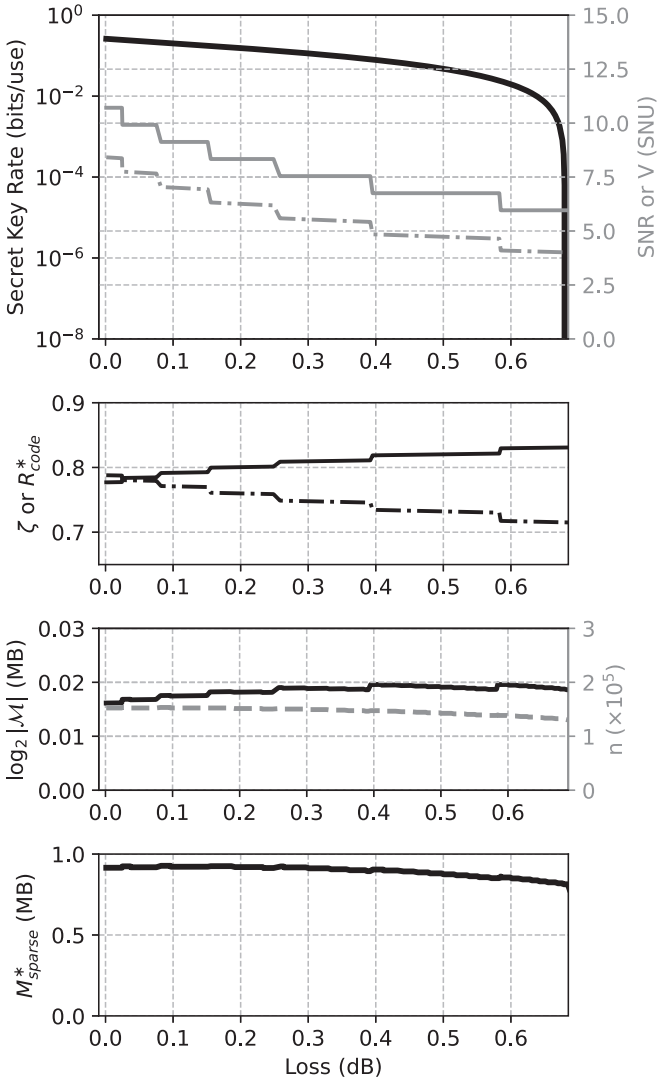


FIG. 1. In the top panel, we plot the secret-key rate of Eq. (24) (black line) of the Gaussian modulation protocol with coherent states and homodyne detection in direct reconciliation against the loss in dB. We have optimized the secret-key rate over the Gaussian modulation variance V (gray thin line) and the PE ratio of sacrificed channel uses. We have assumed $N = 2\times 10^5$, $d = 4$, and $p_{\text{ec}} = 0.9$. The rest of the parameters are given in Table II. We also plot the corresponding SNR (gray dash-dotted line) by replacing the optimal V in the formula given by Ref. [[62], Eq. (33)]. In the second panel, we plot the corresponding values of the reconciliation efficiency ζ (black line), which follow the same pattern as the Gaussian modulation variance values. However, we can see that for a constant value of V, the reconciliation efficiency increases linearly with the loss in dB. Subsequently, we plot the associated R^*_{code} (black dash-dotted line). In the third panel, we plot the corresponding block size (gray dashed line) after optimizing the number of sacrificed channel uses during PE. We also plot the associated leakage (black line), i.e., $\log_2|\mathcal{M}|$ from Eq. (16). In the bottom panel, we plot Eq. (34) for the corresponding values of the secret-key rate and loss in dB.

TABLE II. The common parameters used to plot the secret-key rate of Eq. (24) in all figures.

ξ	Excess noise	0.01
η_d	Detection efficiency	0.8
u_{el}	Electronic noise	0.01
τ	Channel transmissivity	$10^{-\text{dB}/10}$
ϵ_{h}	Failure of privacy amplification	2^{-32}
ϵ_{cor}	Correctness	2^{-32}
ϵ_{pe}	Failure of parameter estimation	2^{-32}
ϵ_{s}	Smoothing parameter	2^{-32}

SNR for the optimized rate value. These correspond to the values of V and n presented in the same figure. Furthermore, using the same parameters for every point in the plot of the secret-key rate, we have plotted the corresponding points for M_{sparse} in Eq. (32). The rest of the parameters used for this plot are summarized in Table II.

In Fig. 2, we plot the corresponding RR secret-key rate, where it is clear that the performance has increased in terms of loss tolerance. This is expected due to the 3 dB loss limit of the direct reconciliation protocol in the asymptotic regime, which degrades when one assumes finite-size effects. Clearly, the RR protocol is robust against higher losses (see Fig. 11), especially when one uses a large block size. This is not achievable by the DR protocol. We also include plots for the corresponding R^*_{code} that is needed to achieve the specific performance for the given SNR and, similarly, the corresponding M^*_{sparse}.

By contrast, in Figs. 3 and 4, we plot the secret-key rates for the DR and RR protocols, respectively, against the block size. We set the loss to 0.02 dB and choose the other parameters in the same way. We mainly observe that the DR protocol is advantageous in this regime of losses: we can achieve high rates by using smaller block size. Here, we consider a moderate block size of the order of 10^5–10^6. This is because we would like to investigate regimes of operation where the high-rate performance over long distances is not a priority. These regimes are described by fast sharing of small keys assuming the smallest requirements of hardware equipment, either because of space constraints or cost-effective implementations. For example, this can be described by QKD implementations over networks of small sensor devices, IoT nodes, wearable devices, or drones operating inside a building, outside, connected with fiber, or with free-space links [75,76].

Apart from the limitations due to the communication links such as the noise and the losses, one should take into account as a priority the hardware requirements of the classical data postprocessing. This can be done effectively, in the finite-size regime, using composable terms connected to every performance aspect. Therefore, we combine tools developed in previous studies to characterize the requirements in storage during one-way EC and especially the encoding part, which is executed by one of the parties. Note that the DR protocol is advantageous in this regime because it can give higher rates for smaller block size. In fact, it can support lightweight and agile transmitters responsible for the EC encoding, while the bulkier receivers can be better suited for computationally intensive tasks such as the EC decoding in an asymmetric scenario: for example, a network of small sensors transmitting to a central processing unit.

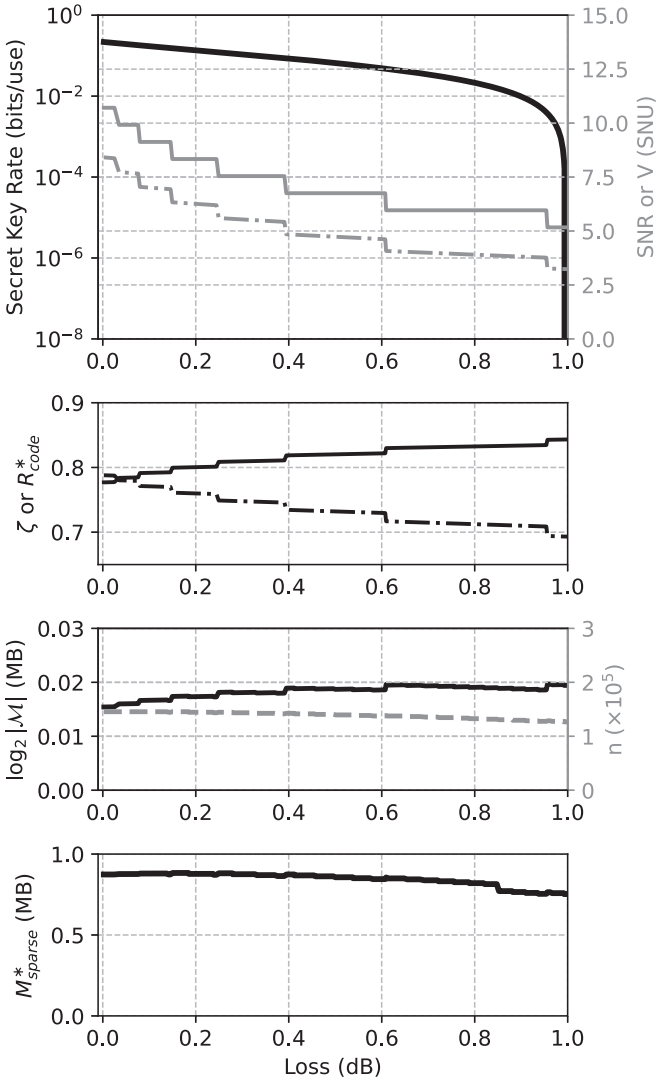


FIG. 2. In the top panel, we plot the secret-key rate of Eq. (24) (black line) of the Gaussian modulation protocol with coherent states and homodyne detection in RR against the loss in dB. The rest of the parameters and settings have been considered the same as in Fig. 1. Here we observe a better performance in terms of tolerable loss compared with the direct reconciliation protocol in Fig. 1 (see, also, Fig. 12). This is expected due to the known 3 dB loss limit of the DR protocol in the ideal asymptotic regime, which decreases even more when one considers finite-size effects and channel noise.

