Information Rates With Non Ideal Photon Detectors in Time-Entanglement Based QKD

Dunbar Paul Birnie IV, Christopher Cheng, and Emina Soljanin, *Fellow, IEEE*

***Abstract*— We develop new methods of quantifying the impact of photon detector imperfections on possible secret key rates in Time-Entanglement based Quantum Key Distribution (QKD). We address photon detection timing jitter, detector downtime, and dark photon counts and show how each may decrease the maximum achievable secret key rate differently. We begin with a standard Discrete Memoryless Channel (DMC) model to get a good bound on the mutual information lost due to the timing jitter, then introduce a novel Markov Chain (MC) based model to characterize the effect of detector downtime and show how it introduces memory to the key generation process. Finally, we propose a new method of including dark counts in the analysis that shows how dark counts can be especially detrimental when using the common Pulse Position Modulation (PPM) for key generation. Our results show that these three imperfections can significantly reduce the achievable secret key rate when using PPM for QKD. One of our main results is providing tooling for experimentalists to predict their systems' achievable secret key rate given the detector specifications.**

***Index Terms*— Quantum key distribution, time binning, detection jitter, detector downtime, single photon detection.**

I. Introduction

QUANTUM Key Distribution (QKD) generates and distributes secret classical encryption keys between two or more users at different locations. We consider QKD protocols where two parties, Alice and Bob, establish a secret key by communicating over a quantum and a classical channel, which can be accessed by an eavesdropper, Eve. The quantum channel is essential in QKD for preventing undetected eavesdropping and generating randomness when using entanglement-based QKD. At a high level, there are two main QKD steps. In the first step, Alice and Bob generate *raw key* bits by using a quantum channel. Their respective raw keys may disagree at some positions, could be partly known to Eve, and may not be uniformly random. In the second step, Alice and Bob process the raw key to establish a shared *secret key*. They communicate through the public classical channel to reconcile differences between their raw keys, amplify the privacy of the key concerning Eve's knowledge, and compress their sequences to achieve uniform randomness. At the end of the protocol, Alice and Bob 1) have identical uniformly random sequences and 2) are confident the shared sequence is known only to them. Therefore the secret key is private and hard to guess. For an in-depth survey of the most prominent QKD protocols, we refer the reader to [1].

One of the main challenges QKD protocols face today is approaching the secret key generation rates supported by AES and other standard classical key distribution methods [2]. Primarily, this comes down to an inability to extract all the information communicated over the quantum channel. Initially, QKD protocols relied on polarization-entangled photons [3]. Alice and Bob could extract a single key bit by measuring the polarization of entangled photon pairs. Thus, such protocols restrict the maximum secret key rate to the maximum entangled photon pair generation rate. Due to the difficulty of generating entangled photon pairs, these experiments operate in what is referred to as photon-starved conditions. To exceed this limit and to offer high key rates in photon starved conditions, we need some way of extracting multiple bits from each photon pair.

Time-entanglement QKD promises to increase the secret key rate and distribution distances compared to other entanglement-based QKD schemes [4]. In time-entanglement QKD schemes, an independent source (or one of the participants) randomly generates entangled photon pairs. Entangled photon inter-generation times are independent and identically exponentially distributed, giving a source of perfect randomness. Moreover, we theoretically could get arbitrarily many bits from a single photon arrival time given detectors with sufficiently precise measurements. However, such sensors do not exist.

Alice and Bob extract the raw key bits from the arrival times of entangled photons through time binning. Each of them individually discretizes their timeline into *time bins* and groups adjacent time bins into *time frames*. They record photon arrivals as occupied bins within frames. They then use the position of the occupied bins within a time frame to generate random bits. The bit extraction scheme may follow pulse position modulation or some recently proposed adaptive strategies [5], [6]. Under ideal conditions, photon inter-arrival times (as their inter-generation times) are independent and identically exponentially distributed. Several groups studied such systems, most notably [4] who constructed an end-to-end, high dimensional time-entanglement experiment.

Manuscript received 11 July 2022; revised 14 November 2022 and 31 December 2022; accepted 27 January 2023. Date of publication 22 February 2023; date of current version 18 April 2023. This research is based upon work supported by the National Science Foundation under Grant # FET-2007203. The associate editor coordinating the review of this article and approving it for publication was I. B. Djordjevic. *(Corresponding author: Dunbar Paul Birnie IV.)*

The authors are with the Department of Electrical and Computer Engineering, Rutgers, The State University of New Jersey, Piscataway, NJ 08854 USA (e-mail: birnie.nick@gmail.com; emina.soljanin@rutgers.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCOMM.2023.3244244.

Digital Object Identifier 10.1109/TCOMM.2023.3244244

0090-6778 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

However, non-ideal detectors suffer from jitter, dark counts, and downtime. Jitter occurs because of imprecision in the time tagging, which causes discrepancies between Alice's and Bob's raw keys. Dark counts occur due to light leakage into the system and also cause discrepancies between the raw keys. These errors reduce the secret key rate by increasing Alice and Bob's public exchange of information for key reconciliation. Downtime is the time following a photon detection during which no other detection can occur. Thus the perceived arrivals are no longer independent. In their experiment [4], Zhong et al. used a 50:50 beam-splitter to distribute the photon arrivals to two detectors at each station to overcome the loss of photon detections caused by downtime. These dependencies reduce the secret key rate and require Alice and Bob to compress their sequences to achieve uniform randomness.

The adverse effects of various detector imperfections in QKD protocols have been recognized. A recent survey paper [7] extensively studies secure quantum key distribution with realistic devices in the context of prepare-and-measure protocols, such as BB84 [8]. We focus on time entanglement-based QKD. In particular, we model detector jitter, downtime, and dark counts and show how these imperfections affect the secret key rates.

This paper is organized as follows: In Sec. II, we introduce a typical experimental time-entanglement QKD setup. We then establish a system model and describe the various imperfections we will analyze. In Sec. IV, we present the methods used to calculate key rate and show the trade-off between increasing raw key rate and increasing errors due to detector jitter. In Sec. V, we introduce the novel Markov Chain analysis of detector downtime and use these models to investigate the severity of downtime on secret key generation rates. In Sec. VI we pivot and investigate the dark counts' effect on key generation rates. Finally, in Sec. VII, we briefly summarise our results and observations.

II. Time-Entanglement QKD System Model

A. Sources of Entangled Photons

Entangled photons are commonly generated by Spontaneous Parametric Down-Conversion (SPDC). This process occurs when a non-linear crystal is irradiated with a laser, referred to as the pump. The crystal may split the pump photon, resulting in two new photons. This event occurs randomly and relatively rarely at a rate on the order of $10^6 s^{-1} mW^{-1}$ [9]. The emission of a photon pair from such sources is equally likely to occur anywhere within a window equal to the pump coherence time τ_c, and it does not depend on the previous emissions. Because generation times are uniformly random and independent, photon pairs are generated according to a Poisson Process with mean λ_p. Therefore the photon inter-arrival times are exponentially distributed with the rate λ_p. This parameter can be regulated by varying the energy of the pump. Fig. 1 shows a simplified depiction of an SPDC source. One photon is sent to Alice, and one to Bob. The entangled state of the emitted pair is given by

$$|\phi(t_0)\rangle_{AB} \propto \int_{t_0-\tau_c}^{t_0} dt \ |t\rangle_A |t\rangle_B$$

Observe that the state represents a uniform superposition in time within the pump coherence time τ_c from the moment t_0 when the SPDC photons leave the crystal. This process is 1) our source of randomness and 2) the method that ensures Alice and Bob base their keys on correlated information.

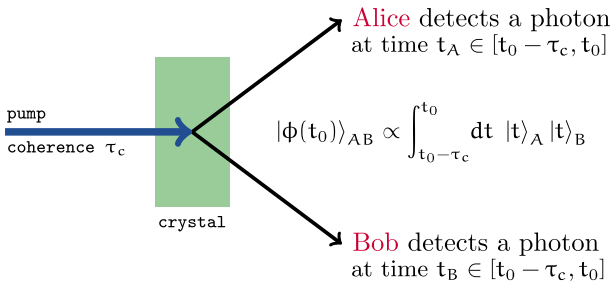

Fig. 1. Generation of time-entangled photons. Ideally, Alice and Bob receive their individual photons at identical and uniformly random times, but that is not true when they use practical, imperfect detectors.

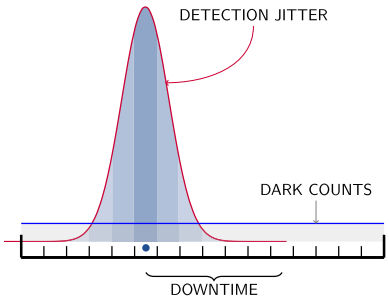

Fig. 2. Detector imperfections include jitter errors, downtime, and dark counts. The Gaussian curve sketches the PDF of the detector jitter. Multiple bins may be affected, each with likelihood proportional to the area under the curve above it. Dark counts occur uniformly within the frame and here are represented by the uniform distribution.

B. Single Photon Detectors

The most common single-photon detectors are Superconducting Nanowire Single-Photon Detectors (SNSPDs). These detectors currently exhibit properties closest to those of the ideal sensors. They have high efficiency, meaning they detect the majority of incident photon arrivals accurately. They have low dark count rates, meaning they rarely report a photon detection without a photon arrival. Furthermore, they have low detector downtime d and slight detector timing jitter that manifests as Gaussian noise with zero mean and variance σ_d^2. Unfortunately, these effects are non-negligible: 1) detector jitters and dark counts cause disagreements between Alice's and Bob's keys, and 2) the downtime introduces memory within the raw key bits. This paper focuses on the secret key rate loss due to these non-ideal properties. Fig. 2 illustrates three of these detector imperfections.

