Secure post-processing for non-ideal photon arrival time based quantum random number generator

Balázs Solymos, and László Bacsárdi, *Member, IEEE*

***Abstract*—Utilizing the inherently unpredictable nature of quantum mechanics, quantum random number generators (QRNGs) can provide randomness for applications where quality entropy (like in the case of cryptography) is essential. We present a post-processing scheme utilizing min-entropy estimation and hashing for optical QRNGs based on measuring individual photon arrival times. Our method allows for the handling of possible errors due to non-ideal components or even a potential attacker, given some basic assumptions to reliably produce a safe, good quality, uniformly distributed bitstream as output. We validate our results with an intentionally non-ideal measurement setup to show robustness, while also statistically testing our final output with four popular statistical test suites.**

***Index Terms*—quantum communication, quantum random number generation, statistical testing, hashing, entropy.**

I. Introduction

QUALITY randomness is used as a resource in a wide variety of applications, from numerical simulations to classical and even some quantum cryptography protocols [1], [2], that rely on entropy sources as fundamental building blocks. Due to this reliance, using a lower-quality source presents the danger of compromising the correctness of the schemes utilizing its output [3], especially in the field of cryptography. While pseudorandom number generators can provide fast and cheap random-like output, due to their inherently deterministic nature (use of complex but deterministic algorithms) are often considered a liability [4].

Quantum random number generators (QRNGs) [5] aim to harness the unpredictability of quantum mechanical processes. They have the advantage of relying on phenomena proved to be random by the laws of physics, thus giving a solid guarantee of quality in theory. Practical realization of these devices is a formidable engineering challenge, however, as the various imperfections and error sources potentially influencing the measurement have to also be handled. Due to advancements in quantum optics, architectures based on measuring various random properties of light, like path superposition of a photon [6], [7], photon number [8]–[10] or arrival time statistics [11]–[14], amplified spontaneous emission [15], [16], vacuum or phase fluctuations [17], [18], or even Raman scattering [19] have been proposed, while there are already some commercially available products on the market [20] and new chip-based solutions [21]–[23] are also emerging.

We use a simple generator architecture based on photon arrival times, with a continuously running clock, which is different from the ideal case of using a restartable clock but permits simpler and cheaper hardware. Even in the ideal case, the measurement statistics (exponential) differ from the expected uniformly distributed output, so a post-processing step is necessary. For this, various methods can be used, from simply comparing records [11] to utilizing complex algorithms based on entropy estimation and privacy amplification [24], [25]. In this work, we present a post-processing framework that can incorporate possible errors due to non-ideal components or even a potential attacker given some basic assumptions to reliably produce safe, quality output based on universal hashing and entropy estimation. Our framework potentially also enables us to relax minimum hardware requirements at the cost of output speed and the need for robust post-processing. This may prove especially useful for cases, where hardware options are limited either due to physical constraints (e.g., integrated optics), or any other reason (e. g. low budget to spend on quality components.).

II. Concept

A. Generator architecture

Our generator is based on time differences between photon arrival times of an attenuated laser source, counting the number of elapsed clock cycles between detections. Ideally, this statistic follows a geometric distribution, governed by the underlying exponential distribution of the physical process of photon emission, which is different from the expected uniform output, already mandating the need for post-processing. Additionally, effects from the concrete physical realizations and non-idealities further distort the measured statistic, making the generation of guaranteed quality output non-trivial.

In the following sections, we rely heavily on the concept of $H_\infty(D)$ min-entropy to characterize the safely extractable randomness from our measurement results:

$$H_\infty(D) = \min_n(-\log_2 p_n) = -\log_2 \max_n p_n, \qquad (1)$$

where $\max_n p_n = p_{\max}$ is the probability of the most likely measurement result. It is important to note, that attempting to create a uniform output corresponding to more entropy than contained in the measurement results, yields poor quality or even insecure output, while underestimating extractable entropy may only lead to suboptimal output rate, but preserves

The authors are with the Department of Networked Systems and Services, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, Budapest, Hungary.
(E-mail: solymosb@hit.bme.hu, bacsardi@hit.bme.hu)

B. Solymos thanks the support of the UNKP-22-3-II-BME-238 New National Excellence Program of the Ministry for Culture and Innovation from the source of the National Research, Development, and Innovation Fund. L. Bacsárdi thanks the support of the János Bolyai Research Scholarship of the Hungarian Academy of Sciences (Grant No. BO/00118/20).

DOI: 10.36244/ICJ.2024.1.2

quality. Our goal is, therefore, to give a safe lower bound for min-entropy (upper bound for p_{max}), which holds even in non-ideal conditions.

