Data post-processing for the one-way heterodyne protocol under composable finite-size security

Alexander G. Mountogiannakis, Panagiotis Papanastasiou, and Stefano Pirandola
Department of Computer Science, University of York, York YO10 5GH, United Kingdom

The performance of a practical continuous-variable (CV) quantum key distribution (QKD) protocol depends significantly, apart from the loss and noise of the quantum channel, on the post-processing steps which lead to the extraction of the final secret key. A critical step is the reconciliation process, especially when one assumes finite-size effects in a composable framework. Here, we focus on the Gaussian-modulated coherent-state protocol with heterodyne detection in a high signal-to-noise ratio regime. We simulate the quantum communication process and we postprocess the output data by applying parameter estimation, error correction (using high-rate, non-binary low-density parity-check codes) and privacy amplification. This allows us to study the performance for practical implementations of the protocol and optimize the parameters connected to the steps above. We also present an associated Python library performing the steps above.

I. INTRODUCTION

Based on physical laws and not on computational assumptions, quantum key distribution (QKD) ensures the creation of long secret keys between two distant authenticated parties, which can be later used for the exchange of symmetrically encrypted secret messages [1]. In particular, the parties can trace any eavesdropper's action on their communication over the intermediate (insecure) quantum channel that links them. According to Heisenberg's principle, any attempt of the eavesdropper to interact with the travelling quantum signals leaves a trace [2]. Through this trace, the parties can quantify the leaked amount of information and compress their exchanged data appropriately, in order to decouple the eavesdropper from the final secret key.

Traditionally, the parties exchange quantum states described by discrete degrees of freedom, such as the polarization of a photon [4]. Such schemes are called discrete-variable (DV) QKD protocols and their security has been studied copiously [1, 5]. More recently, quantum systems described by continuous degrees of freedom have also been studied and continuous-variable (CV) QKD protocols [6, 7] have emerged as alternative to standard schemes. These degrees of freedom are observables such as the position and momentum of the electromagnetic field [1, 8].

A great advantage of CV-QKD is that the current telecommunications infrastructure is capable of handling the preparation, exchange and detection of the corresponding quantum signals. Thus, it provides a cost-effective and practical solution, when compared with DV-QKD. Such protocols also provide high key rates over metropolitan-area distances [9], with values approaching the theoretical limit of the secret key capacity, also known as the repeaterless PLOB bound [10]. Lately, they have surpassed their previous performance in terms of achievable distances, which are now comparable to these of DV-QKD protocols [11, 12].

In practical applications, where the finite-size effects [13] are important and the parties should take into account composable security terms [14, 15], the protocol performance declines. Therefore, the optimization of the protocol parameters becomes an important aspect in CV-QKD [16]. In particular, it is also important to optimize the procedure of data postprocessing, which is made up of various parts: parameter estimation (PE), raw key creation and error correction (EC), and privacy amplification (PA). According to the composable framework, all these steps have associated error parameters that quantify the probability of failure for each process. These parameters are then combined into a final epsilon parameter that identifies the overall level of security provided by the protocol.

In this work, we focus on the heterodyne protocol with Gaussian modulation of coherent states [7] in a high signal-to-noise ratio regime. We simulate the quantum communication process and then postprocess the generated data via PE, EC and PA by means of a dedicated Python library [17]. In this way, we can evaluate the performance of this protocol, when it is deployed in realistic conditions and employed for high-speed quantum-secure communications at relatively short ranges.

This is the summary of the manuscript. In Sec. II, we present the protocol and the calculation of its asymptotic key rate. In Sec. III, after the simulation of the quantum communication, we connect all the relevant parameters describing all the details of the post-processing steps with the composable key rate. In Sec. IV, we present the simulation specifications of the classical postprocessing while, in Sec. V, we comment and illustrate the results of our investigation. Finally, Sec. VI is for conclusions.

II. REVIEW OF CV-QKD WITH HETERODYNE DETECTION

A. Protocol

Alice draws samples from the variable x, which follows a normal distribution with zero mean and variance $\sigma_x^2 =$

$\mu-1$, i.e., described by the density function

$$p(x) = (2\pi\sigma_x^2)^{-\frac{1}{2}} \exp\left[-\frac{x^2}{2\sigma_x^2}\right]. \tag{1}$$

We denote the samples with $[x]_i$, where $i = 1,\ldots,2N$. Then she groups them in instances $[\mathbf{x}]_j = ([Q_x]_j, [P_x]_j) = ([x]_{2j-1}, [x]_{2j})$ for $j = 1\ldots N$ and encodes them in $n_{\rm bks}$ blocks of coherent (signal) states $|\alpha_j\rangle$, where $\alpha_j = ([x]_{2j-1} + {\rm i}[x]_{2i})/2$. We say that the block size is N. Note that we adopt the notation of Ref. [8, Sec.II] for the quadrature operators $(\hat{Q}, \hat{P})$ so that $[\hat{Q}, \hat{P}] = 2{\rm i}$ and the vacuum noise variance is equal to 1.

The coherent states travel to Bob through an optical fiber with length L and loss rate A_L. This is simulated by a thermal-loss channel with transmissivity $T = 10^{-\frac{A_L L}{10}}$ and $\bar{n}$ environmental photons. This is equivalent to assuming a beam splitter with transmissivity T mixing the traveling mode A from Alice with a mode E of the environment in a thermal state with variance $\omega = 2\bar{n}+1$. Then one may assume the dilation of the environmental state into a two-mode squeezed-vacuum (TMSV) state Φ_{Ee} held by the eavesdropper, Eve. This is a zero-mean Gaussian state with covariance matrix (CM)

$$\mathbf{V}_{Ee}(\omega) = \begin{pmatrix} \omega\mathbf{I} & \sqrt{\omega^2-1}\mathbf{Z} \\ \sqrt{\omega^2-1}\mathbf{Z} & \omega\mathbf{I} \end{pmatrix}, \tag{2}$$

where $\mathbf{I} := {\rm diag}\{1,1\}$ and $\mathbf{Z} := {\rm diag}\{1,-1\}$. Note that this is the so-called "entangling cloner" attack; it is the most realistic form of a collective Gaussian attack [18].

Bob then decodes the signal states by applying a heterodyne measurement on the arriving mode B. The heterodyne measurement is performed by mixing B with a vacuum mode through a balanced beam splitter. Then, homodyne measurement is applied to each output of the beam splitter, with respect to a different (conjugate) quadrature. In that way, Bob obtains to output instances $[\mathbf{y}]_j = ([Q_y]_j, [P_y]_j) = ([y]_{2j-1}, [y]_{2j})$ for the jth coherent state that encodes Alice's instances $[\mathbf{x}]_j$ [25].

More precisely, Bob's detectors are characterized by efficiency η and electronic noise $\upsilon_{\rm el}$. As a result, the decoding variable is connected to the encoding one via

$$y = \sqrt{T\eta}x + z, \tag{3}$$

where z is a Gaussian noise variable characterizing Bob's output. It has zero mean and variance equal to

$$\sigma_z^2 = 2 + \upsilon_{\rm el} + \Xi, \tag{4}$$

where $\Xi = \eta T\xi$ is the variance of the channel's noise and

$$\xi := \frac{1-T}{T}(\omega-1), \tag{5}$$

is the channel's excess noise.

B. Asymptotic rate

In the asymptotic regime, where N is large, one may calculate the mutual information between the parties theoretically based on the input-output relation of Eq. (3). The variance of Bob's variable is given by

$$\sigma_y^2 = \eta T\sigma_x^2 + \sigma_z^2, \tag{6}$$

while the corresponding conditional variance on the input x is given by

$$\sigma_{y|x}^2 = \sigma_y^2(\sigma_x^2 = 0) = \sigma_z^2. \tag{7}$$

Because the variables x and y are Gaussian, the mutual information is given by

$$I(\mathbf{x}:\mathbf{y}) = 2I(x:y) = \log_2\left(\frac{\sigma_y^2}{\sigma_z^2}\right) = \log_2\left(1+{\rm SNR}\right), \tag{8}$$

with

$${\rm SNR} = \frac{\sigma_x^2}{\sigma_z^2/(\eta T)}. \tag{9}$$

Asymptotically, the maximum shared information between the parties is quantified by Eq. (8). This is true when the efficiency of the reconciliation between the parties is ideal: In a practical reverse reconciliation scenario, Bob helps Alice's guessing of his outcome by publicly revealing more information than needed. This extra information leads to $\beta I(\mathbf{x}:\mathbf{y})$, where $\beta \in (0,1]$ is known as the reconciliation efficiency.

In line with the definition of collective Gaussian attack, we assume that Eve stores her modes (after Gaussian interaction with the signal modes) into a quantum memory which she can optimally measure at the end of all quantum communication between the parties. The parties are able to quantify the maximum possible amount of leaked information by virtue of the Holevo bound. This is computed from the von Neumann entropies $S(\rho_{E'e})$ and $S(\rho_{E'e|y})$, in turn calculated from the joint CM of Bob and Eve. In particular, we have that

$$\mathbf{V}_{BeE'} = \begin{pmatrix} b\mathbf{I} & \gamma\mathbf{Z} & \theta\mathbf{I} \\ \gamma\mathbf{Z} & \omega\mathbf{I} & \psi\mathbf{Z} \\ \theta\mathbf{I} & \psi\mathbf{Z} & \phi\mathbf{I} \end{pmatrix} \tag{10}$$

with

$$b := \eta T(\mu+\xi) + 1 - T\eta + \upsilon_{\rm el}, \tag{11}$$

$$\gamma := \sqrt{\eta(1-T)(\omega^2-1)}, \tag{12}$$

$$\theta := \sqrt{\eta T(1-T)}(\omega-\mu), \tag{13}$$

$$\psi := \sqrt{T(\omega^2-1)}, \tag{14}$$

$$\phi := T\omega + (1-T)\mu. \tag{15}$$

By tracing out mode B from Eq. (10), we obtain $\mathbf{V}_{eE'}$.