The syndrome creation, i.e., encoding process, is the less difficult part of the one-way EC with LDPC codes. In contrast, the decoding process is rather demanding and can be effectively handled by larger stations rather than a constrained device. In such an asymmetric scenario in terms of computational power, for an appropriate loss tolerance, the party operating through the constrained device is the transmitter and also the encoder during EC. This describes the DR protocol. In this setting, one can exploit the trade-off between limitations in robustness against losses with the mitigation of computational-power requirements.

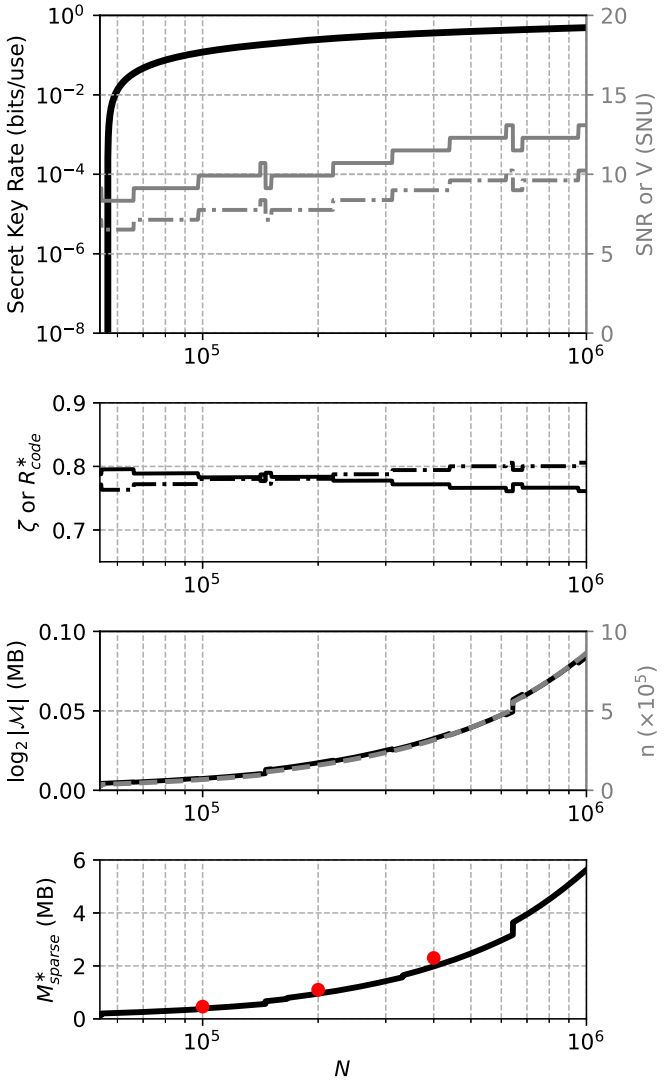


FIG. 3. In the top panel, we plot the secret-key rate of Eq. (24) (black line) for the Gaussian modulation protocol with coherent states and homodyne detection in direct reconciliation against N. We set the loss to 0.02 dB. We have optimized over V and n/N. The rest of the parameters, settings, and lines are the same as in Fig. 1. We plot in red the points included in Table I of the corresponding M_{sparse}, i.e., the actual storage needed for these sparse matrices after creating them in CRS format using the GMCS simulation library in [83].

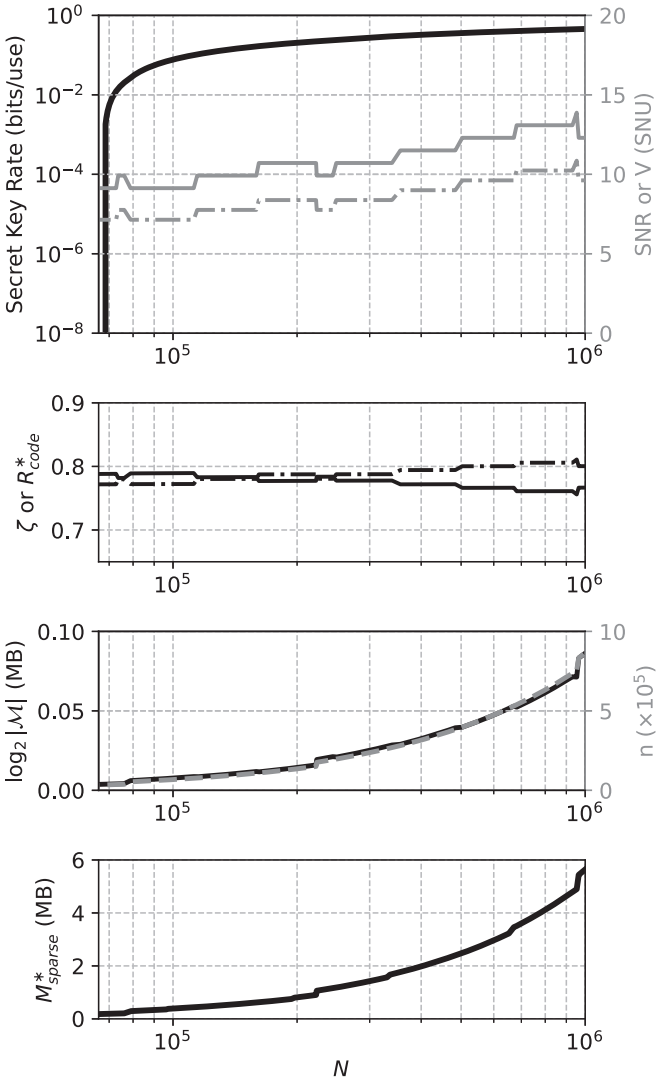


FIG. 4. In the top panel, we plot the secret-key rate of Eq. (24) (black line) for the Gaussian modulation protocol with coherent states and homodyne detection in RR against N. We have set the loss to 0.02 dB. The rest of the parameters, settings, and lines are the same as in Fig. 1. We observe here that we need a larger block size in order to obtain a secret-key rate compared to the DR protocol in Fig. 3 (see, also, Fig. 13). This is because the DR protocol offers higher rates than the RR protocol in the low-loss regime.

Then, one still needs to check compatibility with the storage requirements for the parity-check matrix as the main aspect of the EC encoding procedure. In particular, the amount of leakage calculated through Eq. (10) that achieves a specific performance in terms of secret-key rate in Eq. (24) can be mapped to the associated LDPC code rate in Eq. (30) and, in turn, to the dimensions of the related parity-check matrices that give the associated leakage. Then we can predict the related storage requirements by Eq. (34).

In Fig. 3, we can see the behavior of M^*_{sparse} against the block size. For specific block sizes, we have created the parity-check matrices for the corresponding encoding process in EC and stored them in sparse form. The details of these parity-check matrices are presented in Table I while, for the sake of comparison, the storage for each matrix is described by the red points in the bottom panel. We see that the results are very close to the predicted values for M_{sparse}. Finally, in Appendix C, we examine the behavior of the protocols in terms of losses when one considers different values for block size, successful EC probability, and digitization.