B. Hashing for post-processing

Universal hash functions can be used for post-processing [25], since, with them, we can construct a $(k_{\text{e}}, \epsilon, n_{\text{e}}, d_{\text{e}}, m_{\text{e}})$ extractor, so that

$$\text{Ext} : \{0,1\}^{n_{\text{e}}} \times \{0,1\}^{d_{\text{e}}} \mapsto \{0,1\}^{m_{\text{e}}} \tag{2}$$

for every probability distribution D on $\{0,1\}^{n_{\text{e}}}$ with at least $H_\infty(D) \geq k_{\text{e}}$ min-entropy, the probability distribution $\text{Ext}(D, U_{d_{\text{e}}})$ is ϵ-close statistically to the uniform distribution on $\{0,1\}^{m_{\text{e}}}$. This means, that with the help of a random $U_{d_{\text{e}}}$ seed of d_{e} bits, we can take a longer, but only partially random stream of n_{e} bits and create a smaller, but close to uniform output. The reusability of this seed is a crucial requirement for extractors (Since the randomness needed for continual reseeding would exceed the randomness extracted.). Fortunately, universal hash functions are proven to be strong extractors by the Leftover Hash Lemma [26], stating this reusability.

From these, we chose the popular Toeplitz hash to serve as a basis for our randomness extraction method. An $m_{\text{e}}+n_{\text{e}}-1$ bit long random seed is needed for initialization to construct a random Toeplitz matrix of $n_{\text{e}} \times m_{\text{e}}$. Then during operation, we split our data into n_{e} long input vectors, and one-by-one multiply them with the initialized random Toeplitz matrix to get m_{e} long output vectors, which we then assemble into an output bitstream. The k_{e} extractable entropy contained in the n_{e} long input defines the possible values for m_{e} according to

$$m_{\text{e}} = k_{\text{e}} + 2\log\epsilon. \tag{3}$$

Given a target ϵ, from $H_\infty(D)$ and n_{e} all the other parameters can be derived, so our goal is to present a framework for safely determining these.

C. Error sources

1) Additive noise: Coherent light sources based on stimulated emission like lasers are generally assumed to be Poissonian photon sources [27] (meaning exponentially distributed arrival time differences between photon emissions), due to the underlying physical working principle. In reality, photons from spontaneous emission (e.g. thermal effects) may also have a small superpoissonian contribution to the output distribution of the source, though this effect has been shown to be vanishing with increasing attenuation [28]. Still, we can model this unwanted process by introducing additional photon counts mixed with the ideal exponential statistics. This idea can be extended to include any additive error sources, like afterpulsing effects, or even a potential attacker.

With this in mind, we assume that our count statistic is made up of photons coming from an underlying true exponential source with C_{exp} number of independent counts for a given time period, responsible for the majority of the total counts, and a smaller at most C_{noise} amount of counts coming from noise processes or even potential attackers. This essentially means a limit on noise/attacker intensity, while also assuming an attacker is not capable of influencing photons from the trusted exponential photon source.[1]

The goal is to give a worst-case lower estimate for min-entropy. For this, we propose that there exists an interval series in the joint exponential and noise statistic for which

$$\begin{aligned}\underline{H_\infty(D)} &= -(C_{\text{exp}} - C_{\text{noise}}) \log_2 p'_{\text{max}} \\ &= -(C_{\text{exp}} - C_{\text{noise}}) \log_2 \left(\frac{p_{\text{max}} C_{\text{exp}} + C_{\text{noise}}}{C_{\text{exp}} - C_{\text{noise}}} \right)\end{aligned} \tag{4}$$

is a lower bound in min-entropy[2].

Let $S_0, S_1, ..., S_i, ..., S_{C_{\text{exp}}-1}$ be the arrival times of photons from our ideal source, with $D_0, D_1, ..., D_i, ..., D_{C_{\text{exp}}-1}$ cycle long measured intervals between them and note arrival times of noise/attacker photons with $N_0, N_1, ..., N_i, ..., N_{C_{\text{noise}}-1}$ Since we allow the noise to have any distribution and use any strategy, even allowing dependence on other counts, we do not consider the min-entropy contribution of intervals where noise counts are involved (see Fig. 1), only the entropy contribution of intervals from the sub-series $\{D_j\}_{j \in J}$, where $J = \{j \mid \nexists N_i : S_{j-1} < N_i < S_j\}$ (the intervals not affected by noise counts)

We also have to consider the possible distorting effect the noise can have on the overall distribution and, therefore, the distribution of our remaining considered series. From the point of min-entropy, this means the possible change of the original p_{max} to a new p'_{max} (change in the probability of the most frequent result). Assuming a worst-case scenario, additional noise counts can have the following effects:

- Noise is positioned so that all original counts corresponding to p_{max} (originally most likely outcome of the ideal source) are kept in the considered sub-series.
- Interval statistics from noise counts further increase p_{max} for at most an additional C_{noise} new counts contributing to the measurement result corresponding to p_{max}.