Let t_A be the time when one of the photons of the pair generated by the source is registered by Alice's detector, and

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

t_B be the time the other photon is registered by Bob's detector. Then

$$t_A = u + \eta_A \text{ and } t_B = u + \eta_B. \tag{1}$$

Here η_A and η_B are independent samples of a Gaussian distribution with the zero mean and variance σ_d^2, and u is a sample of a random variable U distributed over the interval $[t_0 - \tau_c, t_0]$. We take u to be the actual arrival time of the SPDC photon.

Let T_A and T_B be the random variable associated with the times Alice and Bob register their photon arrivals. Then the discrepancy between T_A and T_B follows a Gaussian distribution with twice the variance of each detector jitter:

$$T_B - T_A \sim N(0, 2\sigma_d^2).$$

Dark counts arrive uniformly and independently at either detector and are indistinguishable from the SPDC photons. Thus, they can cause significant errors if they make up a large fraction of the detected photons. Dark counts are primarily a result of light leakage into optical lines. When an SPD detects a photon, it enters a temporary state wherein no other photon arrivals can be detected. This means a possibility of our previously mentioned interval (which was $[t_0 - \tau_c, t_0]$) being truncated or eliminated entirely. This effect further limits the observed photon arrival rate.

C. Eavesdropping Model

The described system allows raw key generation at two locations. Since there is no passive eavesdropping possible on a quantum channel, Alice and Bob can always detect the presence of Eve. Systems implementing this kind of QKD experiment have been recently shown to achieve photon information efficiency up to 4.082 secure-key bits/photon and a secure-key rate up to 237-kbit/s [10]. They commonly pass a fraction of photons through a special interpreter to produce entangled photons in the maximally entangled state $|\varphi_{AB}\rangle \propto |0_A 0_B\rangle + |1_A 1_B\rangle$. Alice and Bob can quantify Eve's information gain based on such photons by playing a variant of the CHSH game. We assume that they halt the key distribution if they detect an eavesdropper beyond the non-classical bound of the CHSH game. Thus, the raw keys that Alice and Bob receive have guaranteed security against eavesdropping on the quantum channel. Equivalently, we could say there has been no eavesdropping on the quantum channel. We focus on the second phase of the key distribution process wherein Alice and Bob take their arrival sequences and map them into identical secret keys.

Due to the detector imperfections, Alice's and Bob's raw keys do not entirely agree (jitters and dark counts cause errors). They are not uniformly random either (detector downtime introduces memory). Therefore, Alice and Bob must first perform information reconciliation to establish key agreement. This process involves transmission over the public channel. Alice and Bob must then perform privacy amplification to regain the lost randomness due to downtime and recover the secrecy lost in the information reconciliation process. This ultimately reduces the key rate such that both these goals may be met. This paper focuses on computing information-theoretic bounds on the key rate loss in the privacy amplification step. In practice, information reconciliation is carried out by using error-correcting codes whose rate will ultimately determine the secret key rate. Practical information reconciliation error-correcting codes for PPM schemes have been recently proposed in [11] and [12].

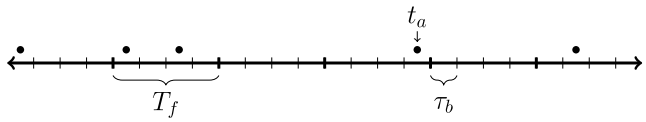

Fig. 3. Time frames of duration T_f consist of $n = 4$ time bins. Each bin has the width $\tau_b = T_f/n$ Photon arrivals are shown as points above the line. Each is said to "occupy" the bin it lands within.

D. Raw Key Extraction

Alice and Bob rely on the correlated random photon arrivals to generate their secret keys. There are many ways to extract keys from this correlated information. One popular method is similar to Pulse Position Modulation (PPM); see, e.g., [4] and references therein. In PPM, Alice and Bob synchronize their clocks and discretize their timelines into time frames of size T_f each consisting of n time bins. Each time bin has a width of $\tau_b = T_f/n$. In PPM, Alice and Bob agree to retain only time frames in which only one bin is occupied, while discarding all other frames. Individual photons are considered to occupy a time bin depending on where within the frame it arrives – if a photon arrives at time t_A, the occupied frame is frame number $\lfloor t_A/T_f \rfloor$ and the PPM decoding of this frame is $\lfloor (t_A \mod T_f)/\tau_b \rfloor$.

The number of raw key bits K_{raw} that PPM decoding can extract from each retained frame is

$$K_{\text{raw}} = \log_2(n) \text{ bits per frame.} \tag{2}$$

To find the overall number of raw key bits extracted on average R_{raw}, we need to know the probability that a frame is valid, and for that, we need to know the probability that a bin is occupied. Since photon inter-arrival times are exponentially distributed (see Sec. II-A), the number of photons that arrive in a single bin is Poisson distributed. From this, we can derive the probability p of a bin being occupied,

$$p = 1 - \exp(-\lambda_p \tau_b). \tag{3}$$

For now we assume that the probability of each bin being occupied is independent of other bins. We can thus treat the occupancy of each bin as being modeled by a Bernoulli distribution, with parameter p. Hence, the probability of a frame being retained is the probability that only a single bin is occupied, which is $np(1-p)^{n-1}$. Therefore, the highest key rate achievable when using PPM (in bits per frame) is

$$R_{\text{raw}} = K_{\text{raw}} np(1-p)^{n-1} = \log_2(n) np(1-p)^{n-1} \tag{4}$$

Fig. 3 illustrates the described time of arrival key generation.

To illustrate potential disagreements between Alice's and Bob's extracted raw key bits, we consider an example in Fig. 4. Alice and Bob had discretized their timelines into time frames

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

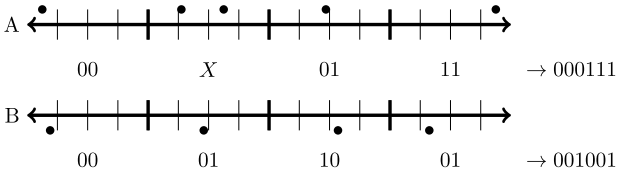

Fig. 4. Two arrival sequences at Alice's and Bob's stations. PPM decoding is below each frame and the resulting key sequences is at the right.

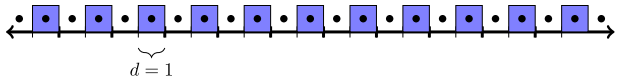

Fig. 5. An example of a deterministic sequence that appears acceptable to Alice and Bob because of the detector downtime. Each time frame consists of $n = 2$ time bins, and the photon generation probability is high. Photon arrivals are shown as points above the line. Every bin is occupied, but the detector downtime (shown in blue) occludes every other arrival tricking Alice and Bob into accepting and decoding each time frame.

consisting of four time-bins. We can see that in the first frame bit extraction works as expected, and both parties extract the same key bits despite some slight timing jitter that stays within the time bin. However, they both throw out the second frame because Alice detected an extra dark count in her frame. The third time frame shows how the timing jitter can cause errors between the extracted keys. Here the jitter caused the photons to be detected in separate bins and in this case caused two bitwise errors in the final key sequences. The fourth frame shows an example where Alice and Bob detect a dark count in the same time frame. Before information reconciliation, these seem like valid PPM frames despite the resultant bits being entirely uncorrelated. The rates of these two kinds of errors are significantly affected by the specific design parameter choice. For example, smaller bins lead to more potential bits per time frame but increase the likelihood of jitter errors, while larger time frames can lead to a greater number of unusable time frames due to multiphoton arrival events (as in the second pair of frames in Fig. 4.)

Finally, Fig. 5 illustrates the impact of detector down time on key extraction. In this example, we have naively parameterized PPM so that the time bin duration equals the detector downtime. We assume that the photon arrival rate is sufficiently high to ensure each bin will detect a photon arrival with a high probability. The resulting key sequence is the all-zero string and is entirely predictable (i.e., carries no randomness). This particular case is contrived and we can avoid such scenarios, but as we'll see, the entropy reduction phenomenon ultimately persists albeit to a lower degree.

III. Paper Goals and Summary of Results

We are now ready to more precisely state this paper's goals and summarise the results. This paper has three goals addressed in the next three sections. We briefly describe them below.

A. Information Loss Due to Detector Jitter

Alice and Bob get their raw key bits based on the correlated (ideally coincidental) arrival times of the SPDC-generated entangled photon pairs. The level of correlation is determined by the detector jitter noise variance and the time bin size (since we extract the raw key bits by time binning). The smaller the time bin, the higher the raw key rate. On the other hand, the smaller the time bin, the more Alice's and Bob's raw keys may disagree due to jitter. An increase in the disagreement between their keys necessitates an increase in the public communication for key reconciliation, thus, reducing the secret key length. We compute the number of bits β_r that have to be sent from Alice to Bob to reconcile their raw keys as a function of the jitter variance and bin size. We observe that when the detector jitter variance increases, the bin count that maximizes the post reconciled key rate decreases. These results and trade-offs are shown in Sec. IV.

B. Information Loss Due to Detector Downtime

When the detector exhibits downtime, Alice's and Bob's raw key bits are not uniformly random, as we illustrated in the exaggerated example of Fig. 5. Alice and Bob have to compress their reconciled raw keys to achieve uniform randomness. We compress the keys by the compression rate C_r^d, which is a function of the detector downtime, photon generation rate, and the number of bins per frame. To characterize the impact of detector downtime on the system, we model combined detector and time binning operations by Markov Chains (MC). The entropy rate of the system's MC determines the minimum compression rate to guarantee the key's uniform randomness. These MCs are complex, and we developed an algorithm to create them for various parameter values, implemented in an online tool available at https://cc1539.github.io/qkd-binning-demo-2. Sec. V presents these results and shows how the compression rate C_r^d depends on downtime.

C. Modeling the Impact of Dark Counts

We introduce a novel method for analyzing the impact of dark counts in entanglement QKD using PPM. We develop a way to model coincident dark counts indistinguishable from valid PPM frames. In doing so, we characterize the severity of dark counts in PPM more accurately than in prior efforts [13]. This work is shown in Sec. VI.