Based on this analysis, short-range networks of devices with a constrained EC encoding memory may support CV-QKD connections. This is a good starting point which, with further refinements and support from technological advances, can lead to constrained-device networks such as IoT, drones, and wearable devices to fully support CV-QKD connections.

V. CONCLUSION

Small device detectors and IoT device networks play a crucial role in modern society, enabling real-time monitoring, automation, and seamless connectivity across various sectors, including healthcare, smart cities, industrial automation, and environmental monitoring. Their ability to collect and process vast amounts of data enhances efficiency, reduces human intervention, and supports decision making in critical applications. However, as these devices become deeply integrated into daily life, their security is of paramount importance. Cyber threats targeting IoT networks can lead to privacy breaches, unauthorized surveillance, or even large-scale disruptions in infrastructure, posing risks to public safety and economic stability.

Robust protection against cyber threats is therefore essential for these devices; QKD is one candidate, offering an information-theoretic security advantage. Since these devices operate at short distances, such as within a room, a house, or a warehouse, CV-QKD, which has an advantage in this regime and especially the GMCS protocols, can provide higher secret-key rates. In particular, by using nonbinary LDPC codes in a practical implementation of such a protocol, the parties exploit the high mutual information between their continuous variables.

However, this is quite challenging due to the increased requirements of the data postprocessing stage in computational power or storage, not to mention implementing a QKD protocol on such constrained devices in the first place. Therefore, in order to investigate the performance under those circumstances, one needs to develop rigorous theoretical tools. Here we have combined a composable security proof taking into account the main stages of the data postprocessing of a QKD protocol with a tight bound for EC adapting it to the nonbinary LDPC regime.

This allows us to predict optimal secret-key performance in terms of reconciliation efficiency and leakage and match this performance to operational code rates for given signal-to-noise ratios and error-correction success probabilities. Based on this, we developed a tool that models the encoding storage requirements for a nonbinary LDPC EC associated with the given secret-key rate performance.

We combined a composable security framework for the secret-key rate of the GMCS protocols with a tight finite-size bound on one-way EC leakage of nonbinary LDPC dependent on the given SNR and successful EC probability. Through this tool, one can theoretically calculate the code rate for near-optimal performance and the dimensions of the associated parity-check matrix. In turn, one may calculate the storage requirements of the EC encoding process, which is crucial, for example, for the implementation of CV-QKD with constrained devices. Note here that optimal leakage corresponds to minimal storage use.

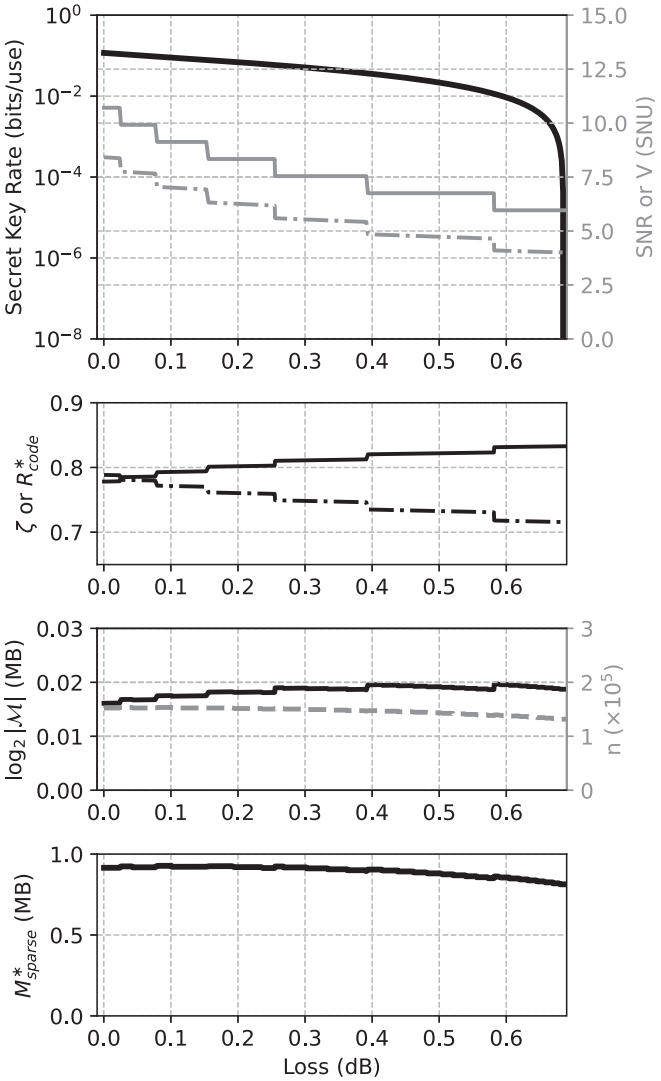


FIG. 5. We present similar plots to Fig. 1. All the parameters are the same, apart from $p_{ec} = 0.4$. We observe that the secret-key rate is lower than the corresponding one in the previous figure. However, the achievable loss has been slightly improved. We see also very similar performance in terms of the other parameters: ζ, R^*_{code}, leakage, and M_{sparse}.

ACKNOWLEDGMENTS

P.P. thanks Juan Vieira Giestinhas and Alexander George Mountogiannakis for insightful discussions and acknowledges the University of York's Viking high-performance computing facility. This work is supported, in part, by EPSRC and DSIT TMF-uplift: CHEDDAR: Communications Hub For Empowering Distributed ClouD Computing Applications and Research (Grants No. EP/X040518/l and No. EP/Y037421/l). S.P. acknowledges support from EPSRC and UKRI, via the Integrated Quantum Networks (IQN) Research Hub (Grant No. EP/Z533208/1).

DATA AVAILABILITY

The software that supports the findings of this article is openly available [84]. The majority of the numerical results and plots in this manuscript were produced using custom Python code developed for the calculation of tight leakage bounds in the context of nonbinary LDPC codes. This code represents a central technical contribution of the present work and is publicly available at [84]. The three red data points in Fig. 3 were obtained using an independent implementation, as referenced in the main text. All simulations were performed

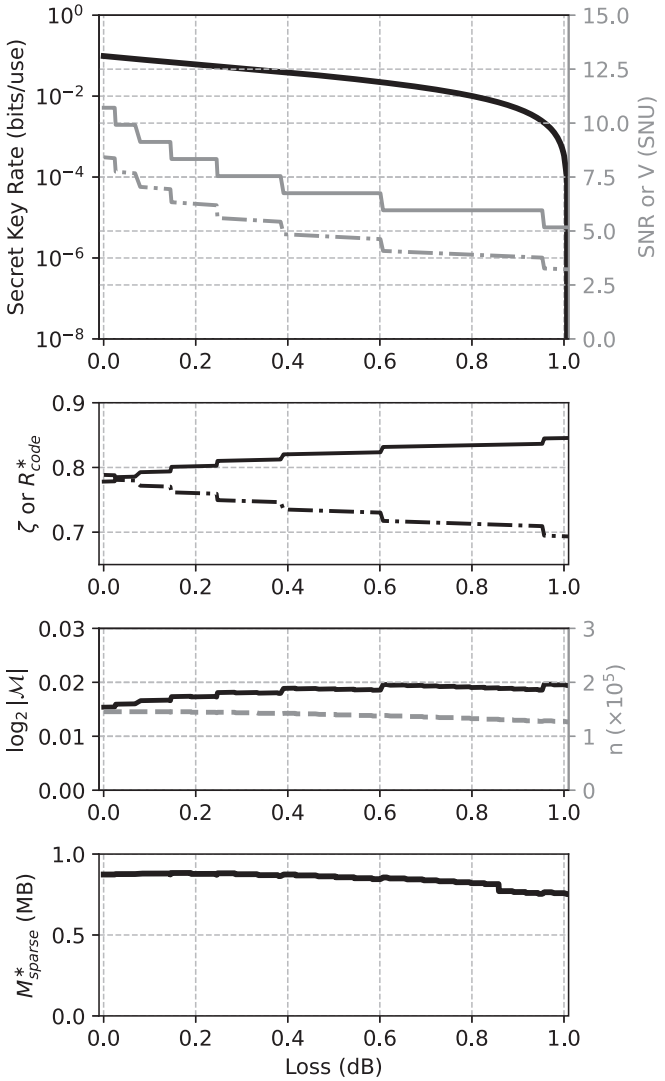


FIG. 6. We present similar plots to Fig. 2. All the parameters are the same, apart from $p_{ec} = 0.4$. We observe similar behavior for the rate as in Fig. 5.