Then, by setting

$$\mathbf{C} = \left(\gamma\mathbf{Z} \;\; \theta\mathbf{I}\right), \tag{16}$$

and applying the formula for the heterodyne measurement [8], we obtain Eve's conditional CM

$$\mathbf{V}_{eE'|\mathbf{y}} = \mathbf{V}_{eE'} - (b+1)^{-1}\mathbf{C}^T\mathbf{C} \tag{17}$$

$$= \begin{pmatrix} \omega\mathbf{I} & \psi\mathbf{Z} \\ \psi\mathbf{Z} & \phi\mathbf{I} \end{pmatrix} - (b+1)^{-1}\begin{pmatrix} \gamma^2\mathbf{I} & \gamma\theta\mathbf{Z} \\ \gamma\theta\mathbf{Z} & \theta^2\mathbf{I} \end{pmatrix}. \tag{18}$$

Then we may write the Holevo information as

$$\chi(E:\mathbf{y}) = S(\rho_{E'e}) - S(\rho_{E'e|\mathbf{y}}) \tag{19}$$

$$= h(\nu_+) + h(\nu_-) - h(\tilde{\nu}_+) - h(\tilde{\nu}_-), \tag{20}$$

where

$$h(\nu) := \frac{\nu+1}{2}\log_2\frac{\nu+1}{2} - \frac{\nu-1}{2}\log_2\frac{\nu-1}{2} \tag{21}$$

and $\{\nu_\pm\}$, $\{\tilde{\nu}_\pm\}$ are the symplectic spectra of $\mathbf{V}_{eE'}$ and $\mathbf{V}_{eE'|\mathbf{y}}$ respectively. Finally, the asymptotic secret key rate will be given by

$$R_{\text{asy}} = \beta I(\mathbf{x}:\mathbf{y}) - \chi(E:\mathbf{y}) \tag{22}$$

$$= R(\beta, \mu, \eta, \upsilon_{\text{el}}, T, \xi). \tag{23}$$

III. COMPOSABLE KEY RATE

In this section, we describe the effects of PE, EC and PA on the final secret key rate in the finite-size regime where these steps cannot be considered ideal but may have outputs that fail to have the desired properties with a small probability, i.e., $\tilde{\epsilon}_{\text{PE}}$, ϵ_{cor} and ϵ_{sec} respectively.

A. Channel parameter estimation

For each block, the parties randomly choose m instances $[\mathbf{x}]_j$ and $[\mathbf{y}]_j$ and broadcast them through the public channel. The parties use the corresponding samples $[x]_i$ and $[y]_i$ from all the blocks assuming a stable channel [26]. Based on these $M = 2mn_{\text{bks}}$ samples they define the maximum likelihood estimators (MLEs)

$$\widehat{T} = \frac{1}{\eta(\sigma_x^2)^2}\left(\widehat{C}_{xy}\right)^2 \tag{24}$$

with

$$\widehat{C}_{xy} = \frac{1}{M}\sum_{k=1}^{M}[x]_k[y]_k \tag{25}$$

and

$$\widehat{\Xi} = \widehat{\sigma_z^2} - \upsilon_{\text{el}} - 2 \;\;\text{for}\;\; \widehat{\sigma_z^2} = \frac{1}{M}\sum_{k=1}^{M}\left([y]_k - \sqrt{\eta\widehat{T}}[x]_k\right)^2. \tag{26}$$

Based on a theoretical analysis as in Ref. [16], one can find worst-case values for the above estimators so as to bound Eve's accessible information. These are given by

$$T_M = \widehat{T} - w\sigma_{\widehat{T}}, \quad \Xi_M = \widehat{\Xi} + w\sigma_{\widehat{\Xi}} \tag{27}$$

with

$$\sigma_{\widehat{T}}^2 = \frac{2}{M}\widehat{T}^2\left(2 + \frac{\widehat{\sigma_z^2}}{\eta\widehat{T}\sigma_x^2}\right), \quad \sigma_{\widehat{\Xi}}^2 = \frac{(\widehat{\sigma_z^2})^2}{M} \tag{28}$$

and

$$w = \sqrt{2}\text{erf}^{-1}(1-\epsilon_{\text{PE}}) \tag{29}$$

where ϵ_{PE} is the failure probability of T_M and Ξ_M to be the worst-case scenario values for bounding Eve's information. The overall failure probability (combining the two events) is

$$2\epsilon_{\text{PE}}(1-\epsilon_{\text{PE}}) + \epsilon_{\text{PE}}^2 \leq 2\epsilon. \tag{30}$$

Taking into consideration the previous parameters, we can derive the asymptotic rate after parameter estimation

$$R_M = \beta I(\mathbf{x}:\mathbf{y})|_{\widehat{T},\widehat{\Xi}} - \chi(E:\mathbf{y})|_{T_M,\Xi_M}. \tag{31}$$

From the formula in Ref. [19, Eq. (8.56)], the mutual information of the variables x and y

$$I(x:y) = \frac{1}{2}\log_2\left[1+\text{SNR}\right] = \frac{1}{2}\log_2\left[(1-\rho^2)^{-1}\right] \tag{32}$$

is connected with their correlation

$$\rho := \frac{\mathbb{E}(xy)}{\sigma_x\sigma_y} = \sqrt{\frac{\text{SNR}}{1+\text{SNR}}}. \tag{33}$$

Therefore, one may derive the estimator for the correlation between the variables by replacing the MLEs of transmissivity and noise in Eq. (9), namely,

$$\widehat{\rho} = \sqrt{\frac{\sigma_x^2}{\sigma_x^2 + \widehat{\sigma}_z^2/(\eta\widehat{T})}}. \tag{34}$$

Note that this is going to be used later in the *a priori* probabilities of Sec. III B.

B. Error correction

Given that M signal states have been processed through PE (m per block), only $n = N - m$ per block are available for secret key extraction. More specifically, before the step of PA, Alice and Bob need to reconcile over their raw data strings ($2nn_{\text{bks}}$ samples), in order to end up with identical strings up to some small error probability ϵ_{cor}. The preprocessing of EC contains the steps of normalization, discretization and splitting. During EC, blocks of data with errors that cannot be corrected get discarded with probability $1-p_{\text{EC}}$. The remaining blocks are combined into a large string, which is used as input to the next step of PA.

1. *Normalization*

Alice and Bob concatenate the $n = N - m$ sample from each block in order to calculate the estimated variancess [27]

$$\widehat{\sigma_x^2} = \frac{1}{n_{\text{ent}}} \sum_{k=1}^{n_{\text{ent}}} [x]_k^2, \quad \widehat{\sigma_y^2} = \frac{1}{n_{\text{ent}}} \sum_{i=1}^{n_{\text{ent}}} [y]_k^2 \tag{35}$$

for $n_{\text{ent}} = 2nn_{\text{bks}}$. Then they divide the values $[x]_i$ by the standard deviation $\widehat{\sigma_x} = \sqrt{\widehat{\sigma_x^2}}$ and the values $[y]_i$ by the other standard deviation $\widehat{\sigma_y} = \sqrt{\widehat{\sigma_y^2}}$, therefore creating the normalized samples $[X]_i$ and $[Y]_i$, following a bivariate normal distribution with CM

$$\Sigma_{XY} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}. \tag{36}$$

In terms of a practical calculation from the data, we use $\widehat{\rho}$ from Eq. (34).

2. *Discretization*

Bob maps each of the samples $[Y]_i$ into a number $\mathsf{l} = 0, \ldots, 2^p - 1$ for p an integer and obtains the corresponding samples $[\mathsf{l}]_i$ according to a one-dimensional lattice with cut-off parameter α and step $\delta = 2\alpha 2^{-p}$. More specifically, he computes δ-size intervals, i.e., bins, $[a_\mathsf{l}, b_\mathsf{l})$ with boundary points given according to

$$a_\mathsf{l} = \begin{cases} -\infty & \text{for } \ \mathsf{l} = 0, \\ \\ -\alpha + \mathsf{l}\delta & \text{for } \ \mathsf{l} > 0, \end{cases} \tag{37}$$

and

$$b_\mathsf{l} = \begin{cases} -\alpha + (\mathsf{l}+1)\delta & \text{for } \mathsf{l} < 2^p - 1, \\ \\ \infty & \quad \text{for } \mathsf{l} = 2^p - 1. \end{cases} \tag{38}$$

Alice computes the conditional probability of the value l given the value X. She then obtains

$$P(\mathsf{l}|X) = \frac{1}{2}\text{erf}\left(\frac{b_\mathsf{l} - \widehat{\rho}X}{\sqrt{2(1-\widehat{\rho}^2)}}\right) - \frac{1}{2}\text{erf}\left(\frac{a_\mathsf{l} - \widehat{\rho}X}{\sqrt{2(1-\widehat{\rho}^2)}}\right). \tag{39}$$

3. *Splitting*

Bob then splits each sampled symbol $[\mathsf{l}]_i$ into top $[\bar{\mathsf{l}}]_i$ and bottom $[\underline{\mathsf{l}}]_i$ symbols. More specifically, he chooses numbers q and d, such that $q + d = p$, and breaks each p-ary symbol l into a q-ary symbol $\bar{\mathsf{l}}$ and d-ary symbol $\underline{\mathsf{l}}$ respectively according to the rule:

$$\mathsf{l} = \bar{\mathsf{l}} 2^d + \underline{\mathsf{l}}. \tag{40}$$

Alice then calculates the probability for a specific top symbol $\bar{\mathsf{l}} = 0, \ldots, 2^q - 1$, given its bottom counterpart $\underline{\mathsf{l}} = 0, \ldots, 2^d - 1$ and the variable X. She then obtains

$$P(\bar{\mathsf{l}}|X\underline{\mathsf{l}}) = \frac{P(\bar{\mathsf{l}}, \underline{\mathsf{l}}|X)}{\sum_{\underline{\mathsf{l}}} P(\bar{\mathsf{l}}, \underline{\mathsf{l}}|X)}, \tag{41}$$

where $P(\bar{\mathsf{l}}, \underline{\mathsf{l}}|X)$ is given by Eq. (39).