D. Estimating the Secret Key Rate

The key rate simultaneously suffers from all impairments discussed above. This paper only looks into the rate loss caused by the detector jitters and downtime and provides and proposes a model for dark counts. We address the detector jitter noise and downtime separately because these two impairments cause rate loss in very different ways. We next describe their combined effect on the key rate.

We define two key rates of interest. First, $K_{\text{reconciled}}$ corresponds to the number of key bits per PPM frame after reconciliation and privacy amplification. Then, K_{uniform} corresponds to the number of key bits per PPM frame after compression to remove downtime-induced memory.

$$K_{\text{reconciled}} = K_{\text{raw}} - \beta_r \text{ and } K_{\text{uniform}} = C_r^d \cdot K_{\text{raw}} \quad (5)$$

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

TABLE I
QKD SYSTEM PARAMETERS AND VARIABLES

λ_p	–	entangled photon generation rate
d	–	detector downtime
σ_d^2	–	detector jitter variance
T_f	–	PPM frame width
n	–	number of time bins per frame
τ_b	–	time bin width, $\tau_b = T_f/n$
p	–	bin occupancy probability, $p = 1 - \exp(-\lambda_p \tau_b)$
K_{raw}	–	number of raw key bits per PPM frame, $K_{\text{raw}} = \log_2(n)$
β_r	–	number of bits sent for reconciliation
C_r^d	–	downtime compression ratio

Combining the two, we obtain K_{secret}. This represents the number of secret key bits per PPM frame, and is given by

$$K_{\text{secret}} = C_r^d (K_{\text{raw}} - \beta_r). \tag{6}$$

We have one step left to our ultimate performance metric. Recall that PPM only retains frames with a single arrival. Therefore to get the average key rate R, the key bits per valid frame K have to be multiplied by the probability that a frame contains exactly one occupied bin, which is $np(1-p)^{n-1}$:

$$R_{\text{secret}} = np(1-p)^{n-1} K_{\text{secret}} \tag{7}$$

The following sections will show how to separately evaluate the rate-loss metrics in equation (6) that computes their joint influence on the secret key rate. We do not look into different impairments simultaneously in the following sense. In evaluating the downtime effect in Sec. V, we disregard jitter errors. Therefore, only downtime introduces memory. In evaluating the jitter effect in Sec. IV, we assume that the difference between Alice's and Bob's photon detection is the difference between two zero-mean Gaussian random variables. Instead, if we took the influence of the downtime simultaneously, the mean of these Gaussian random variables would also be a random variable whose mean is zero because of the symmetric jitter, which could have caused Alice's downtime to start before or after Bob's. Thus our jitter analysis is the expected case analysis. More detail is provided in the corresponding sections below.

E. Notation and Assumptions

As a reminder and a reference, we list the previously defined system parameters and variables in Table I. The table also lists some relationships between the parameters. This paper assumes that T_f, λ_p, and d are system parameters determined by a particular system setup and equipment. The design parameter we can choose in an attempt to optimize performance metrics is the number of bins per frame, n.

We adopt models widely used in the literature, e.g., Poisson photon statistics (described in textbooks such as [14, Chapter 5]) and Gaussian jitters which are commonly denoted in terms of full width half maximum (FWHM) [15] (and described this way in detector specification sheets such as [16]). For further experimental details and device specifications we direct the reader to [4] and [17]. However, our theoretical approach, derivations, and numerical evaluations do not rely on such assumptions. The parameter values used in our numerical examples are selected based on the corresponding instances in the literature and to emphasize specific points.

IV. INFORMATION LOSS DUE TO DETECTOR JITTER

A. The Source Model Secret Key Agreement Rate

The secret key rate is the "maximum rate at which Alice and Bob can agree on a secret key while keeping the rate at which Eve obtains information arbitrarily small" [18]. In the case of time-entanglement-based QKD, Alice and Bob obtain correlated streams of bits (raw keys) based on their respective Time of Arrival (ToA) measurements, as described in the previous section. However, they must communicate to agree on a key, i.e., reconcile their differences. Every communication required for this process must be considered open communication accessible to Eve. Here, we focus on one-way information reconciliation schemes in which Alice sends information about her sequence to Bob, who uses it to correct the differences between his and Alice's raw keys.

After one-way information reconciliation, Alice and Bob share Alice's initial raw key. However, since they communicated over a public channel, the shared key is not secret. To correct that, Alice and Bob perform privacy amplification. They commonly hash their shared keys, establishing secrecy but shortening the key. The secret key rate is the post-privacy amplification key length per second. This section aims to characterize the secret key rate when Alice and Bob obtain their raw key rates in the presence of detector jitters. Recall that in this case, Alice and Bob base their secret key generation on correlated arrival times given in (1). We know that, for this model (referred to as the source model in Information Theory [19, Chapter 22.3]), when the eavesdropper has access to the public communication but does not have correlated prior information (see, e.g., [19, p. 567]), then the secrecy capacity is equal to the mutual information between Alice's detected arrival time T_A and Bob's detected arrival time T_B,

$$C_K = I(T_A; T_B). \tag{8}$$

.

B. Secret Key Rates in PPM Key Extraction

Let us consider the case where Alice and Bob use the PPM scheme to extract raw key bits, and jitter errors are present as described in Sec. II-B with variance σ_d. The two observe discrete correlated random variables identifying the occupied bin in the frame. When the number of bins per frame is n, Alice observes X_n and Bob observes Y_n given by

$$X_n = U_n + J_{A,n} \text{ and } Y_n = U_n + J_{B,n} \tag{9}$$

where U_n is uniform over $\{0, 1, \ldots, n-1\}$, and $J_{A,n}$ and $J_{B,n}$ have integer support, taking value k with the probability

$$\begin{aligned} P\big[J_{A,n} = k\big] &= \frac{1}{\sqrt{2\pi\sigma_d^2}} \int_{(k-1/2)\tau_b}^{(k+1/2)\tau_b} \exp(-t^2/(2\sigma_d^2))\, dt. \\ &= \frac{1}{\sqrt{2\pi}} \int_{(k-1/2)T_f/(n\sigma_d)}^{(k+1/2)T_f/(n\sigma_d)} \exp(-\tau^2/2)\, d\tau. \end{aligned} \tag{10}$$

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

Because of the jitter in the previous frame, the downtime causes a zero-mean difference in U_n between Alice and Bob.

The rate of disagreement (RoD) between Alice's and Bob's raw keys is given by

$$\text{RoD} = 1 - P\big[J_{A,n} = J_{B,n}\big] \tag{11}$$

Observe from (10) and (11), that the key-bit disagreement happens at a rate that depends on the ratio of the detector noise variance to time frame ratio σ_d/T_f and the number of bins per frame n. The PPM time frame size is constrained by the pump coherence time. The reported coherence time in [4] is 250-330 ns. Since the number of bits per PPM-valid frame equals $\log_2(n)$, increasing n should give more raw key bits per frame. However, the bin width decreases as n increases, making the binning process more prone to errors; See Fig. 2. In Fig. 6a, we vary the number of bins per frame and observe the key disagreement rates. The values for σ_d/T_f are selected by choosing the equivalent ratio for an SNSPD with FWHM jitter of 80ps or $\sigma_d = 33.97\ ps$, and $T_f = 330\ ns$ as in [4].

The secrecy capacity of our PPM scheme for the number of bins equal to n is

$$C_{K,n} = I(X_n; Y_n), \tag{12}$$

where X_n and Y_n are given by (9). We say that $I(X,Y) = \lim_{n\to\infty} I(X_n;Y_n)$ is the ultimate achievable secret key rate.

To compute $I(X_n;Y_n)$, we observe that Bob perceives Alice's detection as corrupted by the difference of the two independent Gaussian random variables (Alice's and Bob's noise). We can therefore compute the transition probabilities $p(y|x)$ of the event that Bob detects a photon arrival in bin y given that Alice detects her entangled photon arrival in bin x. We calculate these transitions by integrating the double-jitter error distribution over the domain of the specific bin where Bob has detected his arrival, as

$$p(y|x) = \frac{1}{\sqrt{4\pi\sigma_d^2}} \int_{(y-x-1/2)\tau_b}^{(y-x+1/2)\tau_b} \exp\big[-t^2/(4\sigma_d^2)\big]\, dt. \tag{13}$$

We note that for a given x the transition probabilities are a function of the difference $x - y$. This phenomenon allows us to approximate the mutual information in (12) as follows (see [20, Ch. 7.2]):

$$\begin{aligned} K_{\text{reconciled}} < C_{K,n} &= I(X_n;Y_n) = \log_2(n) \ - H(Y_n|x) \\ &= K_{\text{raw}} - H(Y_n|x) \text{ bits per frame,} \end{aligned} \tag{14}$$

where $H(Y_n|x)$ does not depend on x and can be computed based on the transition probabilities defined in (13). Expression (14) represents an upper bound on the number of secret key bits per frame after information reconciliation. The parameter $\beta_r = K_{\text{raw}} - K_{\text{reconciled}}$ (see (5) above) depends on the rate of the error correcting code used for reconciliation. The value of β_r depends on the efficiency of the chosen code. A more thorough discussion on key rates, other ways to derive them, and coding is given in [12].

Fig. 6b plots $C_{K,n}$ for three values of σ_d/T_f selected as described above for the disagreement rate plot in Fig. 6a. We see that there is hardly any increase in the maximum achievable secret rate beyond a certain number of bins per frame which depends on σ_d/T_f. Comparing the plots in the two parts of Fig. 6, we can observe that the noticeable rate increase stops almost as soon as the key disagreement begins.