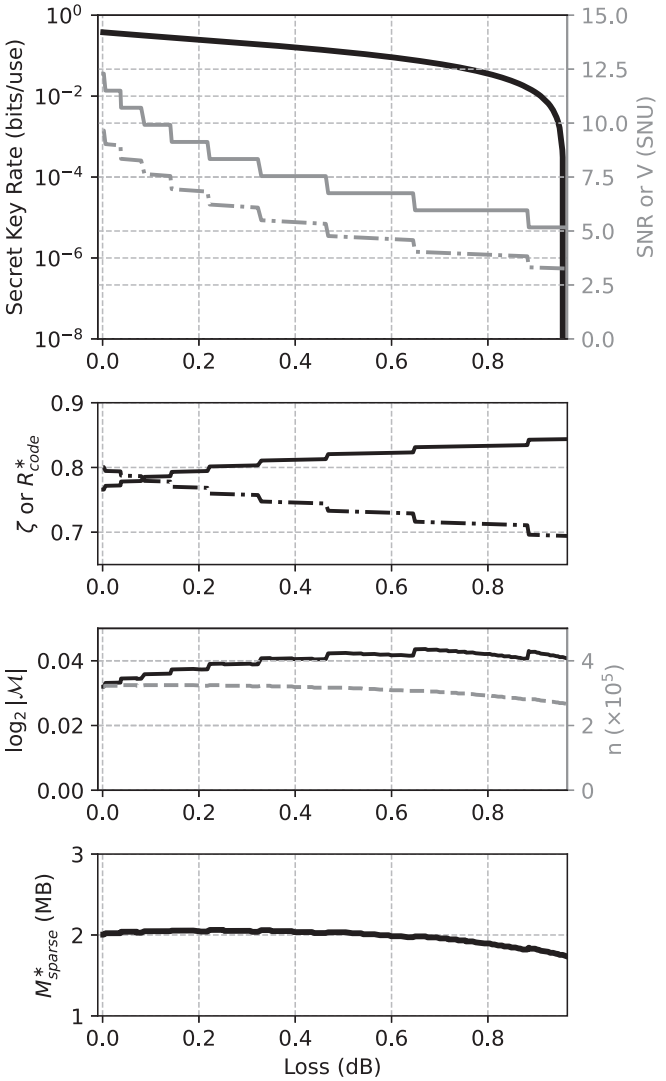


FIG. 7. We present similar plots to Fig. 1. All the parameters are the same, apart from assuming a doubled value for block size $N = 4 \times 10^5$. We observe that the secret-key rate takes higher values and the achievable loss has been significantly increased. The leakage level and n have been increased in the same manner (almost doubled) and, similarly, M^*_{sparse}. The other parameters have remained at similar levels as before.

on nodes of the Viking High Performance Computing cluster [85] at the University of York, equipped with a 2-core AMD EPYC3 7643 processor and 12 GB of memory. All repositories are released under the Apache License 2.0 and include documentation to support reproducibility.

APPENDIX A: VIRTUAL CONCATENATION

Before digitization, the parties may apply classical data postprocessing procedures such as rotations and rescaling, which bring the CM in its normal form, i.e., cross-quadrature correlation terms vanish. After digitization, one may apply the concatenation step useful only on the protocols using heterodyne detection. This is a virtual step that results in a common description of the GMCS protocols using either homodyne or heterodyne detection. This is summarized by

$$\mathsf{k} = k \tag{A1}$$

for the case of homodyne detection and

$$\mathsf{k} = k_q 2^d + k_p \tag{A2}$$

for the case of heterodyne detection. We observe that

$$p(\mathsf{k}) = p(k_q, k_p) = p(k_q)p(k_p), k_q = k_p = k \Rightarrow p(k_q) = p(k_p). \tag{A3}$$

Therefore,

$$\begin{aligned} H(\mathsf{k}) = H(k_q, k_p) &= -\sum_{k_q,k_p} p(k_q, k_p) \log_2 p(k_q, k_p) \\ &= -\sum_{k_q,k_p} p(k_q, k_p) \log_2 p(k_q) - \sum_{k_q,k_p} p(k_q, k_p) \log_2 p(k_p) \\ &= -\sum_{k_q} p(k_q) \log_2 p(k_q) - \sum_{k_p} p(k_p) \log_2 p(k_p) \\ &= -\sum_{k} p(k) \log_2 p(k) - \sum_{k} p(k) \log_2 p(k) = 2H(k). \end{aligned} \tag{A4}$$

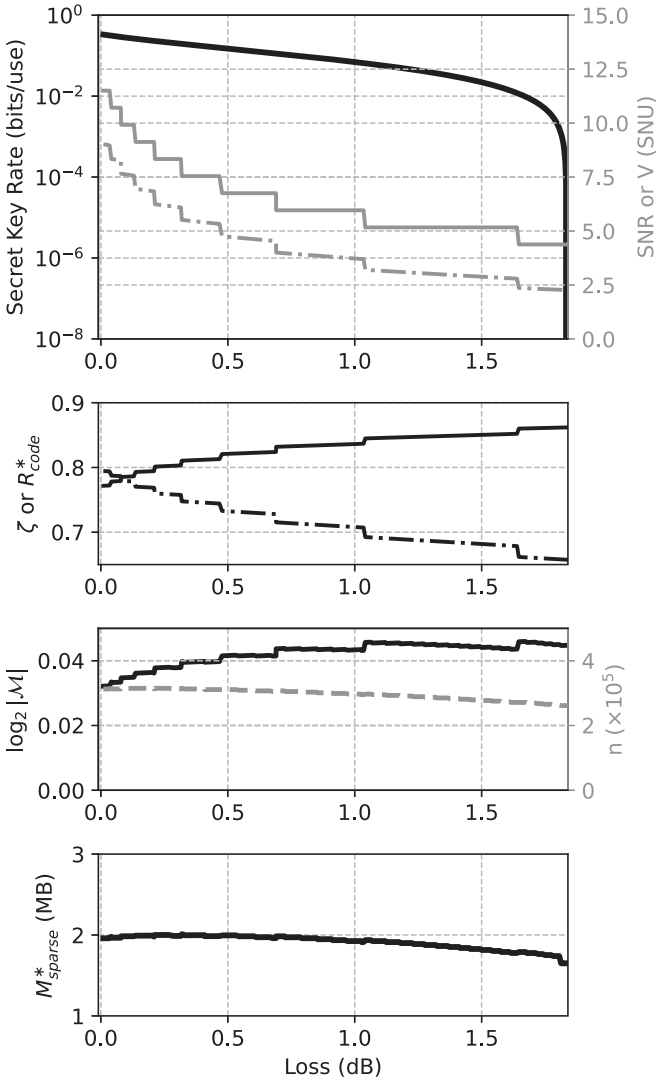


FIG. 8. We present similar plots to Fig. 2. All the parameters are the same, apart from assuming a doubled value for block size $N = 4 \times 10^5$. We observe similar behavior for the rate as in Fig. 7.