While the actual physical feasibility of these worst-case effects may at times be questionable, we still consider them, to give a safe lower estimate guaranteed to hold for any possibility. This way, Eq. (4) is a lower bound in min-entropy for the considered sub-series, therefore it is a valid lower bound for the whole series too.

2) Effect of continuous clock and dead time: We can model the effects of using a continuous clock and dead time of the detector on the min-entropy as previously presented in more detail in [29]. Assuming photons arrive according to a Poisson process with rate λ, in the continuous clock case we can divide time into consecutive τ long grids, where τ is the length of a clock cycle, S_i the time of the ith arrival, $T_i = S_i - S_{i-1}$ the ith inter-arrival and γ_i the time between S_i and its preceding τ grid ($0 \leq \gamma_i < \tau$). We measure D_i, the number of τ grids (clock cycles) between S_{i-1} and S_i. An explanatory example case of the model can be seen in Fig. 2. For the distribution

[1] This also means, that, quantum operations, like entangling additional photons with photons from the trusted source, are not allowed either.

[2] For larger values of C_{noise}, where $p_{\text{max}} C_{\text{exp}} + C_{\text{noise}} > C_{\text{exp}} - C_{\text{noise}}$ Eq. (4) can result in negative output. $\underline{H_\infty(D)}$ should be considered 0 in these cases.

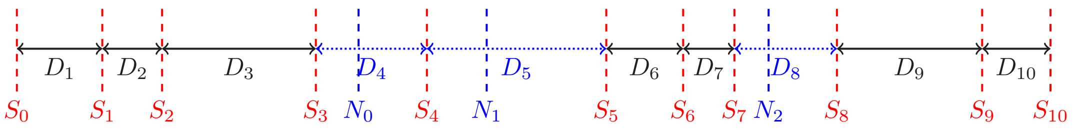

Fig. 1. Example of handling additive noise. Times noted with S_i are counts from the assumed underlying ideal distribution, with D_i intervals between them. After introducing N_i noise counts, we only consider the entropy contribution of intervals not affected by the noise, which are $\{D_1, D_2, D_3, D_6, D_7, D_9, D_{10}\}$ in this pictured example case.

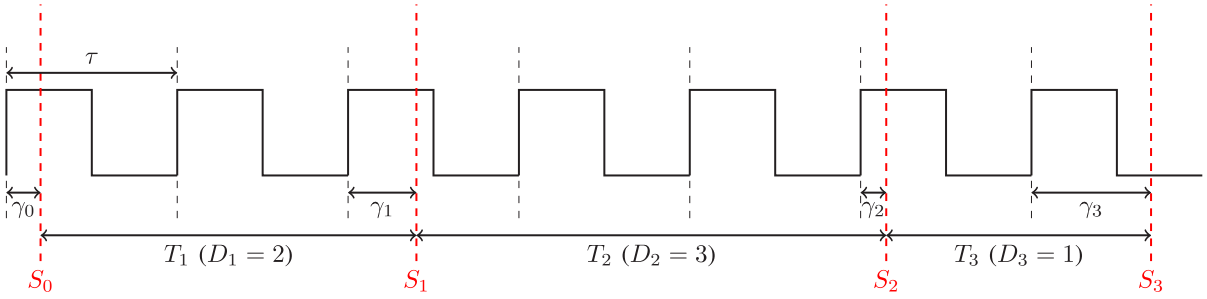

Fig. 2. Continuous clock example from [29]. Photons arrive at times S_i and are counted by a τ resolution running clock. T_i notes the true exponential time between detections and D_i the associated number of counts (our measurement result), while γ_i notes the varying internal starting phases of the counting process.

of D without dead time, we can write

$$\begin{aligned} p_n &= \Pr(D = n \mid \gamma = y) \\ &= \begin{cases} \Pr(y + T < \tau) & \text{if } n = 0, \\ \Pr(n\tau \le y + T < (n+1)\tau) & \text{if } n > 0, \end{cases} \\ &= \begin{cases} 1 - \mathrm{e}^{-\lambda(\tau - y)} & \text{if } n = 0, \\ \left(1 - \mathrm{e}^{-\lambda\tau}\right)\mathrm{e}^{-\lambda(n\tau - y)} & \text{if } n > 0. \end{cases} \end{aligned} \tag{5}$$