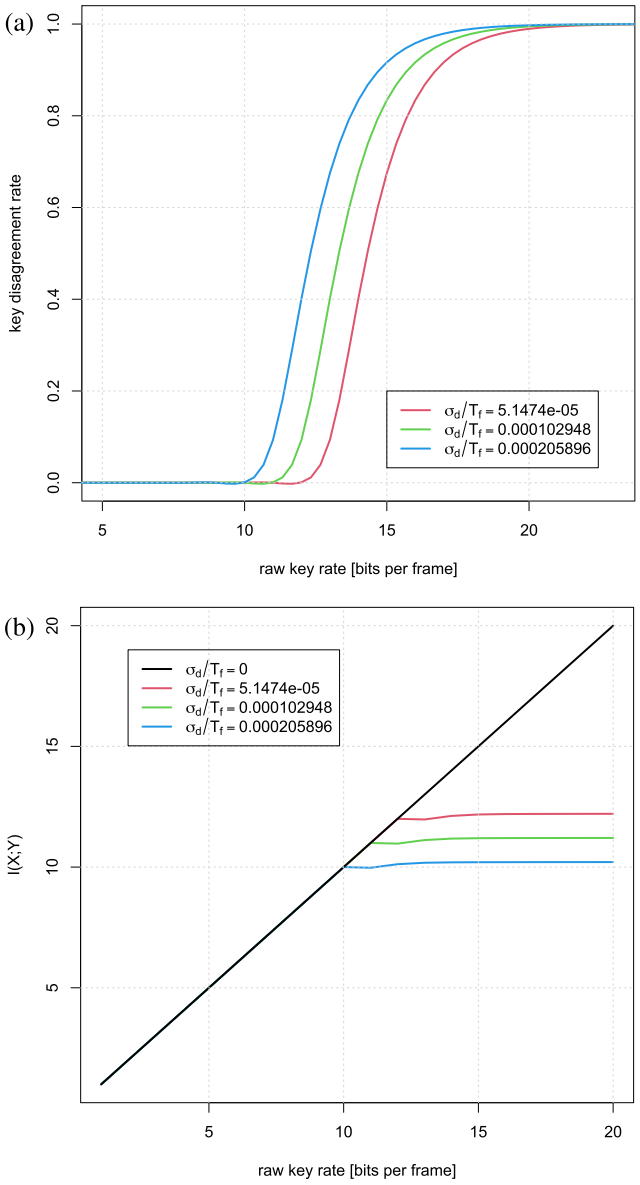

Fig. 6. (a) Dependence of the disagreement rate between Alice's and Bob's keys on the raw key rate (or equivalently, the bin size), cf. (11). As bin size decreases, both the raw key and disagreement rates increase. (b) Dependence of the maximum reconciled key rate on the raw key rate, cf. (14). A larger raw key disagreement leads to more reconciliation information being revealed through the information reconciliation process.

This observation motivates us to ask how close to the bound we can get by selecting the bin size such that it covers the interval $[-\sigma_d/T_f, +\sigma_d/T_f]$ around its central point, or equivalently,

$$n = \lceil 2T_f/\sigma_d \rceil \tag{15}$$

Fig. 7 plots the ultimate bound $I(X,Y)$ on the reconciled key rate for unbounded n. We see that increasing the jitter noise variance to frame size ratio lowers the post-reconciled key rate. Fig. 7 also plots the raw key rate that is given by $\log_2(n)$ where n is estimated in (15) and the secret key rate

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

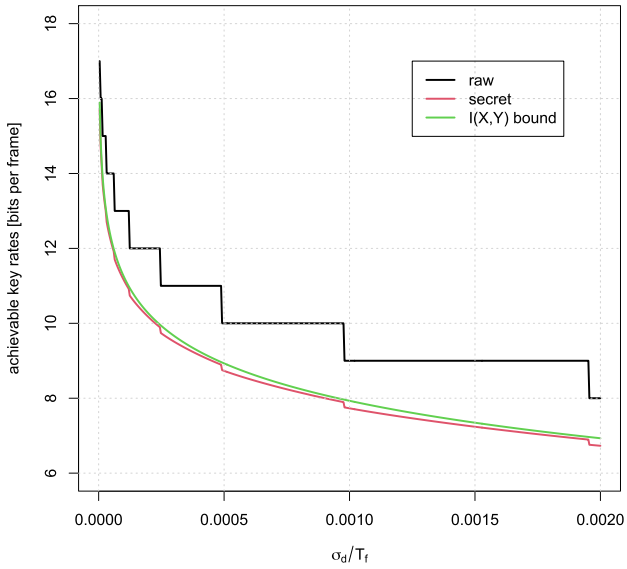

Fig. 7. Dependence of the key rates on the noise level σ_d/T_f. The number of raw key bits per frame determined by (15) is sufficient for achieving a secret key rate that approaches the maximum possible.

that we can achieve with that raw key rate. We see that this secret key rate closely approximates the maximum key rate. Other approximations can be found in our related work [12]

Unlike previous approaches, this DMC modeling approach is straightforward and easily generalizes to any symmetric noise distribution. Earlier work considered jitters to occur within one to three time-bins [13]. This work ignores the possibility of having significant jitters when using very small bin sizes, which is one of the main areas of interest. Previous approaches also force the analysis of dark counts to be considered separately. This is inconsistent with how coincident dark counts manifest in PPM. Our model accounts for all possible jitter errors, not only those close to the original bin, while simultaneously considering the effect of dark counts. We address the case of non-zero dark counts in Sec. VI, where the model is the same, but the jitter error distribution builds to account for dark counts in PPM.

V. Information Loss Due to Detector Downtime

Under the effect of downtime, Alice's and Bob's raw keys are no longer uniformly random bits, as illustrated in the exaggerated example of Fig. 5. Recall that our sequence of bin occupancies (what we run through a binning scheme to extract the raw key bits) is a Bernoulli trial sequence. We have Bernoulli trials when there is no jitter or when the jitter is symmetric, which is the case here. With downtime, this is no longer the case. Photon detection in one bin prevents detection in subsequent bins, thus removing the memoryless behavior of photon arrivals. Therefore, Alice and Bob must compress their raw key sequences to obtain a uniformly random shared key.

This section focuses on computing the average number of secret key bits per frame after the compression. A family of simple Markov chains describes the dynamics of downtime-afflicted systems (see Fig. 8) if we don't consider

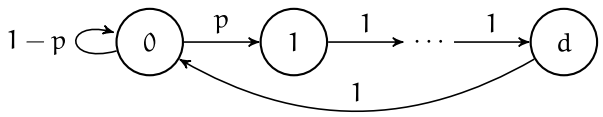

Fig. 8. Possible states of detectors with downtime d and possible state transitions with arrows showing the probability of transitions between each state. This Markov Chain shows that the detector capable of detecting photons sees a photon with probability p, and once it does, it enters into a period of downtime lasting for d states.

the use of PPM (or any binning scheme) to extract raw key bits. The entropy rates of these Markov chains would then determine the secret key rate. When we bring PPM back into the picture, however, the Markov chains used quickly end up with many states, complicating the computation of their entropy rates. To combat this increase in computational complexity, we introduce a family of augmented Markov chain models that can replace the original Markov chain method in certain conditions to calculate a tighter bound on the entropy rate while keeping the number of states tractable.

A. The Impact of Downtime

When we consider detector downtime behavior but ignore the PPM bit extraction, we can model the detector dynamics by the Markov Chain in Fig. 8. The entropy rate of this gives us an upper bound on the information rate of our extracted key,

$$R_{ToA}(p,d) \leq \frac{h(p)}{n(1+pd)} \quad \text{bits/frame}, \tag{16}$$

where $h(p)$ is the binary entropy.

Observe that as d increases, the maximum key rate decreases. Naively, we may set the value of p high, thinking that more photons will necessarily give us a higher information rate. However, that is not the case, especially for binning schemes such as PPM. To see why, consider plot a) in Fig. 9, which shows the key rate as a function of p when we use PPM to extract key bits. As p approaches 1, the raw key rate seems to a maximum. However, we saw before that the information rate approaches 0 at the extremes of p. Indeed, when p is high, the entropy of the raw key gets lower and lower until, at $p = 1$, we theoretically have a deterministic sequence that may yield many bits but no randomness and, thus, no security against attackers aware of the system configuration. Consider plot b) in Fig. 9, which considers the loss of randomness due to downtime.

Note that if downtime is low enough compared to the frame size, this effect does not manifest, and the raw key rate changes just as the information rate does. However, in practice, values for downtime and frame size such as those provided by experiments [21] suggest that downtime is typically much longer than a single frame width. In this regime, we see the phenomenon explained earlier, with the raw key rate increasing while the average information extracted by each frame decreases. The behavior observed in plot b) of Fig. 9 is thus the behavior we might expect to see in live systems. To reiterate, while increasing the raw key rate (without changing bin size)

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

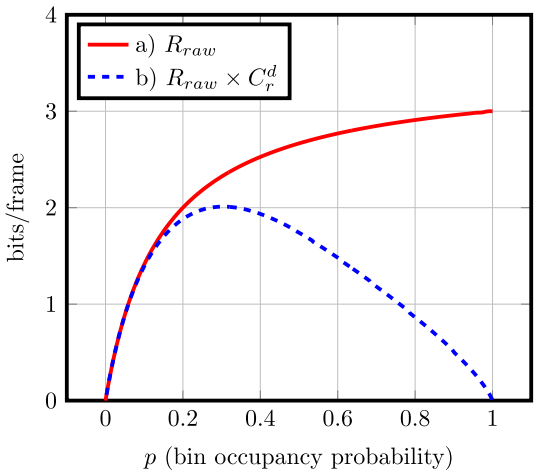

Fig. 9. The PPM key rate as a function of p, the probability that a bin is occupied. The parameter p is given as a function of λ_p and τ_b in (3). In plot a), we are *not* taking into account the loss of entropy by any mechanism and only show the number of bits we get per bin as a result of using the PPM binning scheme on the input we get from the detector. Note that this is not a plot of (4), which was derived under the assumption that all bins were i.i.d., an assumption that no longer holds due to downtime. In plot b), we *are* taking into account the loss of entropy as a result of downtime, giving us a different value which we can call $R_{\text{raw}} \times C_r^d$. These plots were generated using our online tool, available at https://cc1539.github.io/qkd-binning-demo-2/.

improves your key rate up to some point, one must be careful not to increase λ_p (and subsequently p) too high without taking the effects of downtime into account.