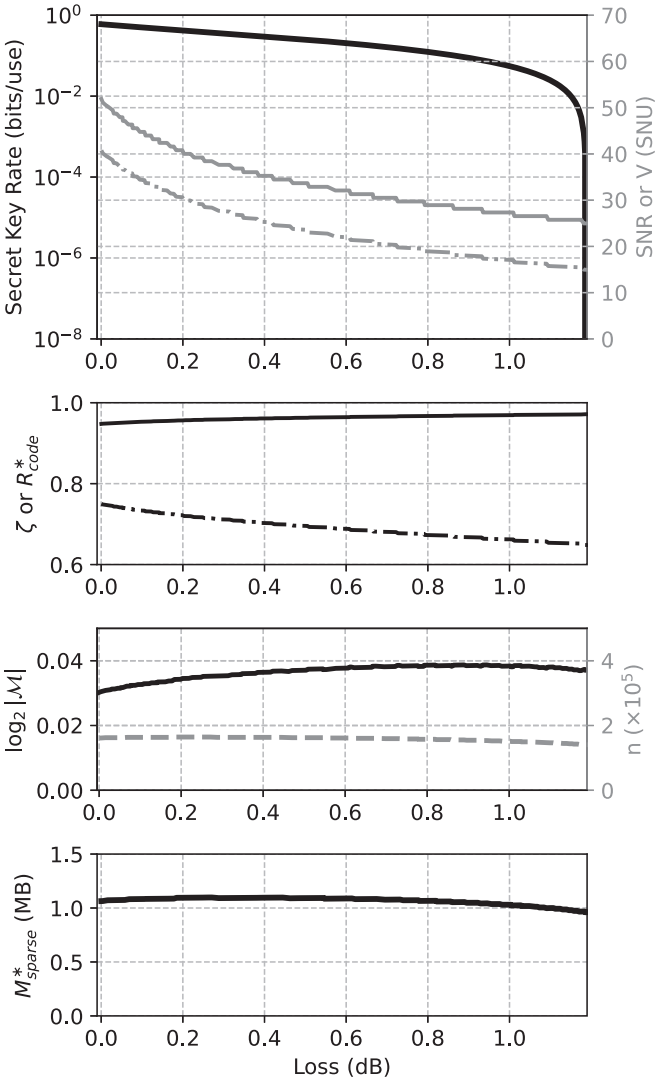


FIG. 9. We present similar plots to Fig. 1. All the parameters are the same, apart from assuming a digitization of $d = 6$. We observe that the secret-key rate takes higher values and the achievable loss has been significantly increased. The improvement is better compared to increasing the block size, as in the case of Fig. 7. By increasing the digitization, we allow the choice for larger optimal values for the Gaussian modulation variance which leads to higher SNRs and very high reconciliation efficiency.

APPENDIX B: SECRET-KEY RATE SUMMARY

Here we write, in a more concise way, the final equation of the secret-key rate used for the calculations of the previous results. By combining Eq. (21) with Eq. (23) and multiplying with the term $\frac{n}{N}$ connected with the PE ratio and the probability of remaining blocks p_{ec}, we obtain

$$R := p_{\text{ec}} \frac{n}{N} \Big\{ [\zeta I(x:y) - \chi(x:E)|_{\tau^{\epsilon_{\text{pe}}}, \xi^{\epsilon_{\text{pe}}}}] - \frac{\Delta^{\epsilon_s}_{\text{aep}}(h)}{\sqrt{n}} + \frac{\theta}{n} - \mathcal{O}[\delta(n)/n] \Big\}, \tag{B1}$$

where ζ is given by Eq. (22), θ is given by Eq. (4), and $\Delta^{\epsilon_s}_{\text{aep}}(h)$ is given by Eq. (3). For RR, the mutual information $I(x:y)$ and the Holevo information $\chi(x:E)$ are given in Refs. [61,62]. For DR, one uses the same rationale as in Refs. [61,62], but using the following conditional CMs: for heterodyne detection,

$$\mathbf{V}_{E|x} = \begin{pmatrix} \text{diag}\{\phi_0, \phi\} & \psi\mathbf{Z} \\ \psi\mathbf{Z} & \omega\mathbf{I} \end{pmatrix}, \tag{B2}$$

and for homodyne detection,

$$\mathbf{V}_{E|x} = \begin{pmatrix} \phi_0\mathbf{I} & \psi\mathbf{Z} \\ \psi\mathbf{Z} & \omega\mathbf{I} \end{pmatrix}, \tag{B3}$$

with

$$\phi_0 = \tau\omega + (1-\tau), \tag{B4}$$

$$\phi = \tau\omega + (1-\tau)(V+1), \tag{B5}$$

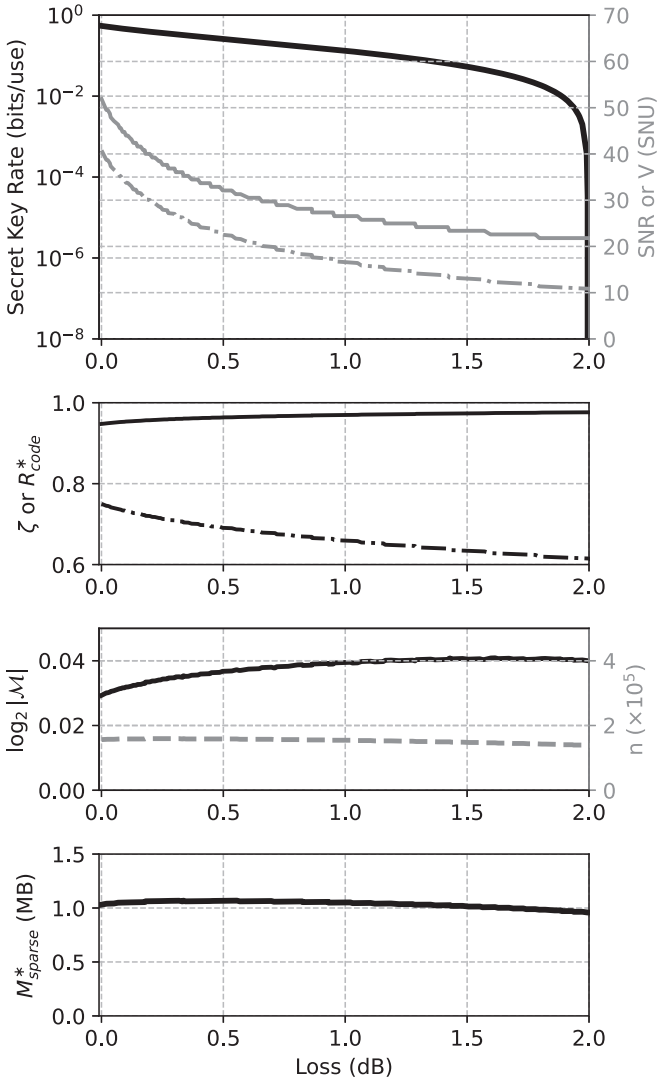


FIG. 10. We present similar plots to Fig. 2. All the parameters are the same, apart from assuming a digitization of $d = 6$. We observe similar behavior for the rate as in Fig. 9.

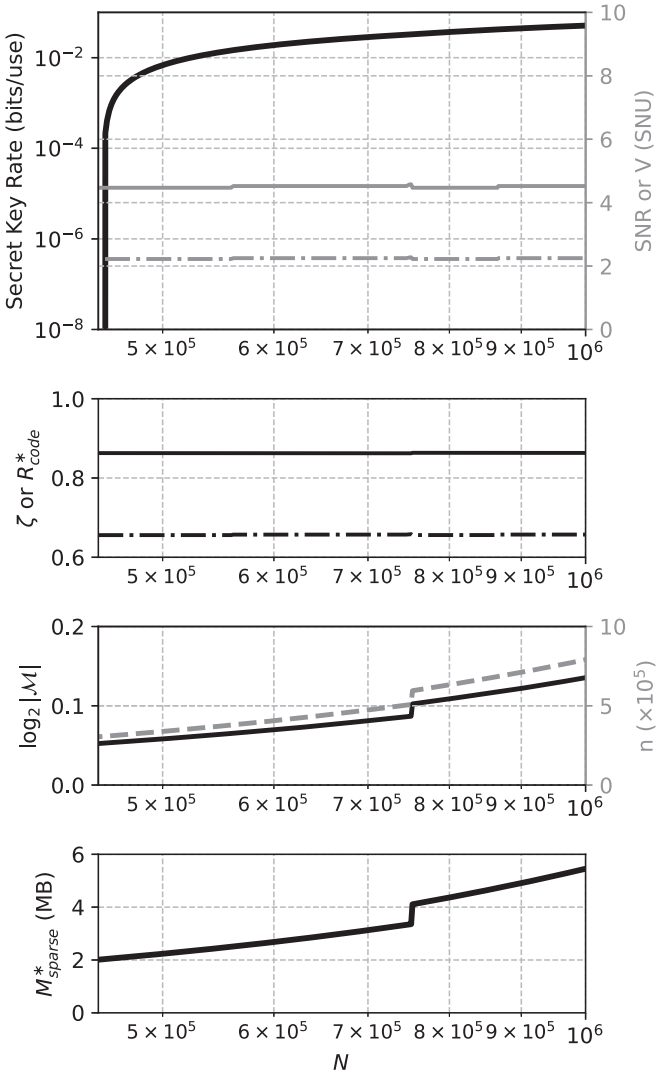


FIG. 11. We present similar plots to Fig. 4. All the parameters are the same, apart from setting the losses to 2 dB. We observe that the RR protocol can operate at higher losses given an increased block size above 4×10^5.

$$\psi = \sqrt{\tau(\omega^2 - 1)}, \tag{B6}$$

where ω is Eve's noise variance, V is the Gaussian classical modulation, and τ is the transmissivity of the channel. Note that the latter are not dependent on the detection efficiency and/or the electronic noise at Bob's laboratory.