4. *LDPC encoding and decoding*

Let us assume the reverse reconciliation scenario, where Alice guesses Bob's sequence. Ideally, Bob's sequence is described by the continuous variables $\mathbf{y}$. Given that Alice knows the variable $\mathbf{x}$ correlated with $\mathbf{y}$ by the quantum channel and that Bob's entropy is $H(\mathbf{y})$, Bob needs to send $H(\mathbf{y}|\mathbf{x})$ bits of information through a public channel, if we wanted Alice's accessible information to be equal to the mutual information

$$I(\mathbf{x} : \mathbf{y}) = H(\mathbf{y}) - H(\mathbf{y}|\mathbf{x}). \tag{42}$$

Note that the previous entropic quantities refer to the average number of bits exchanged per signal state (i.e. including both quadratures). Let us assume the variable l to be the discretized version of Y. After the previous classical post processing, it holds that

$$H(\mathbf{y}) = H(Q_y, P_y) = 2H(y) \geq 2H(Y) \tag{43}$$

$$\geq 2H(\mathsf{l}) = H(Q_\mathsf{l}) + H(P_\mathsf{l}) = H(Q_\mathsf{l}, P_\mathsf{l}) = H(l) \tag{44}$$

where the variables Q_l and P_l correspond to samples with odd and even index respectively and

$$l = Q_\mathsf{l} 2^p + P_\mathsf{l} \tag{45}$$

is a bidirectional mapping. Note that Eq. (43) is true, because Q_y and P_y are independent (the same holds, later, for Q_l and P_l as different samples of an i.i.d. variable). Furthermore, we compare the (differential) entropy of two Gaussian variables, y with variance σ_y^2 and Y with unit variance as the normalized version of y, that is dependent only on the variances of the two variables [19, Th. 17.2.3]. For passing from Eq. (43) toEq. (44) one may use the joint entropy of Y and l [24] and observe that l is a deterministic outcome of Y (while the opposite is not true). The last equation in Eq. (44) holds because the mapping in Eq. (45) is bidirectional [21]. In particular, the parties estimate $H(l)$ through $H(\mathsf{l})$. To increase the accuracy of the estimation result, the parties estimate the previous quantity including all the samples $[\mathsf{l}]_i$ from all the n_{bks} blocks. Then the estimate is given by

$$\hat{H}(\mathsf{l}) = -\sum_{\mathsf{l}} \nu_\mathsf{l} \log_2 \nu_\mathsf{l} \tag{46}$$

where ν_l is the frequency of the value l in the samples $[\mathsf{l}]_i$ from all the n_{bks} blocks. For this estimator the following inequality is true [20]:

$$H(\mathsf{l}) \geq \widehat{H}(\mathsf{l}) - \delta_{\mathrm{ent}} \tag{47}$$

where

$$\delta_{\mathrm{ent}} = \log_2(n_{\mathrm{ent}})\sqrt{\frac{2\log(2/\epsilon_{\mathrm{ent}})}{n_{\mathrm{ent}}}} \tag{48}$$

up to an error probability ϵ_{ent}.

In a realistic situation, Alice is guessing a sequence of discrete symbols and Bob sends information through the public channel equal to $n^{-1}\mathrm{leak}_{\mathrm{EC}} \geq H(\mathbf{y}|\mathbf{x})$. The top samples are sent to Alice through the public channel, encoded by a regular LDPC with code rate R_{code}, constant for any length $2n$. For the LDPC encoding, the $[\bar{\mathsf{l}}]_i$ are considered to be elements of the Galois field $\mathcal{GF}(2^q)$. Bob builds a $c \times 2n$ sparse parity check matrix $\mathbf{H}$ such that $c/(2n) = 1 - R_{\mathrm{code}}$ [16]. He then calculates the syndrome $\mathsf{l}^c_{\mathrm{syn}} = \mathbf{H}\bar{\mathsf{l}}^{2n}$ for each block and sends it to Alice, while the bottom sequence is publicly revealed. In other words, Bob is sending at most

$$(k/(2n))q + d = (1 - R_{\mathrm{code}})q + d = -R_{\mathrm{code}}q + p \tag{49}$$

bits per sample $[\mathsf{l}]_i$. It is clear that d should be as small as possible, yet not negligible, in order for this reconciliation scheme to succeed. This bounds the leakage term per signal state:

$$n^{-1}\mathrm{leak}_{\mathrm{EC}} \leq 2(-R_{\mathrm{code}}q + p). \tag{50}$$

Remark 1 *Note that in the case of an active concatenation of the quadratures, i.e. creating the variable l of Eq. (45), the parties will have to perform error correction on n symbols, which are described by $2p$ bits for each block. This will demand a higher value for q approximately raised to $q' \simeq 2q$. Subsequently, this will increase crucially the requirements for computational resources, in order to achieve the same speed for EC.*

By replacing $H(\mathbf{y})$ with $H(l)$ from Eq. (44) and $H(\mathbf{y}|\mathbf{x})$ with $n^{-1}\mathrm{leak}_{\mathrm{EC}}$ in (42), we obtain that

$$I(\mathbf{x}:\mathbf{y}) \geq H(l) - n^{-1}\mathrm{leak}_{\mathrm{Ec}} := \beta I(\mathbf{x}:\mathbf{y}), \tag{51}$$

where β is the reconciliation efficiency. Finally, we consider the practical calculation of β. We first bound the term

$$H(l) - n^{-1}\mathrm{leak}_{\mathrm{Ec}} \geq 2(\hat{H}(\mathsf{l}) - \delta_{\mathrm{ent}}) - 2(-R_{\mathrm{code}}q + p) \tag{52}$$

considering the estimation of $\hat{H}(\mathsf{l})$ and the bound for the $\mathrm{leak}_{\mathrm{EC}}$. Then we also assume the value of the mutual information for the estimated channel parameters $I(x:y)_{\hat{T},\hat{\Xi}}$. Therefore one obtains the practical reconciliation efficiency

$$\hat{\beta} = 2\frac{\widehat{H}(\mathsf{l}) + R_{\mathrm{code}}q - p - \delta_{\mathrm{ent}}}{I(x:y)|_{\widehat{T},\widehat{\Xi}}}. \tag{53}$$

Remark 2 *The previous equation can be written in terms of the SNR as*

$$\hat{\beta} = \frac{\widehat{H}(\mathsf{l}) + R_{code}q - p - \delta_{ent}}{\frac{1}{2}\log_2(1 + \widehat{SNR})}. \tag{54}$$

This equation is equal to the corresponding one for the homodyne protocol (see [16, Eq. (56) and (76)]). In fact, it returns the same results, given that the SNR is the same for both protocols (different combination of transmissivity, excess noise and classical modulation variance.)

Then one may set $\hat{\beta}I(\mathbf{x}:\mathbf{y})|_{\widehat{T},\widehat{\Xi}} := 2[\widehat{H}(\mathsf{l}) + R_{\mathrm{code}}q - p - \delta_{\mathrm{ent}}]$ in Eq. (31) to obtain

$$R^{\mathrm{EC}}_M = 2[\widehat{H}(\mathsf{l}) + R_{\mathrm{code}}q - p - \delta_{\mathrm{ent}}] - \chi(E:\mathbf{y})|_{T_M,\Xi_M}. \tag{55}$$

We also obtain

$$R_{\mathrm{code}} = \left(\hat{\beta}I(x:y)|_{\widehat{T},\widehat{\Xi}}/2 + p + \delta_{\mathrm{ent}} - \widehat{H}(\mathsf{l})\right)q^{-1} \tag{56}$$

$$= \left((\beta/2)\log_2(1 + \eta\widehat{T}\sigma_x^2/\widehat{\sigma_z^2}) + p + \delta_{\mathrm{ent}} - \widehat{H}(\mathsf{l})\right)q^{-1}. \tag{57}$$

Alice then uses the probabilities of Eq. (41) to initialize a sum-product algorithm [16] with a maximum number of iterations $\mathrm{iter}_{\mathrm{max}}$. During every iteration, the algorithm finds a sequence $\hat{\mathsf{l}}^{2n}$ that is optimal for the given likelihood, calculates its syndrome and compares it with $\mathsf{l}^c_{\mathrm{sd}}$. If the syndromes are equal, the specific block qualifies for the verification step. If they are not equal, the algorithm continues to the next iteration. In case the maximum number of iterations $\mathrm{iter}_{\mathrm{max}}$ is exceeded, the given sequence is discarded, along with its associated bottom counterpart.

5. *Verification*

The strings $\hat{\mathsf{l}}^{2n}$ and $\bar{\mathsf{l}}^{2n}$ with the same syndrome are turned into binary strings $\hat{\mathsf{l}}^{2n}_{\mathrm{bin}}$ and $\bar{\mathsf{l}}^{2n}_{\mathrm{bin}}$ respectively over which the parties calculate hashes of $\lceil -\log_2 \epsilon_{\mathrm{cor}} \rceil$ bits (For more details on the calculation of the hashes see Ref. [16]). The parties check their hashes and if, they are equal, they are certain that their sequences agree with a probability $1 - \epsilon_{\mathrm{cor}}$ for a very small ϵ_{cor}. Then, they concatenate the binary version of the bottom sequence $\underline{\mathsf{l}}^{2n}_{\mathrm{bin}}$ to $\hat{\mathsf{l}}^{2n}_{\mathrm{bin}}$ and $\bar{\mathsf{l}}^{2n}_{\mathrm{bin}}$, creating the sequences

$$\widehat{S} = \hat{\mathsf{l}}^{2n}_{\mathrm{bin}}\underline{\mathsf{l}}^{2n}_{\mathrm{bin}} \quad \text{and} \quad S = \mathsf{l}^{2n}_{\mathrm{bin}}\underline{\mathsf{l}}^{2n}_{\mathrm{bin}}. \tag{58}$$

If the hashes do not agree, the strings $\hat{\mathsf{l}}^{2n}_{\mathrm{bin}}$, $\bar{\mathsf{l}}^{2n}_{\mathrm{bin}}$ and $\underline{\mathsf{l}}^{2n}$ are discarded. From the ratio of the sequences that pass to the PA over the total number n_{bks} of sequences, one calculates the probability p_{EC} of EC.