B. Constructing Markov Chains to Compute Information Loss

In order to produce plot b) in Fig. 9, we need to expand on the idea of using Markov Chains. The Markov Chain in Fig. 8 is not adequate since it describes the detector on a bin-by-bin basis, while the use of binning schemes means that we must describe the detector on a frame-by-frame basis. For this, we use the techniques described in [22]. The first thing we do is construct what, in this context, we will call *Input* Markov Chains (IMC). These represent what the detector sees, with only the additional processing step of assigning each photon detection to a bin. Analysis of this Markov Chain is sufficient to produce R_{raw}, which is what plot a) in Fig. 9 shows. But to go from a) to b), we need an additional step.

This additional step is the *Output* Markov Chain (OMC). Each state represents the output of a frame after applying your binning scheme of choice. For example, for PPM with $n = 8$, the states of an OMC may correspond to all binary strings of length $\log_2(8) = 3$. The Markov Chain Entropy of the OMC gives us $C_r^d \times 2^{|\text{OMC}|}$, where $|\text{OMC}|$ is the number of states in our OMC. The factor C_r^d represents the degree to which we would need to compress the bits extracted using our binning scheme to achieve perfect randomness. For example, when $d = 0$ we find that $C_r^d = 1$ for all p, reflecting the fact that no compression is required as a result of downtime since there is no downtime. The adjusted key rate, $R_{\text{raw}} \times C_r^d$, is what we see in plot b) of Fig. 9.

One way to build an IMC is to assign each possible frame configuration to one state. Consider Fig. 10.

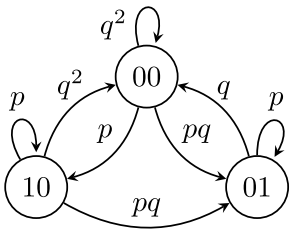

Fig. 10. Markov Chain generated using the BMCM (Basic Markov Chain Method) for the parameters $n = 2$ and $d = 1$. The parameter p is the bin occupancy probability, and $q = 1 - p$. We have three states corresponding to frame configurations 00, 01, and 10 and labeled accordingly. Note how the downtime prevents us from ever being able to observe the 11 frame configuration, and therefore we do not include it in this Markov Chain.

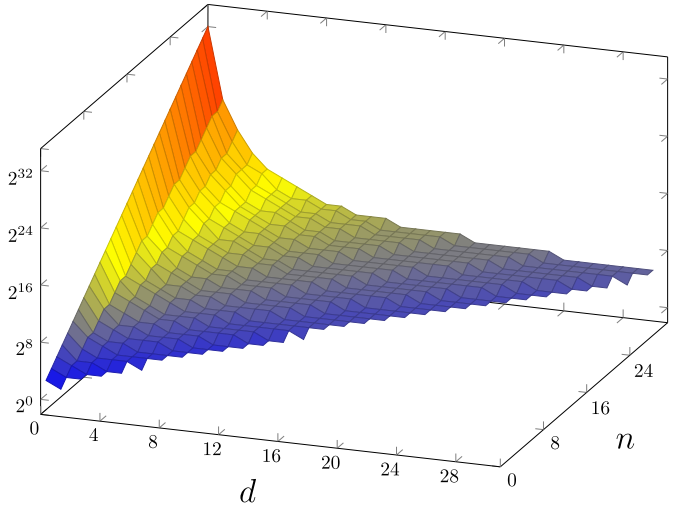

Fig. 11. The number of states in the Markov Chain generated by the BMCM (Basic Markov Chain Method) as a function of n and d. This Markov Chain models the input to the detector on a frame-by-frame basis. Note how the number of states increases rapidly when n increases, especially when d is kept small. Points where $d > n$ are outside the domain of this analysis and so not included in the plot. Points on the plot are colored on a spectrum from blue to red based on their vertical height, which represents the state count.

We will call this method the *Basic Markov Chain Method*, or BMCM. If $d = 0$ (no downtime) this results in 2^n states. With non-zero downtime, some states may be dropped, but the number of states still increases exponentially with n. See Fig. 11.

To avoid this exponential increase in the number of states, we devised an alternative method of assigning frame configurations to Markov Chain states, which we call the *Reduced Markov Chain Method*, or RMCM. In the RMCM, multiple frame configurations can be "combined" and represented by a single, augmented Markov Chain state. These states are augmented in the sense that information about how states are combined is encoded into a triplet of integers (d_o, n_1, d_i) associated with each state, combined or otherwise. We can use these triplets to not only uniquely identify states, but also undo any "compression". The parameter d_o represents the number of bins that are guaranteed to be non-occupied due to downtime from a previous frame leaking into the current frame. The parameter d_i is the number of bins of downtime leaking out of the current frame into the next frame. With d_o and d_i, we can

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

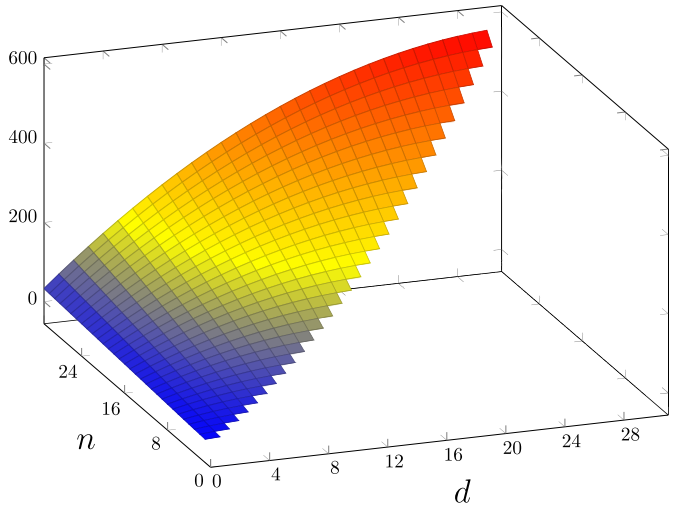

Fig. 12. The number of states generated in the RMCM (Reduced Markov Chain Method), which is used for modeling the input to the detector on a frame-by-frame basis, as a function of n and d. Points where $d > n$ are outside the domain of this analysis and so not included in the plot. Points on the plot are colored on a spectrum from blue to red based on their vertical height, which represents the state count.

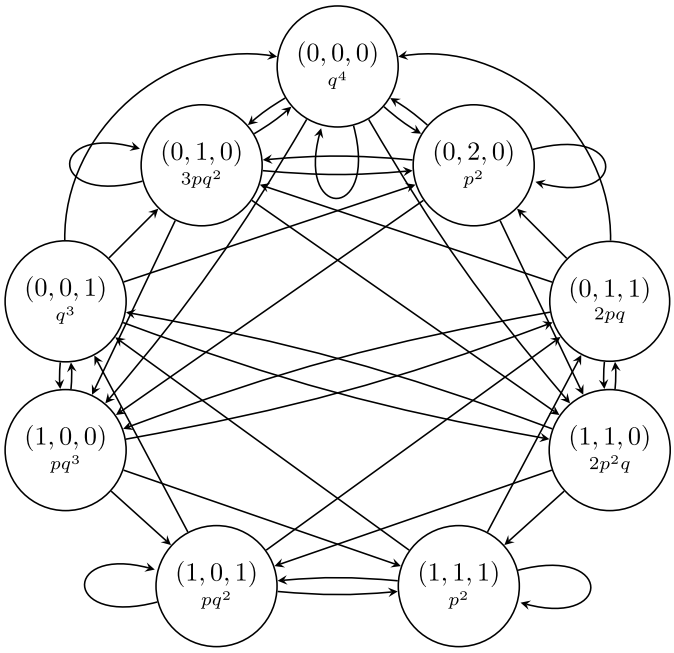

Fig. 13. Markov Chain generated using the RMCM (Reduced Markov Chain Method) for the parameters $n = 4$ and $d = 1$. The parameter p is the probability that a bin is occupied, and $q = 1 - p$. Each state is labeled by a triplet of integers, (d_o, n_1, d_i), as described in section V-B. Instead of showing transition probabilities on each edge, the transition probability is shown on the state itself, under its (d_o, n_1, d_i) identifier.

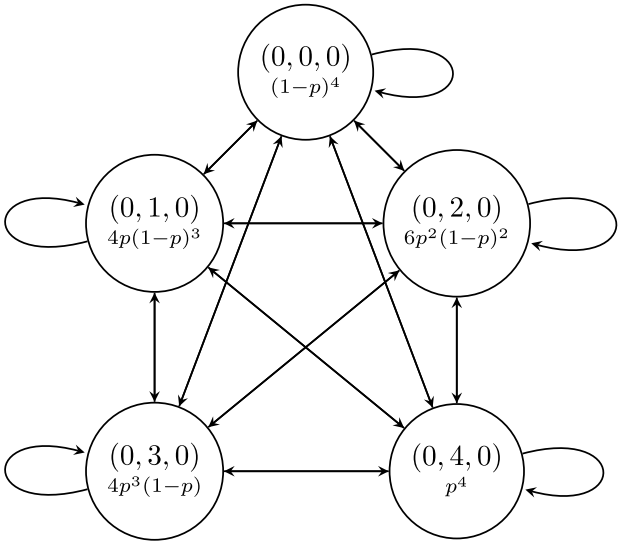

Fig. 14. Markov Chain generated using the RMCM (Reduced Markov Chain Method) for the parameters $n = 4$ and $d = 0$. The parameter p is the bin occupancy probability and $q = 1 - p$. Each state is labeled by a triplet of integers, (d_o, n_1, d_i), as described in section V-B. Instead of showing transition probabilities on each edge, the transition probability is shown on the state itself, under its (d_o, n_1, d_i) identifier. The BMCM would have generated 16 states, but here we have 5 states instead.

describe the occupancy of bins on the edge of the frame. For instance, consider when $d = 5$ and $d_o = 3$. It must then be the case that the last two bins are unoccupied, and the third bin from the last must be occupied. Now we can understand the parameter n_1, which is the number of occupancies inside the remaining space of bins, which we haven't yet been fixed by d_o and d_i. This is where the compression happens; for instance, if $d = 0$ (implying $d_o = 0$ and $d_i = 0$ as well) and $n_1 = 1$, then the state associated with the triplet $(0, 1, 0)$ actually represents n different states at once. This compression results in the number of states increasing polynomially with n, instead of exponentially. See Fig. 12.