APPENDIX C: OTHER RESULTS

We now investigate the behavior of the protocols, in terms of the secret-key rate against losses, assuming either a different successful EC probability, block size, or digitization parameter. For example, we see that changing the p_{ec} from 0.9 to 0.4 affects the performance of both reconciliation directions: in Figs. 5 and 6, we see a considerable drop in terms of the rate, but a minimal improvement in loss tolerance compared to Figs. 1 and 2, respectively.

In Figs. 7 and 8, we doubled the block size to 4×10^5 compared to Figs. 1 and 2, respectively. Here we see an

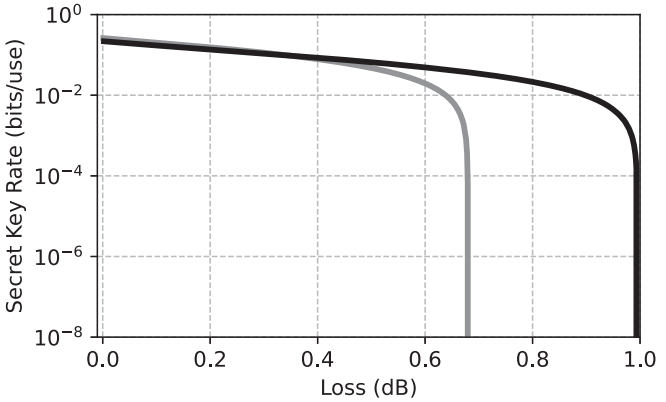


FIG. 12. We plot the secret-key rates from Figs. 1 and 2 in the same panel for clarity. The RR secret-key rate (black line) can surpass the DR one (gray line) in terms of tolerable losses.

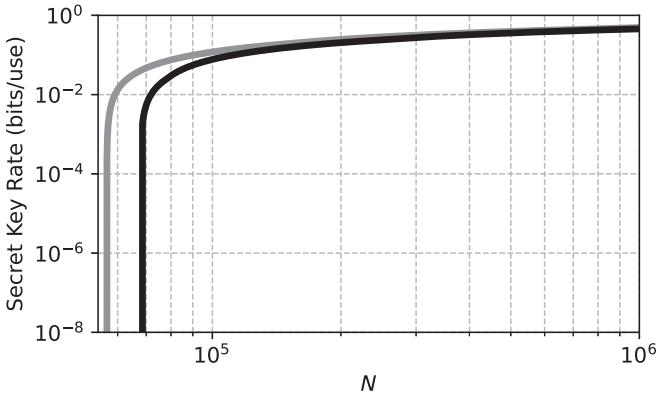


FIG. 13. We plot the secret-key rates from Fig. 3 and 4 in the same frame for a clear comparison. The RR secret-key rate (black line) needs a larger block size compared to the DR one (gray line).

improvement in the performance and loss tolerance: we obtain rates almost in the $3/2$ amount or more of loss in both cases compared to Figs. 1 and 2, respectively. However, the amount of M^*_{sparse} required is almost doubled as expected by the linear dependence of M^*_{sparse} on block size.

When one increases the digitization parameter (here, from $d = 4$ to 6), the performance also increases along with the loss tolerance. We show this tendency in Figs. 9 and 10 for both protocols, respectively. In particular, the tolerable loss is larger than the double in Figs. 1 and 2. In addition, the leakage and M^*_{sparse} increase slightly. This means that by increasing the digitization parameter, we can obtain similar performance as by increasing the block size, avoiding large storage requirements.

Increasing the digitization means that ζ_{digit} approaches 1. In other words, the parties can exploit almost the whole amount of the CV mutual information available to them. This is why CV degrees of freedom are advantageous, especially for small loss. However, the terms $\Delta^{\epsilon_s}_{\text{aep}}$ and $\Delta^{\epsilon_{ec}}_{\text{leak}}$ (also expressed through ζ_{leak}) increase with larger digitization parameters. This will lead to a saturation point for the secret-key rate performance and loss tolerance.

In Fig. 11, we plot the secret-key rate for the RR protocol against the block size for $\text{dB} = 2$. We observe that the RR protocol can tolerate higher losses. Finally, in Fig. 12, we have plotted together the secret-key rate plots from Figs. 1 and 2 for a clearer comparison. We have done the same for Figs. 3 and 4 in Fig. 13.

[1] C. H. Bennett and G. Brassard, Quantum cryptography: Public key distribution and coin tossing, Theor. Comput. Sci. **560**, 7 (2014).

[2] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, Quantum cryptography, Rev. Mod. Phys. **74**, 145 (2002).

[3] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, and M. Peev, The security of practical quantum key distribution, Rev. Mod. Phys. **81**, 1301 (2009).

[4] S. Pirandola, U. L. Andersen, L. Banchi, M. Berta, D. Bunandar, R. Colbeck, D. Englund, T. Gehring, C. Lupo, C. Ottaviani, J. L. Pereira, M. Razavi, J. S. Shaari, M. Tomamichel, V. C. Usenko, G. Vallone, P. Villoresi, and P. Wallden, Advances in quantum cryptography, Adv. Opt. Photon. **12**, 1012 (2020).

[5] V. C. Usenko, A. Acín, R. Alléaume, U. L. Andersen, E. Diamanti, T. Gehring, A. A. E. Hajomer, F. Kanitschar, C. Pacher, S. Pirandola, and V. Pruneri, Continuous-variable quantum communication, Rev. Mod. Phys. **98**, 015003 (2026).

[6] Y. K. Liu and D. Moody, Post-quantum cryptography and the quantum future of cybersecurity, Phys. Rev. Appl. **21**, 040501 (2024).

[7] C. Weedbrook, S. Pirandola, R. Garcia-Patron, N. J. Cerf, T. C. Ralph, J. H. Shapiro, and S. Lloyd, Gaussian quantum information, Rev. Mod. Phys. **84**, 621 (2012).

[8] A. Leverrier, Security of continuous-variable quantum key distribution via a Gaussian de Finetti reduction, Phys. Rev. Lett. **118**, 200501 (2017).

[9] S. Pirandola and P. Papanastasiou, Improved composable key rates for CV-QKD, Phys. Rev. Res. **6**, 023321 (2024).

[10] Y. Zhang, Z. Li, Z. Chen, C. Weedbrook, Y. Zhao, X. Wang, Y. Huang, C. Xu, X. Zhang, Z. Wang, M. Li, X. Zhang, Z. Zheng, B. Chu, X. Gao, N. Meng, W. Cai, Z. Wang, G. Wang, S. Yu, *et al.*, Continuous-variable QKD over 50 km commercial fiber, Quantum Sci. Technol. **4**, 035006 (2019).

[11] T. Wang, P. Huang, Y. Zhou, W. Liu, H. Ma, S. Wang, and G. Zeng, High key rate continuous-variable quantum key distirbution with a real local oscillator, Opt. Express **26**, 2794 (2018).

[12] S. Pirandola, R. Laurenza, C. Ottaviani, and L. Banchi, Fundamental limits of repeaterless quantum communications, Nat. Commun. **8**, 15043 (2017).

[13] F. Grosshans and P. Grangier, Continuous variable quantum cryptography using coherent states, Phys. Rev. Lett. **88**, 057902 (2002).