C. Privacy amplification

Privacy amplification is the final step that creates the secret key from the raw shared data. The parties start with two different sequences of n_{bks} blocks, each block with $2N$ samples. After postprocessing, these are reduced to two indistinguishable (with probability $1-\epsilon_{\mathrm{EC}}$) binary sequences, that consist of $p_{\mathrm{EC}}n_{\mathrm{bks}}$ blocks, each block carrying $2np$ bits (see Eq. (58)).

The parties then decide to further compress their data in order to prevent Eve from having any knowledge of their bit sequences. To do so, they concatenate their previous sequences into large ones $\mathbf{S} \simeq \widehat{\mathbf{S}}$ containing $\tilde{n} := 2p_{\mathrm{EC}}n_{\mathrm{bks}}np$ bits and compress them using a universal hashing: They apply a Toeplitz matrix $\mathbf{T}_{r,\tilde{n}}$ to their sequences (see more details in Ref. [16]) in order to extract the secret key

$$\mathbf{K} = \mathbf{T}_{r,\tilde{n}}\mathbf{S} \simeq \mathbf{T}_{r,\tilde{n}}\widehat{\mathbf{S}} \tag{59}$$

which has length $r = p_{\mathrm{EC}}n_{\mathrm{bks}}n\tilde{R}$ where $\tilde{R}$ is the composable key rate. The latter takes into account of any small distance of the practical protocol from an ideal one. More specifically, each of the processes of PE and EC have small failure probabilities $\tilde{\epsilon}_{\mathrm{PE}}$ and ϵ_{cor}.

In $\tilde{\epsilon}_{\mathrm{PE}} = 2\epsilon_{\mathrm{PE}} + \epsilon_{\mathrm{ent}}$, we include that overall failure probability of PE (see Eq. Eq. (30)) and the failure probability of bounding the Bob's variable entropy ϵ_{ent} (see Eq. (48)). The PA procedure is characterized by the ϵ-secrecy parameter, which quantifies the potential failure to completely exclude Eve from obtaining information about the key with probability ϵ_{sec}. The latter can be broken in two separate parameters: the smoothing parameter ϵ_{s} and the hashing parameter ϵ_{h}, which yield $\epsilon_{\mathrm{sec}} = \epsilon_{\mathrm{s}} + \epsilon_{\mathrm{h}}$. The composition of all these parameters (see also Eq. (A27)) defines the security parameter of the protocol

$$\epsilon = p_{\mathrm{EC}}(2\epsilon_{PE} + \epsilon_{\mathrm{ent}}) + \epsilon_{\mathrm{cor}} + \epsilon_{\mathrm{sec}}, \tag{60}$$

with typical choice $\epsilon_{\mathrm{s}} = \epsilon_{\mathrm{h}} = \epsilon_{\mathrm{cor}} = \epsilon_{\mathrm{PE}} = \epsilon_{\mathrm{ent}} = 2^{-32} \simeq 2.3\times 10^{-10}$, so that for any value of p_{EC} we have $\epsilon \lesssim 10^{-9}$. Finally, the secret key rate of the protocol, in terms of bits per channel use, takes the form [14]

$$R = \frac{np_{\mathrm{EC}}}{N}\tilde{R},\ \ \tilde{R} := \left(R_M^{\mathrm{EC}} - \frac{\Delta_{\mathrm{AEP}}}{\sqrt{n}} + \frac{\Theta}{n}\right), \tag{61}$$

where R_M^{EC} is the rate of Eq. (55) where β is replaced by Eq. (53) and the extra terms are [14, 15]

$$\Delta_{\mathrm{AEP}} := 4\log_2\left(2^p+2\right)\sqrt{\log_2\left(\frac{18}{p_{\mathrm{EC}}^2\varepsilon_{\mathrm{s}}^4}\right)}, \tag{62}$$

$$\Theta := \log_2[p_{\mathrm{EC}}(1-\varepsilon_{\mathrm{s}}^2/3)] + 2\log_2\sqrt{2}\varepsilon_{\mathrm{h}}. \tag{63}$$

Note that the discretization bits p appear in Δ_{AEP} providing a total dimension of 2^{2p} per symbol (see Appendix A). One may also compare the previous rate with the corresponding theoretical rate

$$R_{\mathrm{theo}} = \frac{np_{\mathrm{EC}}}{N}R^\star,\quad R^\star := \left(\bar{R}_M - \frac{\Delta_{\mathrm{AEP}}}{\sqrt{n}} + \frac{\Theta}{n}\right) \tag{64}$$

where $\bar{R}_M$ has been computed based on the initial values of the channel parameters used to produce the simulation data. In fact, one may replace in Eq. (31) the mean value of the estimators and obtain

$$\bar{R}_M = \beta I(\mathbf{x}:\mathbf{y})|_{\bar{T},\bar{\Xi}} - \chi(E:\mathbf{y})|_{\bar{T}_M,\bar{\Xi}_M}, \tag{65}$$

where the following substitutions have been made:

$$\widehat{T} \leftarrow \bar{T} := \mathbb{E}(T) \simeq T + \mathcal{O}(1/M), \tag{66}$$

$$\widehat{\Xi} \leftarrow \bar{\Xi} := \mathbb{E}(\widehat{\Xi}) \simeq \Xi \tag{67}$$

and

$$T_M \leftarrow \bar{T}_M := T - w\sigma_T \tag{68}$$

$$\Xi_M \leftarrow \Xi_M := \Xi + w\sigma_\Xi \tag{69}$$

with

$$\sigma_T^2 = \frac{2}{M}T^2\left(2 + \frac{\sigma_z^2}{\eta T\sigma_x^2}\right),\quad \sigma_\Xi^2 = \frac{(\sigma_z^2)^2}{M}. \tag{70}$$

On the other hand, in the previous rate the parameters p_{EC} and β have been calculated through the simulation; in fact, they are known after EC (see Fig. 1).

IV. SIMULATION

Here, we summarize the steps of the heterodyne protocol simulation taking into account of the finite-size effects in a composable framework. Our approach follows steps similar to those of the homodyne protocol in Ref. [16]. Despite the fact that the simulation steps of the two protocols are quite similar, we want here to present a summary for the heterodyne protocol for the sake of completeness. We also have the opportunity to clarify some differences between the two simulations because of the use of different formulas.

Preparation: Alice encodes $2Nn_{\mathrm{bks}}$ samples $[x]_i$ of the generic variable $x \sim \mathcal{N}(0, \mu-1)$ on the two conjugate quadratures of Nn_{bks} coherent states. In particular, the samples with odd index will be encoded in the Q-quadrature of the Nn_{bks} coherent states, while those with even index will be encoded in the P-quadrature of the coherent states.

Measurement: During the decoding step, Bob obtains $2Nn_{\mathrm{bks}}$ output samples $[y]_i$ of $y = \sqrt{\eta T}x + z$ according to the propagation of the channel and the projection based on the heterodyne measurement.

Public declaration: Bob chooses an average of m instances from each block and reveals them and their positions through the public channel. In each block, an average of n instances are left for key generation.

Estimators: The parties use $M = 2mn_{\text{bks}}$ samples to define MLEs $\widehat{T}$ and $\widehat{\Xi}$ for T and Ξ, respectively. Then, by setting a PE error ϵ_{EP}, they can calculate the values T_M and Ξ_M for the channel parameters. These values constitute the worst-case scenario assumption on the collected data with probability $1-\epsilon_{\text{EP}}$.

Normalization: Alice and Bob normalize the samples $[x]_i$ and $[y]_i$ dividing them by their practical standard deviations $\widehat{\sigma_x}$ and $\widehat{\sigma_y}$, creating the samples $[X]_i$ and $[Y]_i$, respectively. These variables now follow a standard normal distribution.

Discretization: Bob maps every sample $[Y]_i$ into a number $\mathsf{l} = 0, \ldots, 2^p - 1$. To do so, he creates a one-dimensional lattice for the values of the standard normal distribution with cut-off parameter α and step $\delta = \alpha 2^{1-p}$. Alice calculates the conditional probabilities $P(\mathsf{l}|X)$.

Splitting: Bob splits $[\mathsf{l}]_i$ into two samples, $[\bar{\mathsf{l}}]_i$ and $[\underline{\mathsf{l}}]_i$. In fact, he derives a top q-ary symbol and a bottom d-ary symbol from l according to

$$\mathsf{l} = \bar{\mathsf{l}} 2^d + \underline{\mathsf{l}}. \tag{71}$$

Finally, Alice computes the *a priori* probabilities $P(\bar{\mathsf{l}}|X, \underline{\mathsf{l}})$.

LDPC encoding: From the estimated SNR and the practical Shannon entropy $\widehat{H}(\mathsf{l})$, the parties calculate the rate of the LDPC code according to Eq. (56). Bob then calculates the $c \times 2n$ parity-check matrix $\mathbf{H}$, for $c = 2n(1 - R_{\text{code}})$, with entries in $\mathcal{GF}(p)$, and computes the syndrome $\mathsf{l}^c_{\text{sd}} = \mathbf{H}\bar{\mathsf{l}}^{2n}$. Bob sends the syndromes and the bottom sequences $\underline{\mathsf{l}}^{2n}$ to Alice through the public channel for every block.

LDPC decoding: Alice updates the likelihood function (initially equal to the product of the *a priori* probabilities, see Eq. (41)) using a sum-product algorithm. This update takes place with respect to the syndrome l^c_{sd}. After every iteration of the algorithm, the output likelihood function becomes the input for the next iteration. At the same time, Alice finds $\hat{\mathsf{l}}^{2n}$ that maximizes the updated likelihood function. She then compares the syndrome of $\hat{\mathsf{l}}^{2n}$ with l^c_{sd} and, if they are equal, the algorithm terminates and gives as output the string $\hat{\mathsf{l}}^{2n}$, i.e. Alice's guess for $\bar{\mathsf{l}}^{2n}$. Otherwise, the algorithm continues to the next iteration until a maximum number of iterations iter_{max} is reached. If the algorithm is not able to determine a guess after iter_{max}, the given block is discarded and does not participate in the final key.