As seen in Fig. 12, representing states this way has its demerits. Consider the settings $n = 2$ and $d = 1$. In this case, the Markov Chain generated by the RMCM has five states while the BMCM yields three states. This is because in the RMCM, we actually distinguish between frames where an unoccupied bin is observed through chance and frames where an unoccupied bin is only a result of downtime. For instance, consider the frame configuration "01". It could be that there was no downtime in the front of the frame, in which case we will see this frame configuration with probability pq. However, suppose there was some downtime – perhaps from a previous frame – that had leaked into this frame. In that case, the first unoccupied bin is guaranteed, so the probability of seeing this frame is now just p.

When we bump up n to 4 while keeping d at 1, the RMCM starts to close the gap. Consider Fig. 13; this Markov Chain has nine states, while the Markov Chain generated by the BMCM would have eight states. More importantly, we start to see the first signs of states being "combined".

Consider the state in Fig. 13 labeled $(0, 1, 0)$. This single state represents the three frame configurations, "1000", "0100", and "0010", where we have no downtime going into the frame. Note how all frames represented by the state have only a single occupied bin and thus will yield $\log_2(4) = 2$ bits. This is not a coincidence; the way states can be "compressed" was designed with PPM in mind. In this way, the triplet of numbers each state is labeled with is enough to tell us how many bits, on average, the frames represented by that state will yield. This way, we do not have to go through the trouble of undoing the compression and checking each frame

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

configuration one by one, which would partially defeat the purpose of using the RMCM as an alternative to BMCM for calculating R_{raw}.

For an example where the RMCM produces fewer states than the BMCM, see Fig. 14.

Notice how in the example where the RMCM does better than the BMCM, we *decreased* the parameter d from 1 to 0. After doing so, the number of states generated by the BMCM decreases as d increases, but the number of states generated by the RMCM increases. Therefore, we consider both methods in a hybrid method wherein we select the method that results in the fewest number of Markov Chain states. It seems, based on inspection of the data used to produce Fig. 15, that a good enough rule is to choose BMCM when $d > n/2$ and choose RMCM otherwise. This may not always be the case though, so if we absolutely must choose the more efficient method, one may utilize (17) and (18), which give the exact number of states for the BMCM and RMCM, respectively.

The number of states used in the BMCM when given n and d is given by the recursive function N_{BMCM} as defined as

$$N_{\text{BMCM}}(n,d) = \begin{cases} 1 & n < 1 \\ N_{\text{BMCM}}(n-1,d) + N_{\text{BMCM}}(n-1-d,d) & n \geq 1 \end{cases} \tag{17}$$

The number of states used in the RMCM when given n and d is described by the function N_{RMCM} as defined in (18).

$$N_{\text{RMCM}}(n,d) = \sum_{d_i=0}^{d} \Big(N'(d_i,0) + \sum_{d_o=\max(1+d+d_i-n,1)}^{d} N'(d_i,d_o) \Big) \tag{18}$$

where

$$N'(d_i,d_o) = \left\lfloor \frac{n'(d_i,d_o)}{d+1} \right\rfloor + 1 \text{ and}$$

$$n'(d_i,d_o) = \begin{cases} n-d_i & d_o = 0 \\ (n-d_i)-(d+1-d_o) & d_o > 0 \end{cases}.$$

With an IMC in hand, we can calculate R_{raw}. Suppose we know the stationary probability of each state. In that case, we also know how likely the detector is to observe a certain frame configuration (since each state is mapped to one or more frame configurations). We will also know how many bits that frame configuration will get us if we know what binning scheme we are using. We can thus calculate a weighted average to get R_{raw}. In this context, the stationary probability is the weights, and bits from each frame configuration are averaged.

However, to ensure the uniform randomness of the key, we need to find C_r^d, which will tell us by how much we need to compress the resulting bitstream to achieve perfect randomness. We now need to generate an OMC from the IMC. This involves mapping each Markov Chain state to a new state – or states – that correspond to any output bits. We will also need to find the stationary entropy of this new Markov Chain since we need it for the Markov Chain Entropy formula. This jump from the IMC to the OMC is necessary because if we applied the Markov Chain Entropy formula to an IMC, we would recover (16). To see how this may be done, consider Fig. 13 (an IMC) and Fig. 16 (its OMC).

For more information on the derivation of an OMC, we direct the reader to [22].

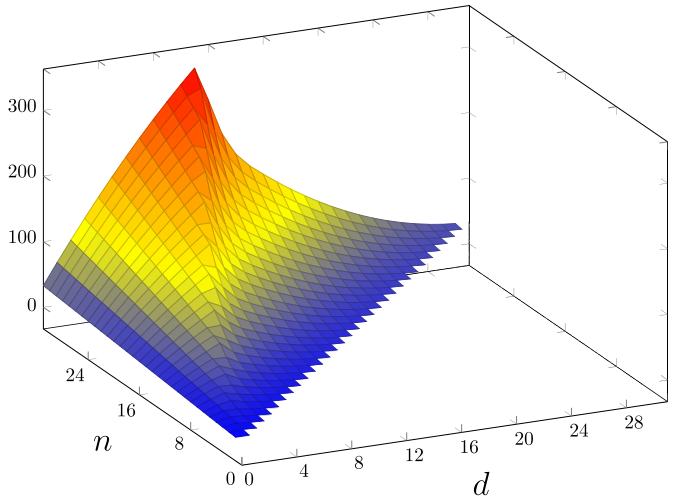

Fig. 15. A 3D plot of the minimum number of states generated in the BMCM (Basic Markov Chain Method) versus the RMCM (Reduced Markov Chain Method) as a function of n and d. Note how the two techniques appear to complement each other, producing a smaller number of states where the other produces a large amount. A ridge exists appearing to lie on the line where $d = n/2$, representing the boundary where the BMCM and RMCM generate approximately the same number of states. Points where $d > n$ are outside the domain of this analysis and so not included in the plot. Points on the plot are colored on a spectrum from blue to red based on their vertical height, which represents the state count.

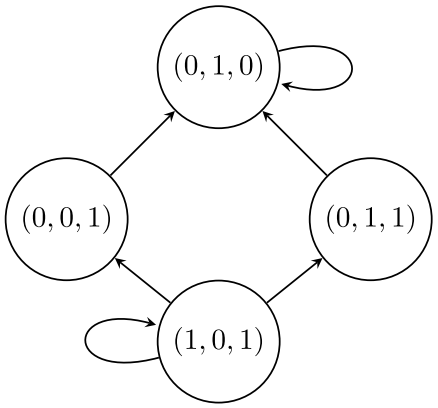

Fig. 16. Output Markov Chain (OMC) for the parameters $n = 4$ and $d = 1$, for the PPM binning scheme. Each state is labeled by a triplet of integers, (d_o, n_1, d_i), as described in section V-B. Transition probabilities are not shown. The method we use to derive the transitions begins with a state vector π for the IMC, where $\pi_i = 1$ when i corresponds to an IMC state that yields a non-zero number of bits. We apply the original transition matrix once. Then we switch to another transition matrix where all states that give a non-zero number of bits transition to themselves with probability 1. We then find the stationary probability using this alternate transition matrix. At the end of this step, all state vector elements π should be zero if the element corresponds to a state that yields no bits. The resulting probabilities of each state that yields bits are the transition probabilities from the state in the OMC corresponding to state i in the IMC to the corresponding states in the OMC.

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

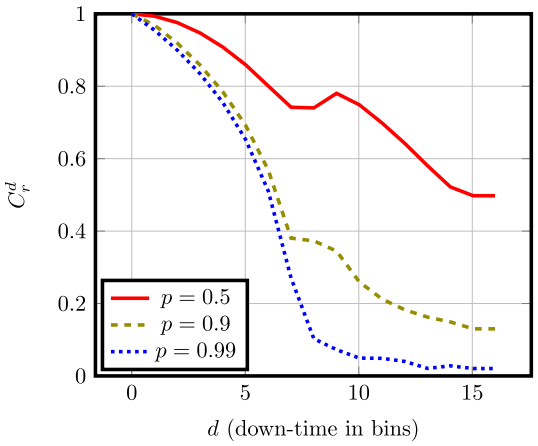

Fig. 17. C_r^d for $n = 16$ and when d varies over the x-axis. Values for p are chosen by how well they show the change in the curve versus the change in the value of p (word this better). This shows how C_r^d is increasingly affected by non-zero downtime as p approaches 1. The data was generated using our online tool that can be found at https://cc1539.github.io/qkd-binning-demo-2/.

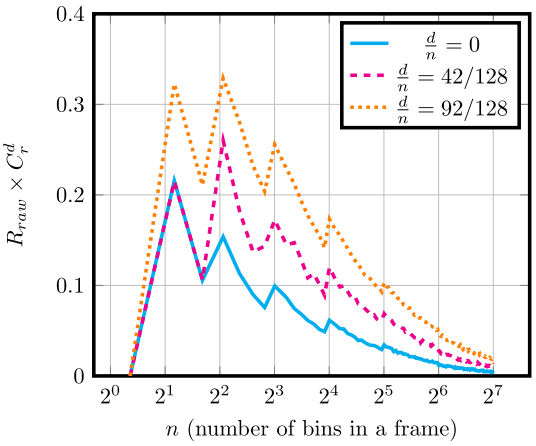

Fig. 18. $R_{\text{raw}} \times C_r^d$ for $\lambda_p \approx \frac{2.3}{\tau_b}$, with n varying over the x-axis from 0 to 128. Here, frame width in time is kept constant, such that if T_f is the duration of a frame, then $\tau_b = T_f/n$. The parameter d is also kept constant with respect to time, so the ratio $\frac{d}{n}$ is also constant. This expression for λ_p corresponds to a value of $p = 0.9$ for the case of $n = 1$. We are using PPM. The data was generated using our online tool available at the following URL: https://cc1539.github.io/qkd-binning-demo-2/.