[14] C. Weedbrook, A. M. Lance, W. P. Bowen, T. Symul, T. C. Ralph, and P. K. Lam, Quantum cryptography without switching, Phys. Rev. Lett. **93**, 170504 (2004).

[15] F. Grosshans and P. Grangier, Reverse reconciliation protocols for quantum cryptography with continuous variables, arXiv:quant-ph/0204127.

[16] R. Filip, Continuous-variable quantum key distribution with noisy coherent states, Phys. Rev. A **77**, 022310 (2008).

[17] S. Pirandola, S. Mancini, S. Lloyd, and S. L. Braunstein, Continuous variable quantum cryptography using two-way quantum communication, Nat. Phys. **4**, 726 (2008).

[18] C. Weedbrook, C. Ottaviani, and S. Pirandola, Two-way quantum cryptography at different wavelengths, Phys. Rev. A **89**, 012309 (2014).

[19] P. Papanastasiou, C. Ottaviani, and S. Pirandola, Gaussian one-way thermal quantum cryptography with finite-size effects, Phys. Rev. A **98**, 032314 (2018).

[20] C. Ottaviani, M. J. Woolley, M. Erementchouk, J. F. Federici, P. Mazumder, S. Pirandola, and C. Weedbrook, Terahertz quantum cryptography, IEEE J. Select. Areas Commun. **38**, 483 (2020).

[21] S. Pirandola, C. Ottaviani, G. Spedalieri, C. Weedbrook, S. L. Braunstein, S. Lloyd, T. Gehring, C. S. Jacobsen, and U. L. Andersen, High-rate measurement-device-independent quantum cryptography, Nat. Photon. **9**, 397 (2015).

[22] Z. Li, Y.-C. Zhang, F. Xu, X. Peng, and H. Guo, Continuous-variable measurement-device-independent quantum key distribution, Phys. Rev. A **89**, 052301 (2014).

[23] C. Ottaviani, G. Spedalieri, S. L. Braunstein, and S. Pirandola, Continuous-variable quantum cryptography with an untrusted relay: Detailed security analysis of the symmetric configuration, Phys. Rev. A **91**, 022320 (2015).

[24] P. Papanastasiou, C. Ottaviani, and S. Pirandola, Finite-size analysis of measurement-device-independent quantum cryptography with continuous variables, Phys. Rev. A **96**, 042332 (2017).

[25] X. Zhang, Y.-C. Zhang, Y. Zhao, X. Wang, S. Yu, and H. Guo, Finite-size analysis of continuous-variable measurement-device-independent quantum key distribution, Phys. Rev. A **96**, 042334 (2017).

[26] C. Lupo, C. Ottaviani, P. Papanastasiou, and S. Pirandola, Parameter estimation with almost no public communication for continuous-variable quantum key distribution, Phys. Rev. Lett. **120**, 220505 (2018).

[27] C. Lupo, C. Ottaviani, P. Papanastasiou, and S. Pirandola, Continuous-variable measurement-device-independent quantum key distribution: Composable security against coherent attacks, Phys. Rev. A **97**, 052327 (2018).

[28] P. Papanastasiou, A. G. Mountogiannakis, and S. Pirandola, Composable security of CV-MDI-QKD with secret key rate and data processing, Sci. Rep. **13**, 11636 (2023).

[29] A. I. Fletcher, C. Harney, M. Ghalaii, P. Papanastasiou, A. Mountogiannakis, G. Spedalieri, A. A. E. Hajomer, T. Gehring, and S. Pirandola, An overview of CV-MDI-QKD, Rep. Prog. Phys. **88**, 084001 (2025).

[30] C. Ottaviani, C. Lupo, R. Laurenza, and S. Pirandola, Modular network for high-rate quantum conferencing, Commun. Phys. **2**, 118 (2019). See, also, High-rate secure quantum conferencing, arXiv:1709.06988v2.

[31] P. Papanastasiou, C. Weedbrook, and S. Pirandola, Continuous-variable quantum key distribution in fast fading channels, Phys. Rev. A **97**, 032311 (2018).

[32] M. Ghalaii, P. Papanastasiou, and S. Pirandola, Composable end-to-end security of Gaussian quantum networks with untrusted relays, npj Quantum Inf. **8**, 105 (2022).

[33] T. Symul, D. J. Alton, S. M. Assad, A. M. Lance, C. Weedbrook, T. C. Ralph, and P. K. Lam, Experimental demonstration of post-selection-based continuous- variable quantum key distribution in the presence of Gaussian noise, Phys. Rev. A **76**, 030303(R) (2007).

[34] K. N. Wilkinson, P. Papanastasiou, C. Ottaviani, T. Gehring, and S. Pirandola, Long-distance continuous-variable measurement-device-independent quantum key distribution with postselection, Phys. Rev. Res. **2**, 033424 (2020).

[35] N. Hosseinidehaj, A. M. Lance, T. Symul, N. Walk, and T. C. Ralph, Finite-size effects in continuous-variable quantum key distribution with Gaussian postselection, Phys. Rev. A **101**, 052335 (2020).

[36] A. Leverrier and P. Grangier, Continuous-variable quantum-key-distribution protocols with a non- Gaussian modulation, Phys. Rev. A **83**, 042312 (2011).

[37] P. Papanastasiou, C. Lupo, C. Weedbrook, and S. Pirandola, Quantum key distribution with phase-encoded coherent states: Asymptotic security analysis in thermal-loss channels, Phys. Rev. A **98**, 012340 (2018).

[38] S. Ghorai, P. Grangier, E. Diamanti, and A. Leverrier, Asymptotic security of continuous-variable quantum key distribution with a discrete modulation, Phys. Rev. X **9**, 021059 (2019).

[39] P. Papanastasiou and S. Pirandola, Continuous-variable quantum cryptography with discrete alphabets: Composable security under collective Gaussian attacks, Phys. Rev. Res. **3**, 013047 (2021).

[40] S. Bäuml, C. Pascual-García, V. Wright, O. Fawzi, and A. Acín, Security of discrete-modulated continuous-variable quantum key distribution, Quantum **8**, 1418 (2024).

[41] A. A. E. Hajomer, F. Kanitschar, N. Jain, M. Hentschel, R. Zhang, N. Lütkenhaus, U. L. Andersen, C. Pacher, and T. Gehring, Experimental composable key distribution using discrete-modulated continuous variable quantum cryptography, Light Sci. Appl. **14**, 255 (2025).

[42] A. A. E. Hajomer, A. Bomhals, C. Bruynsteen, A. Sidhique, O. Kovalenko, I. Derkach, V. C. Usenko, U. L. Andersen, X. Yin, and T. Gehring, Chip-based 16 GBaud continuous-variable quantum key distribution, Optica **13**, 1035 (2026).

[43] A. A. E. Hajomer, C. Bruynsteen, I. Derkach, N. Jain, A. Bomhals, S. Bastiaens, U. L. Andersen, X. Yin, and T. Gehring, Continuous-variable quantum key distribution at 10 GBaud using an integrated photonic-electronic receiver, Optica **11**, 1197 (2024).

[44] J. Aldama, S. Sarmiento, L. Trigo Vidarte, S. Etcheverry, I. López Grande, L. Castelvero, A. Hinojosa, T. Beckerwerth, Y. Piétri, A. Rhouni, E. Diamanti, and V. Pruneri, Integrated InP-based transmitter for continuous-variable quantum key distribution, Opt. Express **33**, 8139 (2025).

[45] Y. Piétri, L. T. Vidarte, M. Schiavon, L. Vivien, P. Grangier, A. Rhouni, and E. Diamanti, Experimental demonstration of continuous-variable quantum key distribution with a silicon photonics integrated receiver, Opt. Quantum **2**, 428 (2024).

[46] L. Li, T. Wang, X. Li, P. Huang, Y. Guo, L. Lu, L. Zhou, and G. Zeng, Continuous-variable quantum key distribution with on-chip light sources, Photon. Res. **11**, 504 (2023).

[47] J. W. Silverstone, D. Bonneau, J. L. O'Brien, and M. G. Thompson, Silicon quantum photonics, IEEE J. Sel. Top. Quant. Electron. **22**, 390 (2016).