Verification: Alice's guess $\hat{\mathsf{l}}^{2n}$ and Bob's sequence $\bar{\mathsf{l}}^{2n}$ are converted into binary sequences $\hat{\mathsf{l}}^{2n}_{\text{bin}}$ and $\bar{\mathsf{l}}^{2n}_{\text{bin}}$ respectively. Then both parties compute hashes of $\lceil -\log_2 \epsilon_{\text{cor}} \rceil$ bits over their sequences. Bob discloses his hash and Alice compares it with hers. In case they are identical, they concatenate their string with the binary version of the bottom string $\underline{\mathsf{l}}_{\text{bin}}$ and obtain the strings

$$\widehat{S} := \hat{\mathsf{l}}^{2n}_{\text{bin}} \underline{\mathsf{l}}^{2n}_{\text{bin}} \simeq S := \bar{\mathsf{l}}^{2n}_{\text{bin}} \underline{\mathsf{l}}^{2n}_{\text{bin}} \tag{72}$$

respectively, which are further promoted to the privacy amplification step (PA). Otherwise, the strings $\hat{\mathsf{l}}^{2n}_{\text{bin}}$, $\bar{\mathsf{l}}^{2n}_{\text{bin}}$ and $\underline{\mathsf{l}}^{2n}_{\text{bin}}$ are discarded and the given block does not participate in the final key.

Privacy amplification: The parties concatenate the strings $\widehat{S}$ and S from every block into long binary sequences $\widehat{\mathbf{S}}$ and $\mathbf{S}$ of $\tilde{n} = p_{\text{EC}} n_{\text{bks}} n 2p$ bits. Given a level of secrecy ϵ_{sec}, the parties calculate the composable rate $\tilde{R}$ and compress the sequences $\widehat{\mathbf{S}} \simeq \mathbf{S}$ with the use of a Toeplitz matrix $\mathbf{T}_{r,\tilde{n}}$ into the final secret key $\mathbf{K}$ of length $r := p_{\text{EC}} n_{\text{bks}} n \tilde{R}$.

V. RESULTS

Since the protocol of this paper is better suited to short-range distances, only distances up to 5km are examined. Consequently, the SNR of the performed simulations is relatively high and takes values from ~ 6 to 10. Two features are considered essential in achieving a positive composable secret key rate R. The first is having a sufficient number of total key generation states nn_{bks}. The second is the choice of the reconciliation efficiency, which must be large enough to obtain a high rate but small enough to comfortably execute error correction. A large number of total key generation states will also lead to a better value for the reconciliation efficiency. This connection is provided by the presence of δ_{ent} term in Eq. (53), which becomes smaller as the number of states increases.

A demonstration of sample parameters, that achieve a positive composable key rate and how this rate varies, according to changes in the block size N and the number of blocks n_{bks}, is shown in Figs. 1 and 2 respectively. Alice's signal variance μ is tuned so as to produce a rather high signal-to-noise ratio (SNR = 10). It is observed in Fig. 1 that a block size of at least 2×10^5 is needed. Additionally, Fig. 2 shows that it is possible to yield higher key rates with fewer total states, if an adequately large block size N is specified.

Fig. 3 portrays the composable rate R versus distance L, expressed in km of standard optical fiber. Here, the SNR varies from 5.732 to 6.887. For this simulation, the discretization bits value was set to $p = 6$, in order to reach farther distances. A higher value for p would severely limit the protocol's ability to achieve a positive R at distances larger than 3 km.

Fig. 4 presents an estimate of the maximum tolerable excess noise ξ. The variables used here produce an SNR

Parameter	Value (Fig. 1)	Value (Fig. 2)	Value (Fig. 3)	Value (Fig. 4)	Value (Fig. 5)
L	3	3	variable	4	5
A	0.2	0.2	0.2	0.2	0.2
ξ	0.01	0.01	0.01	variable	0.01
η	0.85	0.85	0.8	0.85	0.85
υ_{el}	0.1	0.1	0.1	0.05	0.1
n_{bks}	50	variable	50	50	50
N	variable	3×10^5	3.6×10^5	4.5×10^5	4×10^5
M	$0.1 n_{\mathrm{bks}} N$	$0.1 n_{\mathrm{bks}} N$	$0.1 n_{\mathrm{bks}} N$	$0.1 n_{\mathrm{bks}} N$	$0.1 n_{\mathrm{bks}} N$
p	7	7	6	6	variable
q	4	4	4	4	4
α	7	7	7	7	7
$\mathrm{iter}_{\mathrm{max}}$	100	100	150	100	150
$\varepsilon_{\mathrm{PE,\ s,\ h,\ corr}}$	2^{-32}	2^{-32}	2^{-32}	2^{-32}	2^{-32}
μ	≈ 29.46	≈ 29.46	20	25	variable

TABLE I: The input parameters for the simulations.

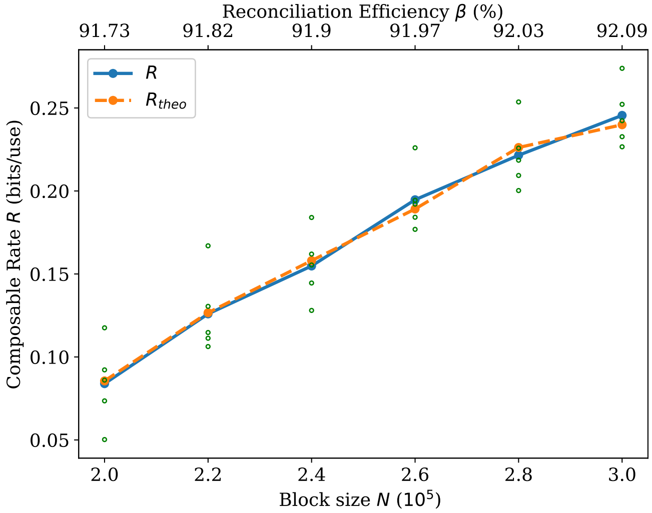

FIG. 1: Composable secret key rate R (bits/use) versus the block size N for SNR $= 10$. We compare the rate of Eq. (61) from five simulations (green points) and their average (blue line) with the theoretical rate R_{theo} in Sec. III C (orange line), where the theoretical guesses for $\tilde{\beta}$ and $\tilde{p}_{\mathrm{EC}}$ are chosen compatibly with the simulations. For every simulation, $\tilde{p}_{\mathrm{EC}} = p_{\mathrm{EC}}$ has been set. All simulations have achieved $p_{\mathrm{EC}} \geq 0.9$. The step of N is 20000. The values of the reconciliation efficiency β are shown on the top axis and are chosen so as to produce $R_{\mathrm{code}} \approx 0.846$. See Table I for the list of input parameters used in the simulations.

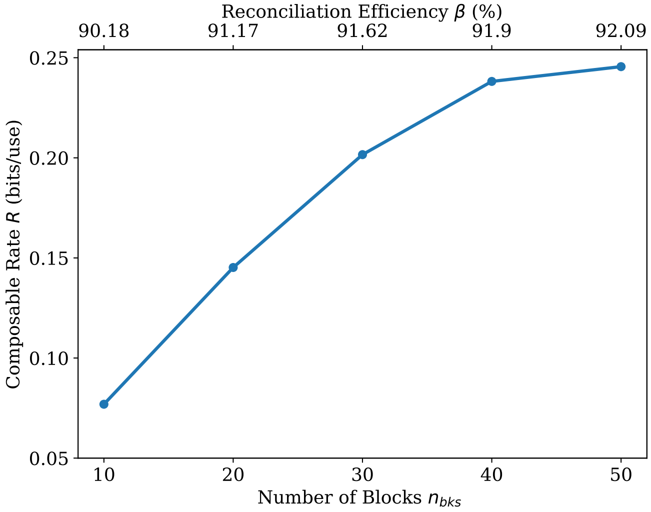

FIG. 2: Composable secret key rate R (bits/use) versus the number of blocks n_{bks} for SNR $= 10$. The step of n_{bks} is 10. The individual block size is fixed and equal to $N = 3 \times 10^5$. Every point represents the average value of R, which is obtained after 5 simulations. All simulations have achieved $p_{\mathrm{EC}} \geq 0.9$. The values of the reconciliation efficiency β are shown on the top axis and are chosen so as to produce $R_{\mathrm{code}} \approx 0.846$. See Table I for the list of input parameters used in the simulations.

of somewhat above 8. While the decrease of the SNR is fairly small as the excess noise increases, the composable rate declines rapidly. In addition, the reconciliation efficiencies used here are in the range of 88.23 - 88.71. Such values provide efficient error correction but are not ideal for attaining a positive rate in the composable framework. Therefore, to achieve a positive rate at $\xi = 0.05$, a large block size ($N = 450000$) has to be used.