Recall that the RMCM is useful if we are using PPM because we can tell how many bits every frame configuration represented by a state yields without "decompressing" it. The situation is similar to the OMC, where things are fine as long as we use PPM because, with PPM, every single frame in a state gets the same amount of bits (so instead of average, we really should say that the number we get is *the* number of bits we get for that state). More importantly, a unique input frame configuration gives that output for every possible binning scheme output, which is not the case for Simple Binning. For example, both 1000 and 0111 would yield the same output bits. These properties of PPM give us the luxury of not having to decompress the IMC to derive an accurate OMC. However, in the general case, more sophisticated techniques are required when we consider newer, more advanced binning schemes such as those described in [6]. In the worst case, the advantage of using the RMCM is not only negated but even subverted, since we will have to decompress it straight away in order to calculate even R_{raw}, but the "base" Markov chain is larger than that of the BMCM.

C. Observations

Recall how when p is small, C_r^d remains close to 1. This makes sense because in that case, photon inter-arrival times are, on average, spaced apart enough for downtime not to have a chance to change anything. It is when p approaches 1 (when λ_p is increased arbitrarily) that C_r^d falls. See Fig. 17.

For an intuitive explanation, consider a toy example where $n = 2$, $d = 1$, and $p = 1$. In this case, the detector will observe the sequence 1010101010 repeated indefinitely. Notice how since $n = 2$, every frame would be a valid one-shot encoding, which gives us one bit per every two bins, and if we did not know better, we would think this is great. However, the frame itself is repeating, so we repeatedly get that same bit, resulting in a perfectly deterministic sequence of all 0s or 1s. As long as d is non-zero, the same effect applies for all p, but usually much less depending on p.

If downtime was zero, we'd never see any of these decreases in C_r^d no matter how high we push λ_p. Does this mean we want $\frac{d}{n}$ to be as small as possible? This may not the case. This time, while considering C_r^d we instead neglected R_{raw}. See Fig. 18.

Notice how $R_{\text{raw}} \times C_r^d$ seems to actually be increasing as $\frac{d}{n}$ increases. To be clear, we *are* taking into account the loss of entropy due to downtime here, and yet the ultimate key rate is still going up as d increases. This indicates that somehow, the increase in the number of key bits caused by increasing downtime wins out against the decrease in the number of bits lost due to the decrease in C_r^d. So does this mean that we want d to be as large as possible? Perhaps we may want to change the binning scheme instead. Today's specifications (namely, λ_p being quite low) mean that we are not affected by downtime too seriously – recall that if p is small enough, d stops playing much of a role. However, as we can achieve greater λ_p, we may need newer, more efficient bit extraction techniques. Such schemes, which unlike PPM do not discard all frames with multiple arrivals, have been proposed [6].

VI. Modeling the Impact of Dark Counts

Occasionally, an SPD announces a photon arrival independent of an SPDC-generated photon event. This phenomenon is known as a dark count. Primarily, dark counts are due to light leakage into the quantum channel. However, even the best systems report a non-negligible dark count rate. Dark counts can cause two types of errors. In Type-One errors a dark count occurs within the same time frame as an SPDC-generated pair and invalidates an otherwise acceptable PPM frame. Thus, the frame's contribution to the key is lost. In Type-Two errors, frame-coincident dark count detections happen at the two stations, which causes Alice and Bob to retain frames with uncorrelated arrivals that are not due to entangled photons but are indistinguishable to them from such frames.

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

This section aims to model the impact of dark count arrivals on the system in light of these two potential types of errors. We first show the natural result that dark counts can decrease the probability of observing a PPM-valid time frame. We then model the frame-coincident dark count arrivals as large jitters. Type-One errors reduce the key rate by decreasing the number of PPM-valid frames. Type-Two errors reduce the key rate by forcing Alice and Bob to communicate more during the privacy amplification stage to reconcile the differences even in their uncorrelated frames.

We model the dark count arrival process at Alice/Bob's measurement station as a Poisson process with rate parameter λ_{DC}. Recall that the arrival process of SPDC-generated photons is also Poisson with rate λ_{p}. Therefore, we can determine the probability that Alice and Bob observe a frame to be PPM valid as the total probability that they each observe a frame with a single photon due to either photon source. From the properties of Poisson processes, we know that the probability of a single event, and zero events, within a time frame will respectively follow the form of

$$p_1 = \lambda T_f \exp\left(-\lambda T_f\right), \text{and } p_0 = \exp\left(-\lambda T_f\right). \tag{19}$$

By substituting the respective rate parameters into these two formulas, we can arrive at the total probability of observing the PPM frame described by

$$p_{\text{PPM}} = p_{\text{SPDC},1} \cdot p_{\text{DC},0}^2 + p_{\text{SPDC},0} \cdot p_{\text{DC},1}^2. \tag{20}$$

The first term in the (20) is the probability that a frame with a single SPDC-generated photon arrival is retained in the absence of Type-One dark count errors. The second term is the probability that a frame is considered valid due only to coincident dark counts. This represents the probability of a Type-Two dark count error. In each of these terms, the dark count probability component is squared to account for the fact that Alice and Bob must both register a dark count to retain the frame as a PPM frame. The SPDC probability is not squared as a single SPDC event generates both photons. Note that if only Alice's or only Bob's frame is corrupted, they both throw the corresponding frame away. Thus, in any scenario other than those described by the two terms in (20), Alice and Bob will throw away the frame. As the ratio of dark counts to SPDC photons increases, the rate of observing PPM valid time frames drops quickly. The decrease in key rate due to type-one dark count errors follows this same trend. Every frame lost corresponds to the loss of n raw key bits. Due to this phenomenon, it is incredibly important to minimize dark counts when using time-entanglement based QKD with PPM.

A. Modeling Dark Counts as Jitters

Using the probabilities found above, we can describe the *observed* timing jitter PDF as a weighted combination of jitter due to detector timing jitter *and* perceived jitter due to the arrivals of coincident dark counts. With this combined jitter error PDF, we can compute the loss of key rate much like in Section IV. Here we substitute the double Gaussian PDF with this new symmetric jitter PDF. In doing so, we can include the key loss due to over-correcting for coincident dark counts in the term β_r from (5). Despite these frames being entirely uncorrelated, if Alice and Bob cannot discern them from SPDC frames then in attempting to correct the errors they leak information about the keys. This reduces the secrecy of the keys including the truly correlated contributions, and thus reduces the reconciled key length.

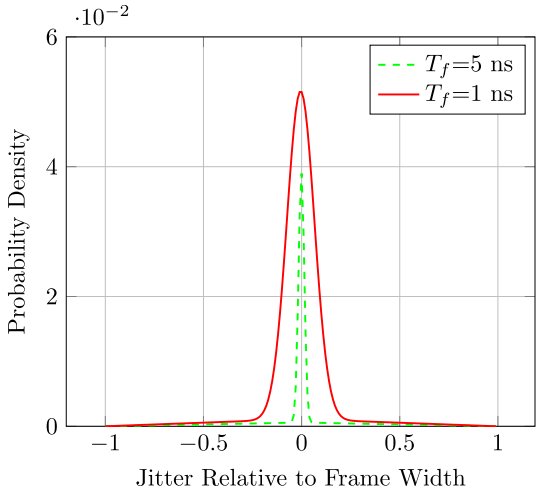

Fig. 19. Two frame configurations showing the theoretical model of observed jitter errors. The plots were generated using (21).

The dark counts are uniformly distributed within a given time frame. Thus, to determine the distribution of the observed jitter due to coincident dark counts, we convolve two uniform PDFs. This results in a triangular distribution with a peak at the jitter value 0 described by $\text{Tri}(x) = \frac{1}{T_f} - \frac{1}{T_f^2}|x|$. We consider this as a different type of observed jitter error. We can write the distribution of observed jitter errors as a weighted sum of the distribution of detector jitter errors and jitter errors due to coincident dark counts. The weighting term is a function of the frame size as it represents the ratio between the probability that a frame is PPM valid due to an SPDC photon pair to the total probability that a frame is PPM valid. For a specific frame size and the expected photon arrival rates, we get

$$\begin{aligned} p(t_j) &= c(T_f) N(0, 2\sigma_d^2) + (1 - c(T_f))\,\text{Tri}(t_j) \\ c(T_f) &= \frac{p_{\text{SPDC},1} \cdot p_{\text{DC},0}^2}{p_{\text{PPM}}} \\ &= \frac{p_{\text{SPDC},1} \cdot p_{\text{DC},0}^2}{p_{\text{SPDC},1} \cdot p_{\text{DC},0}^2 + p_{\text{SPDC},0} \cdot p_{\text{DC},1}^2}. \end{aligned} \tag{21}$$

We can see the effect of considering coincident dark counts on the jitter PDF in Fig. 19. Without considering dark counts, there is some motivation to increase time frame width to allow for increasing the width of time bins while maintaining a high bin count per frame. This approach would intuitively reduce the number of errors due to detector jitter, as any jitter would be more likely to land within the same bin. Here we see that if you are not careful in doing so, you can face different jitter errors due to an increased susceptibility to coincident dark counts. The Gaussian seems to dominate with smaller frame sizes, but we can observe heavier tails to the distribution with larger time frames. These heavy tails indicate

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

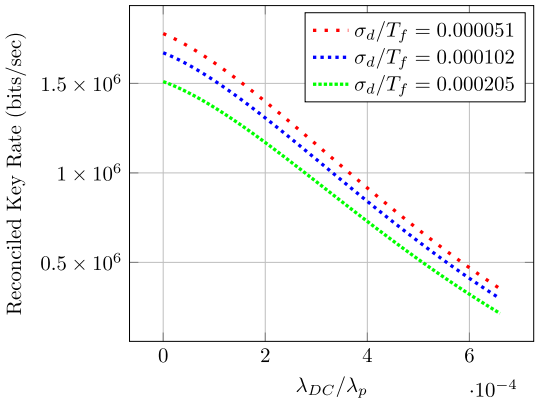

Fig. 20. These plots were generated using equations 19-21 to replace equation 13 with the same methods as used in Fig. 6. Here the reconciled key rate is labeled in MHz as we account for the probability that a frame is considered PPM valid. This is the key rate in time versus the earlier discussed key rate per PPM valid time frame.

a higher frequency of significant jitter errors resulting from the coincident dark count frames.