[48] R. Roman, J. Zhou, and J. López, On the features and challenges of security and privacy in distributed Internet of things, Comput. Networks **57**, 2266 (2013).

[49] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on Internet-scale IoT exploitations, IEEE Commun. Surv. Tutor. **21**, 2702 (2019).

[50] G. Fitzgibbon and C. Ottaviani, Constrained device performance benchmarking with the implementation of post-quantum cryptography, Cryptography **8**, 21 (2024).

[51] D. Joseph, R. Misoczki, M. Manzano, J. Tricot, F. D. Pinuaga, O. Lacombe, S. Leichenauer, J. Hidary, P. Venables, and R. Hansen, Transitioning organizations to post-quantum cryptography, Nature (London) **605**, 237 (2022).

[52] M. Mosca, Cybersecurity in an era with quantum computers: Will we be ready? IEEE Secur. Priv. **16**, 38 (2018).

[53] L. Garms, T. K. Paraïso, N. Hanley, A. Khalid, C. Rdafferty, J. Grant, J. Newman, A. J. Shields, C. Cid, and M. O'Neill, Experimental integration of quantum key distribution and post-quantum cryptography in a hybrid quantum-safe cryptosystem, Adv. Quantum Technol. **7**, 2300304 (2024).

[54] A. K. Fedorov, Deploying hybrid quantum-secured infrastructure for applications: When quantum and post-quantum can work together, Front. Quantum Sci. Technol. **2**, 1164428 (2023).

[55] J. S. Buruaga, A. Bugler, J. P. Brito, V. Martin, and C. Striecks, Versatile quantum-safe hybrid key exchange and its application to MACsec, EPJ Quantum Technol. **12**, 84 (2025).

[56] X. Wang, Y. Zhang, S. Yu, and H. Guo, High speed error correction for continuous-variable quantum key distribution with multi-edge type LDPC code, Sci. Rep. **8**, 10543 (2018).

[57] M. Milicevic, C. Feng, L. M. Zhang, and P. G. Gulak, Quasi-cyclic multi-edge LDPC codes for long-distance quantum cryptography, npj Quantum Inf. **4**, 21 (2017).

[58] P. Jouguet, S. Kunz-Jacques, A. Leverrier, P. Grangier, and E. Diamanti, Experimental demonstration of long-distance continuous-variable quantum key distribution, Nat. Photon. **7**, 378 (2013).

[59] D. Huang, P. Huang, D. Lin, and G. Zeng, Long-distance continuous-variable quantum key distribution by controlling excess noise, Sci. Rep. **6**, 19201 (2016).

[60] C. Pacher, J. Martinez-Mateo, J. Duhme, T. Gehring, and F. Furrer, Information reconciliation for continuous-variable quantum key distribution using non-binary low-density parity-check codes, arXiv:1602.09140.

[61] A. G. Mountogiannakis, P. Papanastasiou, B. Braverman, and S. Pirandola, Composably secure data processing for Gaussian-modulated continuous-variable quantum key distribution, Phys. Rev. Res. **4**, 013099 (2022).

[62] A. G. Mountogiannakis, P. Papanastasiou, and S. Pirandola, Data postprocessing for the one-way heterodyne protocol under composable finite-size security, Phys. Rev. A **106**, 042606 (2022).

[63] W. Weerasinghe, Practical, high-speed, Gaussian modulated coherent state continuous variable quantum key distribution with real-time post processing, Ph.D. thesis, University of Cambridge, 2024.

[64] M. Tomamichel, J. Martinez-Mateo, C. Pacher, and D. Elkouss, Fundamental finite key limits for one-way information reconciliation in quantum key distribution, Quantum Inf. Proc. **16**, 280 (2017).

[65] D. J. C. Mackay, Good error-correcting codes based on very sparse matrices, IEEE Trans. Inform. Theory **45**, 399 (1999).

[66] M. C. Davey and D. MacKay, Low-density parity check codes over GF(q), IEEE Commun. Lett. **2**, 165 (1998).

[67] J. Martinez-Mateo and D. Elkouss, Efficient reconciliation of continuous variable quantum key distribution with multiplicatively repeated non-binary LDPC codes, EPJ Quantum Technol. **12**, 71 (2025).

[68] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, Internet of things security: A survey, J. Network Comput. Appl. **88**, 10 (2017).

[69] A. Kumar, C. Ottaviani, S. S. Gill, and R. Buyya, Securing the future Internet of things with post-quantum cryptography, Security Privacy **5**, e200 (2022).

[70] A. Shafique, S. A. A. Naqvi, A. Raza, M. Ghalaii, P. Papanastasiou, J. McCann, Q. H. Abbasi, and M. A. Imran, A hybrid encryption framework leveraging quantum and classical cryptography for secure transmission of medical images in IoT-based telemedicine networks, Sci. Rep. **14**, 31054 (2024).

[71] E. Zadobrischi, The concept regarding vehicular communications based on visible light communication and the IoT, Electronics **12**, 1359 (2023).

[72] M. Asif, W. U. Khan, H. M. R. Afzal, J. Nebhen, I. Ullah, A. U. Rehman, and M. K. A. Kabar, Reduced-complexity LDPC decoding for next-generation IoT networks, Wireless Commun. Mobile Comput. **2021**, 2029560 (2021).

[73] J. Tang, H. Wen, H.-H. Song, L. Jiao, and K. Zeng, Sharing secrets via wireless broadcasting: A new efficient physical layer group secret key generation for multiple IoT devices, IEEE Internet Things J. **9**, 15228 (2022).

[74] X.-H. Tian, R. Yang, H.-Y. Liu, P. Fan, J.-N. Zhang, C. Gu, M. Chen, M. Hu, F.-Y. Lu, C. Zhu, Z.-Q. Yin, Z.-J. Yin, M. Yuan, S. Wang, W. Chen, Y.-X. Gong, S.-N. Zhu, and Z. Xie, Experimental demonstration of drone-based quantum key distribution, Phys. Rev. Lett. **133**, 200801 (2024).

[75] O. Elmabrok and M. Razavi, Wireless quantum key distribution in indoor environments, J. Opt. Soc. Am. B **35**, 197 (2018).

[76] S. Pirandola, Limits and security of free-space quantum communications, Phys. Rev. Res. **3**, 013279 (2021).

[77] In a QKD session, we compute the estimators and their variances directly from the experimental data. These are those effectively used for the worst-case estimators in a practical implementation. For theoretical analysis, one employs asymptotic or large-block approximations. Since in this analysis some of the estimators are a sample mean of independent observations, the central limit theorem ensures approximate Gaussianity, and the Berry-Esseen theorem provides a quantitative bound on the deviation from the Gaussian limit, which for the parameters used in this study (see Table II) and block sizes 2.5×10^4–10^6 is less than 10^{-2}–10^{-3}. Although this bound is conservative, results based on Stein's method and modern Gaussian-approximation theory show that practical convergence is typically faster than the worst-case rate suggests [78–80]. This supports the Gaussian modeling adopted in our numerical worst-case analysis.

[78] L. H. Y. Chen, L. Goldstein, and Q.-M. Shao, *Normal Approximation by Stein's Method* (Springer, Berlin, 2011).

[79] M. Tomamichel and V. Y. F. Tan, A tight upper bound for the third-order asymptotics for most discrete memoryless channels, IEEE Trans. Inf. Theory **59**, 7041 (2013).

[80] V. Chernozhukov, D. Chetverikov, and K. Kato, Gaussian approximations and multiplier bootstrap for maxima of sums of high-dimensional random vectors, Ann. Stat. **41**, 2786 (2013).

[81] D. S. Slepian and J. K. Wolf, Noiseless coding of correlated information sources, IEEE Trans. Inf. Theory **19**, 471 (1973).

[82] C. Portmann and R. Renner, Security in quantum cryptography, Rev. Mod. Phys. **94**, 025008 (2022).

[83] http://github.com/softquanta/homCVQKD.

[84] http://github.com/eqclabs/tight_bound_leakage.

[85] https://vikingdocs.york.ac.uk/.