Fig. 5 describes the behaviour of the key rate against different SNR values, when the noise terms are fixed and the modulation variance is variable. If the same code rate is used, lower values of p (at a fixed $q = 4$), return higher rates for the corresponding SNR. It is possible for a higher p value to yield a better composable rate than a smaller p, given that a larger code rate, and therefore larger reconciliation efficiency, is employed. An example is given by cases ‘a’ and ‘b’ of SNR $= 9$, whose code rates and reconciliation efficiencies are shown in Ta-

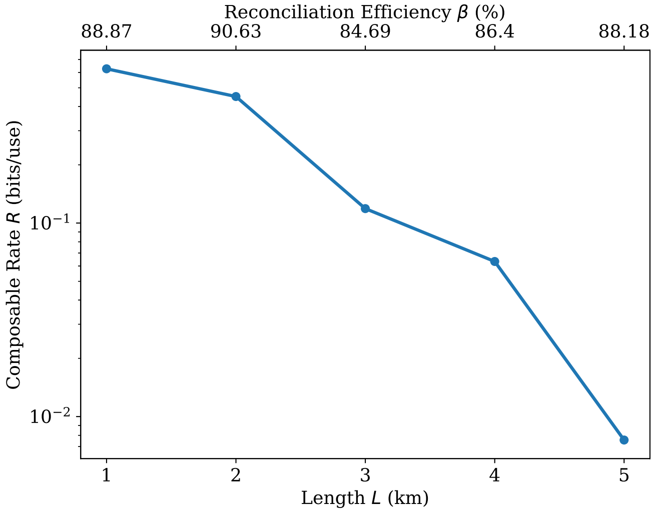

FIG. 3: Composable secret key rate R (bits/use) versus the channel length L (km). Here we use $N = 3.6 \times 10^5$. Every point represents the average value of R, which is obtained after 5 simulations. All simulations have achieved $p_{\mathrm{EC}} \geq 0.9$. The values of the reconciliation efficiency β are shown on the top axis. Other parameters are taken as in Table I.

SNR	$\beta_{p=6}$	$\beta_{p=7}$	$\beta_{p=8}$	R_{code}	d_c
6	0.8651			0.75	8
7	0.8836			0.777	9
8	0.8924	0.8910		0.8	10
9_{a}	0.8953	0.8940		0.818	11
9_{b}			0.9301	0.833	12
10	0.9244	0.9231	0.9229	0.846	13

TABLE II: The chosen reconciliation efficiency β for each SNR of Fig. 5, together with its respective code rate R_{code} and the row weight d_c of the LDPC code. A missing value for the reconciliation efficiency implies that the returned composable key rate will most likely be negative under the specified values. The column weight d_v remains constant and equal to 2 for all simulations.

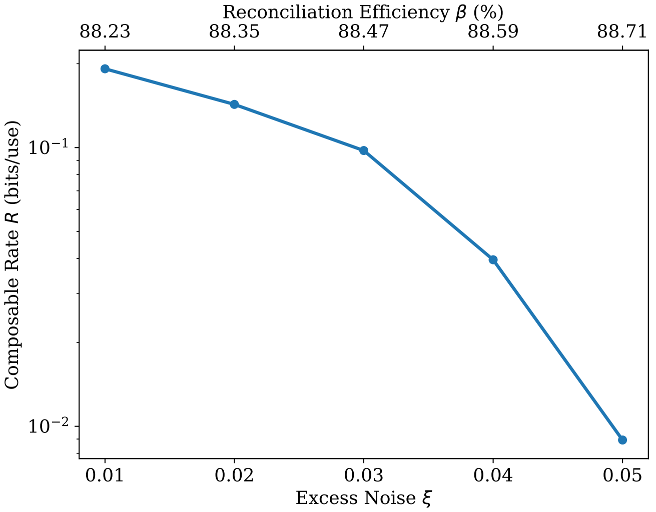

FIG. 4: Composable secret key rate R (bits/use) versus the excess noise ξ. Every point represents the average value of R, which is obtained after 5 simulations. Here we use $N = 4.5 \times 10^5$ and $n_{\mathrm{bks}} = 50$. The values of the reconciliation efficiency β for the heterodyne protocol simulations are chosen so as to produce $R_{\mathrm{code}} \approx 0.8$. Other parameters are taken as in Table I.

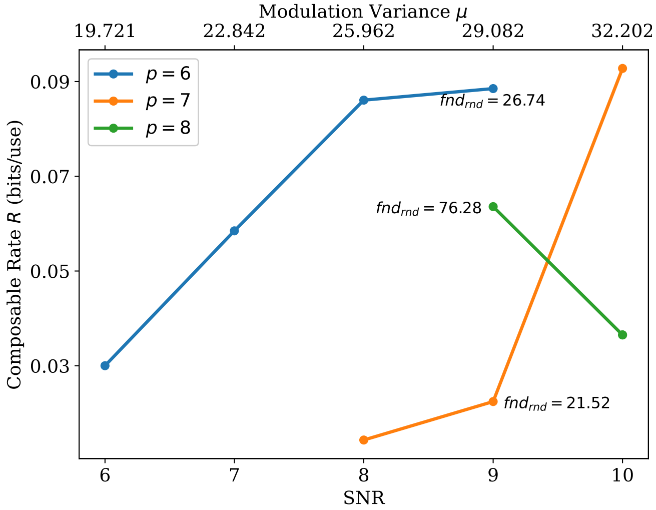

FIG. 5: Composable secret key rate R versus SNR for discretization bits $p = 6$, $p = 7$ and $p = 8$. The chosen reconciliation efficiency β for each value of the SNR is shown in Table II. Every point represents the average value of R, which is obtained after 5 simulations. For SNR $= 9$, the average number of iterations $\mathrm{fnd}_{\mathrm{rnd}}$ needed to decode and verify a block is displayed for every point next to their respective points. The signal variance μ that was used to achieve the respective SNR is displayed on the top axis with an accuracy of 3 decimal digits. Other parameters are chosen as in Table I.

ble II. A combination of $p = 8$ and $\beta = 0.9301$ beats the combination of $p = 7$ and $\beta = 0.894$ in terms of the composable rate by a fairly large margin. However, the trade-off here is that the EC stage of the former combination requires plenty more iteration rounds, making the procedure more computationally expensive. Furthermore, for certain code rates, a minimum value for p is required. Such an occasion is the 'b' case of SNR $= 9$, where error correction can only be achieved for $p = 8$. Smaller values for p would not be able to achieve error correction and, consequently, a positive rate.

VI. CONCLUSION

In this work, we completely characterized the post-processing of data generated from a numerical simulation of the CV-QKD protocol based on Gaussian modulation of coherent states and heterodyne detection. In particular, we designed the data post-processing accounting for the various composable finite-size terms arising from a realistic representation of the protocol. Correspondingly, we provided a Python library for simulation, optimization and data post-processing specifically tailored for the considered heterodyne protocol.

Acknowledgements.– A. M. was supported by the EPSRC via a Doctoral Training Partnership (Grant No. EP/R513386/1). P.P. was supported by the EPSRC via the UK Quantum Communications Hub (Grant No. EP/T001011/1).

[1] 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**, 1012–1236 (2020).

[2] J. Park, "The concept of transition in quantum mechanics," Foundations of Physics **1** 23–33 (1970).

[3] W. Wootters, and W. Zurek, "A Single quantum cannot be cloned," Nature **299**, 802–803 (1982).

[4] C. H. Bennett and G. Brassard, "Quantum cryptography: public key distribution and coin tossing," in Proceedings of the International Conference on Computers, Systems & Signal Processing, Bangalore, India, December 1984, pp. 175–179.

[5] Y. Cao, Y. Zhao, Q. Wang, J. Zhang, S. X. Ng and L. Hanzo, "The Evolution of Quantum Key Distribution Networks: On the Road to the Qinternet," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2022.3144219.

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

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

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

[9] 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**, 397-402 (2015).

[10] S. Pirandola, R. Laurenza, C. Ottaviani, and L. Banchi, "Fundamental Limits of Repeaterless Quantum Communications," Nat. Commun. **8**, 15043 (2017).

[11] Y. Zhang, Z. Chen, S. Pirandola, X. Wang, C. Zhou, B. Chu, Y. Zhao, B. Xu, S. Yu, and H. Guo, "Long-Distance Continuous-Variable Quantum Key Distribution over 202.81 km of Fiber," Phys. Rev. Lett. **125**, 010502 (2020).

[12] C. Zhou, X. Wang, Y. Zhang, Z. Zhang, S. Yu, and H. Guo, "Continuous-Variable Quantum Key Distribution with Rateless Reconciliation Protocol," Phys. Rev. Applied **12**, 054013 (2019).

[13] L. Ruppert, V. C. Usenko, and R. Filip, "Long-distance continuous-variable quantum key distribution with efficient channel estimation", Phys. Rev. A **90**, 062310 (2014).

[14] S. Pirandola, "Limits and Security of Free-Space Quantum Communications," Phys. Rev. Res. **3**, 013279 (2021).

[15] S. Pirandola, "Composable security for continuous variable quantum key distribution: Trust levels and practical key rates in wired and wireless networks," Phys. Rev. Res. **3**, 043014 (2021).

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

[17] Available at https://github.com/softquanta/hetCVQKD.

[18] S. Pirandola, S. L. Braunstein, and S. Lloyd, "Characterization of Collective Gaussian Attacks and Security of Coherent-State Quantum Cryptography," Phys. Rev. Lett. **101**, 200504 (2008).

[19] T. M Cover and J. A. Thomas, "Elements of information theory," (Wiley, 2012).

[20] A. Antos and I. Kontoyiannis, "Convergence Properties of Functional Estimates for Discrete Distributions," Random Structures & Algorithms **19**, 163 (2001).

[21] F. Cicalese, L. Gargano, and U. Vaccaro, "Bounds on the Entropy of a Function of a Random Variable and their Applications," arXiv:1712.07906.

[22] M. Tomamichel, "Quantum Information processing with finite resources," arXiv:1504.00233.

[23] C. Portmann, and R. Renner, " Cryptographic security of quantum key distribution," arXiv:1409.3525v1.

[24] For a function f applied on a random variable X we have

$$\begin{aligned} H(X, f(X)) &= H(X) + \cancelto{0}{H(f(X)|X)} \\ &= H(f(X)) + H(X|f(X)) \end{aligned} \tag{73}$$

The uncertainty for $f(X)$ given X is vanishing which allows for the inequality

$$H(X) \geq H(f(X))$$

to be valid.

[25] In the entanglement based (EB) representation EB, P_y is anti-correlated with P_x but with the same absolute correlation value as the Q-quadrature. As one may see from Eq. (32), the mutual information is not affected by the sign of the correlations. In this case, the parties can change the sign of P_y so that the two quadratures can form a single variable x for Alice (and y for Bob) with the same properties.