Finally, we applied the methods in Sec. IV with the above new jitter model to calculate the secret key rate loss due to information reconciliation. Fig. 20 shows the resulting plot in which we can see the post-reconciliation key rate in MHz as a function of the relative dark count rate to the SPDC generation rate. This effect is compounded by the reduced frame utilization, which reduces the post-reconciled key rate further. This evident trade-off further motivates the need for increasing the SPDC generation rate. However, if increased too much, the downtime effects may come into play. In the short term, these findings may be used to quantify expected experimental key rates given practical implementation details.

VII. Conclusion

Time-entanglement-based QKD seems a natural alternative to standard QKD when the goal is to increase the secret key rate in limited photon generation rates. However, in this model, some potentially catastrophic imperfections threaten to reduce the secrecy of generated keys. In this paper, we characterized several pitfalls related to time-of-arrival QKD. We provided mechanisms to accurately predict the possible secret key rate given the degree an experiment suffers from each impairment. First, we considered how timing measurement jitter introduces errors and necessitates information reconciliation, which decreases key length. Second, we accounted for the loss of entropy in generated keys due to detector downtime, which introduces memory to the system. We also analyzed dark counts, assuming that Alice and Bob cannot distinguish coincident dark count frames from SPDC photon arrival frames.

One of our most important contributions was developing a novel method to calculate entropy loss due to downtime. Under specific parameters, it is possible to simplify the Markov chain model of photon arrivals at a detector given detector downtime. This reduced model is more computationally tractable, allowing us to compute entropy reduction when working with frames with many time bins. An interactive online tool was developed in conjunction with this approach to enable researchers to explore the effect of binning scheme parametrization on the secret key rate in the light of downtime (https://cc1539.github.io/qkd-binning-demo-2/).

Future research in this space can expand on many of these aspects. Instead of considering various impairments separately and combining them afterward, one could think of them simultaneously. We see another interesting angle to consider joint information reconciliation and privacy amplification schemes. We hypothesize that compressing the raw key once for information reconciliation and again for privacy amplification may be overcompensating in some scenarios. That is, the value we would get from 6 – using parameters C_r^d and β_r as calculated in the ways we've described so far – may be lower than it needs to be. To reframe: we study information reconciliation in the context of jitter and privacy amplification in the context of downtime. We hope that accounting for jitter in our downtime analysis (which currently assumes a noiseless channel) may allow us to pull C_r^d closer to 1 without sacrificing key entropy. Additionally, we see potential in more advanced error-correcting codes for the information reconciliation phase. Specifically, we expect that layered codes will do well at dealing with multiple sources of errors. Jitter errors may be corrected separately from the coincident dark count errors, reducing the cost of information reconciliation on the secret key rate.

Acknowledgment

The authors thank the following colleagues: Murat Can Sarıhan for providing and explaining experimental data and Esmaeil Karimi and Phil Whiting for discussions on non-ideal detectors.

References

[1] R. Nandal, A. Nandal, K. Joshi, and A. K. Rathee, "A survey and comparison of some of the most prominent QKD protocols," *SSRN Electron. J.*, vol. 3, no. 5, pp. 1331–1333, May 2016.

[2] E. Diamanti, H.-K. Lo, B. Qi, and Z. Yuan, "Practical challenges in quantum key distribution," *npj Quantum Inf.*, vol. 2, pp. 1–12, Nov. 2016.

[3] A. K. Ekert, "Quantum cryptography based on Bell's theorem," *Phys. Rev. Lett.*, vol. 67, no. 6, pp. 661–663, Aug. 1991.

[4] T. Zhong et al., "Photon-efficient quantum key distribution using time–energy entanglement with high-dimensional encoding," *New J. Phys.*, vol. 17, no. 2, Feb. 2015, Art. no. 022002.

[5] H. Zhou and G. Wornell, "Adaptive pulse-position modulation for high-dimensional quantum key distribution," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2013, pp. 359–363.

[6] E. Karimi, E. Soljanin, and P. Whiting, "Increasing the raw key rate in energy-time entanglement based quantum key distribution," in *Proc. 54th Asilomar Conf. Signals, Syst., Comput.*, Nov. 2020, pp. 433–438.

[7] F. Xu, X. Ma, Q. Zhang, H.-K. Lo, and J.-W. Pan, "Secure quantum key distribution with realistic devices," *Rev. Mod. Phys.*, vol. 92, May 2020, Art. no. 025002.

[8] C. H. Bennett and G. Brassard, "An update on quantum cryptography," in *Advances in Cryptology* (Lecture Notes in Computer Science), vol. 196, G. R. Blakley and D. Chaum, Eds. Santa Barbara, CA, USA: Springer, Aug. 1984, pp. 475–480.

[9] J. Schneeloch et al., "Introduction to the absolute brightness and number statistics in spontaneous parametric down-conversion," *J. Opt.*, vol. 21, no. 4, Feb. 2019, Art. no. 043501.

[10] X. Cheng, M. C. Sarihan, K.-C. Chang, C. Chen, F. N. C. Wong, and C. W. Wong, "Secure high dimensional quantum key distribution based on wavelength-multiplexed time-bin encoding," in *Proc. Conf. Lasers Electro-Optics*, 2021, pp. 1–2.

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.

[11] S. Yang, M. C. Sarihan, K.-C. Chang, C. W. Wong, and L. Dolecek, "Efficient information reconciliation for energy-time entanglement quantum key distribution," in *Proc. 53rd Asilomar Conf. Signals, Syst., Comput.*, M. B. Matthews, ed., Pacific Grove, CA, USA, Nov. 2019, pp. 1364–1368.
[12] J. J. Boutros and E. Soljanin, "Time-entanglement QKD: Secret key rates and information reconciliation coding," 2023, *arXiv:2301.00486*.
[13] T. Brougham, C. F. Wildfeuer, S. M. Barnett, and D. J. Gauthier, "The information of high-dimensional time-bin encoded photons," *Eur. Phys. J. D*, vol. 70, no. 10, Oct. 2016.
[14] M. Fox, *Quantum optics: An Introduction* (Oxford master series in atomic, optical, and laser physics), Oxford, U.K.: Oxford Univ. Press, 2006.
[15] L. You et al., "Jitter analysis of a superconducting nanowire single photon detector," *AIP Adv.*, vol. 3, no. 7, Jul. 2013, Art. no. 072135.
[16] *Superconducting Nanowire Single-Photon Detectors*. Photon Spot Inc, Monrovia, CA, USA, 2022.
[17] I. E. Zadeh et al., "Superconducting nanowire single-photon detectors: A perspective on evolution, state-of-the-art, future developments, and applications," *Appl. Phys. Lett.*, vol. 118, no. 19, 2021, Art. no. 190502.
[18] U. Maurer, "Secret key agreement by public discussion from common information," *IEEE Trans. Inform. Theory*, vol. 39, no. 3, pp. 733–742, May 1993.
[19] A. E. Gamal and Y. Kim, *Network Information Theory*. Cambridge, U.K.: Cambridge Univ. Press, 2011.
[20] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 2006.
[21] M. Can Sarihan, "Private communications," Univ. California, Los Angeles, Los Angeles, CA, USA, Tech. Rep., Sep. 2021.
[22] C. Cheng, "The effect of detector recovery-time on time-entanglement QKD key rates and an online tool for calculating such rates," M.S thesis, Rutgers State Univ. New Jersey, School Graduate Stud., New Brunswick, NJ, USA, 2022.

Dunbar Paul Birnie IV received the B.S. degree in electrical engineering and computer science and the M.S. degree from Rutgers, The State University of New Jersey, USA, in 2020 and 2021, respectively. He is currently employed as a Software Engineer. His research interests are coding, covert communications, quantum computing, and machine learning.

Christopher Cheng received the B.S. degree in electrical engineering and computer science and the M.S. degree from Rutgers, The State University of New Jersey, USA, in 2020 and 2022, respectively. He is currently employed as a Software Engineer. His research interests are computational physics, coding, and machine learning.

Emina Soljanin (Fellow, IEEE) was a member of Technical Staff (Distinguished) for 21 years at the Mathematical Sciences Research of Bell Laboratories, before moving to Rutgers. Since 2016, she has been with Rutgers University, where she is currently a Professor. Her interests and expertise are broad. Over the past quarter of the century, she has participated in numerous research and business projects, including power system optimization, magnetic recording, color space quantization, hybrid ARQ, network coding, data and network security, distributed systems performance analysis, and quantum information theory. She served as an Associate Editor for Coding Techniques, the IEEE TRANSACTIONS ON INFORMATION THEORY, the Information Theory Society Board of Governors, and in various roles on other journal editorial boards and conference program committees. She was an outstanding alumnus at the Texas A&M School of Engineering, the 2011 Padovani Lecturer, a 2016/17 Distinguished Lecturer, and the 2019 President of the IEEE Information Theory Society.

Authorized licensed use limited to: Rutgers University. Downloaded on October 01,2023 at 16:03:20 UTC from IEEE Xplore. Restrictions apply.