[26] We assume that experimentally the coherent state preparation can be done quite fast. In this regime, in the time interval Δt it is feasible to be produced Nn_{bks} states while the transmissivity of the channel can still be considered constant.

[27] We assume here that the variables x and y have a zero mean value. Alternatively, the parties subtract the mean value $\bar{x}$ and $\bar{y}$ of x and y respectively from their instances to create updated centered variables $x \leftarrow x - \bar{x}$ and $y \leftarrow y - \bar{y}$. Then the formulas for estimating the variance keep the same form as in Eq. (35).

Appendix A: Virtual concatenation of the conjugate quadrature variables

What we present here is a review and direct adaptation of the theory developed in Appendix G of Ref. [14]. Let us assume Bob's measurement variables are $\mathbf{y} = (Q_y, P_y)$. Bob maps these variables to $\mathbf{l} = (Q_\mathsf{l}, P_\mathsf{l})$ via analog-to-digital conversion (ADC). Then, the output classical-quantum state (CQ) of Alice, Bob and Eve, after the collective attack will be given by a state in a tensor product form $\rho^{\otimes n}$, where the single copy state will be given by

$$\rho = \sum_{\mathbf{k},\mathbf{l}} p(\mathbf{k},\mathbf{l}) |\mathbf{k}\rangle_{R_A}\langle\mathbf{k}| \otimes |\mathbf{l}\rangle_{R_B}\langle\mathbf{l}| \otimes \rho_E(\mathbf{k},\mathbf{l})$$

where R_A and R_B are Alice's and Bob's classical raw-key registers, $\mathbf{k} = (Q_k, P_k)$ is the corresponding discretized version of Alice's encoding variable and $p(\mathbf{k},\mathbf{l})$ is the joint probability of the discretized variables.

The tensor product state can be then written as

$$\begin{aligned}\rho^{\otimes n} &= \sum_{\mathbf{k}^n,\mathbf{l}^n} p(\mathbf{k}^n,\mathbf{l}^n) |\mathbf{k}^n\rangle_{R_A^n}\langle\mathbf{k}^n| \otimes |\mathbf{l}^n\rangle_{R_B^n}\langle\mathbf{l}^n| \otimes \rho_E^{\otimes n}(\mathbf{k}^n,\mathbf{l}^n) \\ &= \sum_{\mathsf{k}^{2n},\mathsf{l}^{2n}} p(\mathsf{k}^{2n},\mathsf{l}^{2n}) |\mathsf{k}^{2n}\rangle_{R_A^n}\langle\mathsf{k}^{2n}| \\ &\qquad \otimes |\mathsf{l}^{2n}\rangle_{R_B^n}\langle\mathsf{l}^{2n}| \otimes \rho_E^{\otimes n}(\mathsf{k}^{2n},\mathsf{l}^{2n}). \qquad \text{(A1)}\end{aligned}$$

Here, we replace the sequence $\mathbf{l}^n$ with the sequence l^{2n} so that each element $[\mathsf{l}]_{2j-1}$ corresponds to the element $[Q_\mathsf{l}]_j$ and each element $[\mathsf{l}]_{2j}$ to the element $[P_\mathsf{l}]_j$ for $j = 1 \ldots n$.

In RR, Alice guesses Bob's sequence l^{2n} with $\tilde{\mathsf{l}}^{2n}$ using her corresponding sequence k^{2n} and leak_{EC} bits of information from Bob. The parties publicly compare the two hashes of length $\lceil 1 - \log_2 \epsilon_{\text{cor}} \rceil$ computed from $\tilde{\mathsf{k}}^{2n}$ and l^{2n} respectively. If they are equal, the parties continue with the protocol with probability p_{EC}; otherwise they abort. This procedure is associated with a residual failure probability ϵ_{cor}, which bounds the probability of the two sequences being different, even if their hashes coincide

$$p_{\text{EC}}\text{Prob}(\tilde{\mathsf{l}}^{2n} \neq \mathsf{l}^{2n}) \leq p_{\text{EC}} 2^{-\lceil 1-\log_2 \epsilon_{\text{cor}}\rceil} \leq \epsilon_{\text{cor}}. \qquad \text{(A2)}$$

In turn, EC can be simulated by a projection $\Pi_\mathfrak{G}$ of Alice's and Bob's classical registers R_A^n and R_B^n onto a "good" set $\mathfrak{G}$ of sequences. With success probability

$$p_{\text{EC}} = \text{Tr}(\Pi_\mathfrak{G}\rho^{\otimes n}). \qquad \text{(A3)}$$

This quantum operation generates a classical-quantum state

$$\tilde{\rho}^n := p_{\text{EC}}^{-1}\Pi_\mathfrak{G}\rho^{\otimes n}\Pi_\mathfrak{G} \qquad \text{(A4)}$$

which is restricted to those sequences $\{\mathsf{k}^{2n}, \mathsf{l}^{2n}\}$ that can be corrected, i.e., mapped to a successful pair $\{\tilde{\mathsf{l}}^{2n}, \mathsf{l}^{2n}\}$.

The parties continue with the PA step with probability p_{EC} and apply a two-way hash function over $\tilde{\rho}^n$ which outputs the PA state $\bar{\rho}^n$, i.e., $\rho^{\otimes n} \to \tilde{\rho}^n \to \bar{\rho}^n$, with the later approximating the ideal state (defined below)

$$p_{\text{EC}} D(\bar{\rho}^n, \rho_{\text{id}}) \leq \epsilon_{\text{sec}}. \qquad \text{(A5)}$$

In fact, Alice and Bob perform EC and PA over the state $\rho^{\otimes n}$, in order to approximate the s_n-bit ideal classical-quantum state

$$\rho_{\text{id}} := 2^{-s_n} \sum_{z=0}^{2^{s_n}-1} |z\rangle_{R_A^n}\langle z| \otimes |z\rangle_{R_B^n}\langle z| \otimes \rho_{E^n}, \qquad \text{(A6)}$$

with Alice's and Bob's classical registers completely decoupled from Eve and containing the same completely-random sequence z with length s_n. Using the triangle inequality, one obtains [23, Th. 4.1]

$$p_{\text{EC}} D(\tilde{\rho}^n, \rho_{\text{id}}) \leq \epsilon := \epsilon_{\text{cor}} + \epsilon_{\text{sec}}. \qquad \text{(A7)}$$

The state $\bar{\rho}^n$ will contain s_n bits of shared uniform randomness satisfying the direct leftover hash bound

$$s_n \geq H_{\min}^{\epsilon_{\text{s}}}(\mathsf{l}^{2n}|E^n)_{\bar{\rho}^n} + 2\log_2 \sqrt{2}\epsilon_{\text{h}} - \text{leak}_{\text{EC}}. \qquad \text{(A8)}$$

Here $H_{\min}^{\epsilon_{\text{s}}}(\mathsf{l}^{2n}|E^n)_{\bar{\rho}^n}$ is the smooth min-entropy of Bob's sequence l^{2n} conditioned on Eve's system E^n after EC, and the smoothing ϵ_{s} and hashing ϵ_{h} parameters satisfy

$$\epsilon_{\text{s}} + \epsilon_{\text{h}} = \epsilon_{\text{sec}}. \qquad \text{(A9)}$$

In Eq. (A8) we explicitly account for the bits leaked to Eve during EC. In fact, one may write $s_n \geq H_{\min}^{\varepsilon_{\text{s}}}(\mathsf{l}^{2n}|E^n R)_{\bar{\rho}^n} + 2\log_2\sqrt{2}\varepsilon_{\text{h}}$ where R is a register of dimension $d_R = 2^{\text{leak}_{\text{ec}}}$, while E^n are the systems used by Eve during the quantum communication. Then, the chain rule for the smooth-min entropy leads to $H_{\min}^{\varepsilon_{\text{s}}}(\mathsf{l}^{2n}|E^n R)_{\bar{\rho}^n} \geq H_{\min}^{\varepsilon_{\text{s}}}(\mathsf{l}^{2n}|E^n)_{\bar{\rho}^n} - \log_2 d_R$. As we have seen earlier (see Eq. (50)), in the proposed EC procedure, Bob sends to Alice $p - R_{\text{code}}q$ bits for each of the quadratures in a signal state. This allows us to bound the leakage term by

$$\text{leak}_{\text{EC}} \leq 2n(-R_{\text{code}}q + p). \qquad \text{(A10)}$$

We now use that the previous result is connected with the smooth min entropy of $\rho^{\otimes n}$, which will later allow the AEP approximation. In fact, one can show that (see [14, Appendix G2])

$$H_{\min}^{\epsilon_s}(\mathsf{l}^{2n}|E^n)_{\bar{\rho}^n} \geq H_{\min}^{p_{\text{EC}}\epsilon_s^2/3}(\mathsf{l}^{2n}|E^n)_{\rho^{\otimes n}} + \log_2(1 - \epsilon^s/3). \qquad \text{(A11)}$$

Let us assume that the parties concatenate their discretized values corresponding to the two quadrature variables of a single channel use according to the *bidirectional mapping*:

$$l = Q_\mathsf{l} 2^p + P_\mathsf{l}. \qquad \text{(A12)}$$

In that sense, instead of labeling the classical states as in Eq. (A1) by using the combination of two labels, each described by p bits, we use one label described by $2p$ bits. Therefore, we have a classical mapping from a state $\rho^{\otimes n} := \rho^{\otimes n}_{\mathsf{l}^{2n}}$ described by the sequence l^{2n} to the state

$$\rho^{\otimes n}_{l^n} \leftarrow \rho^{\otimes n}_{\mathsf{l}^{2n}} \tag{A13}$$

described by the sequence l^n. In Eq. (A11), this implies the following relation for the smooth min-entropy of the two states:

$$H_{\min}^{p_{\mathrm{EC}}\epsilon_{\mathrm{s}}^2/3}(\mathsf{l}^{2n}|E^n)_{\rho^{\otimes n}_{\mathsf{l}^{2n}}} \geq H_{\min}^{p_{\mathrm{EC}}\epsilon_{\mathrm{s}}^2/3}(l^n|E^n)_{\rho^{\otimes n}_{l^n}}, \tag{A14}$$

where we use Appendix B.

Then, from the AEP theorem, one obtains

$$H_{\min}^{p_{\mathrm{EC}}\epsilon_{\mathrm{s}}^2/3}(l^n|E^n)_{\rho^{\otimes n}_{l^n}} \geq nH(l|E)_\rho - \sqrt{n}\Delta_{\mathrm{AEP}}(p_{\mathrm{EC}}\epsilon_{\mathrm{s}}^2/3, |\mathcal{L}|), \tag{A15}$$

where $H(l|E)_\rho$ is the conditional von Neumann entropy computed over the single-copy state ρ (after applying the mapping of Eq. (A12)) and

$$\Delta_{\mathrm{AEP}}(\epsilon_{\mathrm{s}}, |\mathcal{L}|) = 4\log_2(\sqrt{|\mathcal{L}|}+2)\sqrt{\log_2(2/\epsilon_s^2)} \tag{A16}$$

with $|\mathcal{L}|$ being the cardinality of the discretized variable l, i.e., in our case 2^{2p}. By combining Eqs. (A8), (A11) and (A15), we write the following lower bound

$$\begin{aligned} s_n \geq\ & nH(l|E)_\rho - \sqrt{n}\Delta_{\mathrm{AEP}}(p_{\mathrm{EC}}\epsilon_{\mathrm{s}}^2/3, 2^{2p}) \\ & + \log_2(1-\epsilon^s/3) + 2\log_2\sqrt{2}\epsilon_{\mathrm{h}} - \mathrm{leak}_{\mathrm{EC}}. \end{aligned} \tag{A17}$$

Note that for the conditional entropy, we have

$$H(l|E)_\rho = H(l) - \chi(l:E)_\rho \tag{A18}$$

where $H(l)$ is the Shannon entropy of l and $\chi(E:l)_\rho$ is Eve's Holevo bound with respect to l. In more detail, using the data processing inequality, we have

$$\chi(E:l)_\rho \leq \chi(E:Q_y,P_y) = \chi(E:\mathbf{y}) \tag{A19}$$

where the latter term is calculated using Eq. (19). Therefore we have

$$H(l|E)_\rho \geq H(l) - \chi(E:\mathbf{y}) \tag{A20}$$

Furthermore, we may make the following replacement (see also Eq. (51))

$$H(l) - n^{-1}\mathrm{leak}_{\mathrm{EC}} = \beta I(\mathbf{x}:\mathbf{y}) \tag{A21}$$

where $I(\mathbf{x}:\mathbf{y})$ is calculated from Eq. (8) and

$$\beta = \frac{H(l) - n^{-1}\mathrm{leak}_{\mathrm{EC}}}{I(\mathbf{x}:\mathbf{y})} \tag{A22}$$

is the reconciliation efficiency.

Replacing Eq. (A21) and (A20) in (A17), we derive

$$\begin{aligned} s_n \geq\ & nR_{\mathrm{asy}} - \sqrt{n}\Delta_{\mathrm{AEP}}(p_{\mathrm{EC}}\epsilon_{\mathrm{s}}^2/3, 2p) \\ & + \log_2(1-\epsilon_{\mathrm{s}}^2/3) + 2\log_2\sqrt{2}\epsilon_{\mathrm{h}} \end{aligned} \tag{A23}$$

where we can use the asymptotic secret key rate of Eq. (22). After a successful PE, the parties compute R_{asy} over a state $\tilde{\rho}^n_{\mathrm{wc}}$ (instead $\tilde{\rho}^n$), calculated with respect to the worst-case parameters given in Eq. (27) along with the worst case scenario entropy in Eq. (47). As a result, Eq. (A7) is replaced by the following

$$p_{\mathrm{EC}}D(\tilde{\rho}^n_{\mathrm{wc}}, \rho_{\mathrm{id}}) \leq \epsilon_{\mathrm{cor}} + \epsilon_{\mathrm{h}} + \epsilon_{\mathrm{s}}. \tag{A24}$$

However, there is still the probability that the actual state is a bad state $\tilde{\rho}^n_{\mathrm{bad}}$ with probability $\tilde{\epsilon}_{\mathrm{PE}} = 2\epsilon_{\mathrm{PE}} + \epsilon_{\mathrm{ent}}$. On average, this is given by

$$\rho_{\mathrm{PE}} = (1-\tilde{\epsilon}_{\mathrm{PE}})\tilde{\rho}^n_{wc} + \tilde{\epsilon}_{\mathrm{PE}}\tilde{\rho}^n_{\mathrm{bad}} \tag{A25}$$

whose distance from the assumed worst-case state is

$$p_{\mathrm{EC}}D(\rho_{\mathrm{PE}}, \tilde{\rho}^n_{wc}) \leq p_{\mathrm{EC}}\tilde{\epsilon}_{\mathrm{PE}}. \tag{A26}$$

By using Eqs. (A24) and (A26), together with the triangle inequality, we have that

$$p_{\mathrm{EC}}D(\rho_{\mathrm{PE}}, \rho_{\mathrm{id}}) \leq \epsilon_{\mathrm{cor}} + \epsilon_{\mathrm{h}} + \epsilon_{\mathrm{s}} + p_{\mathrm{EC}}(2\epsilon_{\mathrm{PE}} + \epsilon_{\mathrm{ent}}). \tag{A27}$$

Then the secret key length can be bounded by

$$\begin{aligned} s_n \geq\ & nR_M - \sqrt{n}\Delta_{\mathrm{AEP}}(p_{\mathrm{EC}}\epsilon_{\mathrm{s}}^2/3, 2p) \\ & + \log_2(1-\epsilon_{\mathrm{s}}^2/3) + 2\log_2\sqrt{2}\epsilon_{\mathrm{h}}, \end{aligned} \tag{A28}$$

where R_M has been taken from Eq. (31). Finally, our previous specific analysis of the EC process allows us to connect R_M with the practical rate R_M^{EC} through the parameter $\hat{\beta}$ in Eq. (53). By replacing the latter in the previous secret key bound and multiplying by the successful probability of a block p_{EC} over the number of signals per block N, we obtain the composable secret key rate of Eq. (61).

Note that, although the concatenation of the quadratures may not be applied in practice, theoretically, it has to be considered for the calculation of the discretization parameter $|\mathcal{L}|$ included in the correction term Δ_{AEP}. In fact, considering the proposed EC procedure, $|\mathcal{L}|$ takes the value $2p$ instead of p, compared with the case of the homodyne protocol [16]. In turn, this affects the compression needed to extract a secret key with length s_n.

Appendix B: Classical data mapping and smooth-min entropy

Let us assume a bidirectional mapping $X \leftrightarrow Z = f(X)$ where X is a discrete random variable taking values x in the alphabet $\mathcal{X}$ with probability p_X. Then, Z takes values $z = f(x) \in \mathcal{Z}$ with probability p_Z. In fact, the probability function can absorb the action of f such that

$$p_Z(z) = p_Z(f(x)) = p_X(x). \tag{B1}$$

Therefore, the probabilities for the letters in $\mathcal{Y}$ are the same for the corresponding letter in $\mathcal{X}$.

We want to investigate what is the effect on $H_{\min}^{\epsilon_s}$ of such a mapping, when it is appplied to the classical system of the CQ state

$$\rho_{XE} = \sum_x p_X(x)|x\rangle_X\langle x| \otimes \rho_E(x). \tag{B2}$$

To do so we adapt the proof of [22, Prop. 6.20] for the state ρ_E instead of ρ_{AB}. Thus we apply the isometry $U : U_X \otimes I_E$, with $U_X : |x\rangle \mapsto |x\rangle_{X'}|f(x)\rangle_Z$ being the Stinespring dilation of f and I_E the identity. As a result, we obtain the state

$$\tau_{X'ZE} = U\rho_{XE}U^\dagger. \tag{B3}$$

According to the invariance of the smooth min-entropy under isometries (see [22, Corollary 6.11]), we have the following relation

$$H_{\min}^{\epsilon}(X|E)_\rho = H_{\min}^{\epsilon}(X'Z|E)_\tau. \tag{B4}$$

Furthermore, by using [22, Lemma 6.17], we may write

$$H_{\min}^{\epsilon}(X'Z|E)_\tau \geq H_{\min}^{\epsilon}(Z|E)_\tau \tag{B5}$$

for

$$\tau_{ZE} = \sum_x p_X(x)|f(x)\rangle_Z\langle f(x)| \otimes \rho_E(x). \tag{B6}$$

From Eq. (B4) and (B6) we finally obtain

$$H_{\min}^{\epsilon}(X|E)_\rho \geq H_{\min}^{\epsilon}(Z|E)_\tau. \tag{B7}$$

Note that, in the same way, Eq. (B7) can be extended to the case of two classical systems X and Y considering a Stinespring dilation $U_{XY} = U_X U_Y$ with $U_X : |x\rangle \mapsto |x\rangle_{X'}|f(x)\rangle_Z$ and $U_Y : |y\rangle \mapsto |y\rangle_{Y'}|f(y)\rangle_{Z'}$. Combining then Eq. (B4) and (B6) for the state

$$\begin{aligned}\rho_{XYE} = &\sum_{xy} p_{XY}(xy)|x\rangle_X\langle x| \\ &\otimes |y\rangle_Y\langle y| \otimes \rho_E(x,y),\end{aligned} \tag{B8}$$

one may write

$$H_{\min}^{\epsilon}(XY|E)_\rho \geq H_{\min}^{\epsilon}(ZZ'|E)_\tau, \tag{B9}$$

where

$$\begin{aligned}\tau_{ZZ'E} = &\sum_{xy} p_{XY}(x,y)|f(x)\rangle_Z\langle f(x)| \\ &\otimes |f(y)\rangle_{Z'}\langle f(y)| \otimes \rho_E(x,y).\end{aligned} \tag{B10}$$