To calculate worst-case min-entropy we then maximize p_n:

$$\begin{aligned} &\max_{n,y}(\Pr(D = n \mid \gamma = y)) \\ &\quad = \max_{n,y}\begin{cases} 1 - \mathrm{e}^{-\lambda(\tau - y)} & \text{if } n = 0, \\ \mathrm{e}^{\lambda y}\left(1 - \mathrm{e}^{-\lambda\tau}\right)\mathrm{e}^{-\lambda n\tau} & \text{if } n > 0, \end{cases} \\ &\quad = \max_{n,y}\begin{cases} \left(1 - \mathrm{e}^{-\lambda\tau}\right) & \text{if } n = 0, y \to 0, \\ \mathrm{e}^{\lambda\tau}\left(1 - \mathrm{e}^{-\lambda\tau}\right)\mathrm{e}^{-\lambda n\tau} & \text{if } n > 0, y \to \tau, \end{cases} \\ &\quad = \max_{n,y}\begin{cases} 1 - \mathrm{e}^{-\lambda\tau} & \text{if } n = 0, y \to 0, \\ 1 - \mathrm{e}^{-\lambda\tau} & \text{if } n = 1, y \to \tau, \end{cases} \\ &\quad = 1 - \mathrm{e}^{-\lambda\tau}, \end{aligned} \tag{6}$$

so then the min-entropy is:

$$H_\infty(D) = -\log_2\left(\max_{n,y} p_n\right) = -\log_2\left(1 - \mathrm{e}^{-\lambda\tau}\right). \tag{7}$$

Dead time is a time of detector insensitivity after successful photon detection. Assuming τ_d dead time to be in the form: $\tau_\mathrm{d} = k\tau + \delta$, where k is a non negative integer and $0 \le \delta < \tau$, we can rewrite Eq. (5):

$$\begin{aligned} &\Pr(D = n \mid \gamma = y) \\ &= \begin{cases} 0 \text{ if } n < k, \\ \Pr(y + T + \delta < \tau) \text{ if } n = k, \\ \Pr\left((n-k)\tau \le y + T + \delta < (n-k+1)\tau\right) \text{ if } n = k+1, \\ \Pr\left((n-k)\tau \le y + T + \delta < (n-k+1)\tau\right) \text{ if } n > k+1, \end{cases} \\ &= \begin{cases} 0 \text{ if } n < k, \\ \begin{cases} 1 - \mathrm{e}^{-\lambda(\tau - y - \delta)} & \text{if } y < \tau - \delta, n = k, \\ 0 & \text{if } y \ge \tau - \delta, n = k, \end{cases} \\ \begin{cases} \mathrm{e}^{-\lambda(\tau - y - \delta)}\left(1 - \mathrm{e}^{-\lambda\tau}\right) & \text{if } y < \tau - \delta, n = k+1, \\ 1 - \mathrm{e}^{-\lambda(2\tau - y - \delta)} & \text{if } y \ge \tau - \delta, n = k+1, \end{cases} \\ \left(\mathrm{e}^{-\lambda((n-k)\tau - y - \delta)}\right)\left(1 - \mathrm{e}^{-\lambda\tau}\right) \text{ if } n > k+1. \end{cases} \end{aligned} \tag{8}$$

Maximizing p_n for min-entropy, we then get:

$$\begin{aligned} H_\infty(D) &= -\log_2 \max_{n,y,\tau_\mathrm{d}}(\Pr(D = n \mid \gamma = y)) \\ &= -\log_2\left(1 - \mathrm{e}^{-\lambda\tau}\right), \end{aligned} \tag{9}$$

which is the same result as in the case without dead time.[3] Since this result is also the same as in the ”restartable clock without dead time” case [30], we conclude, that using a continuous clock has no adverse effect on extractable min-entropy.

Dead time also has an effect on the detectable photon rate, since during τ_d no detections are possible. Since the bound for min-entropy is calculated using the original λ, and not the λ_d observed rate with dead time, we have to account for this, giving[4]:

$$\lambda = \frac{\lambda_{\max}}{1 - \lambda_{\max}\tau_\mathrm{d}}. \tag{10}$$

[3] Note that in this calculation of $H_\infty(D)$ we do not restrict τ_d in any way as in (9) we maximize over all possible τ_d. Due to this, $H_\infty(D) = -\log_2\left(1 - \mathrm{e}^{-\lambda\tau}\right) \le H_\infty(D \mid \tau_\mathrm{d} = Z)$ will hold for any possible Z distribution of τ_d.

[4] Note, that λ_d is maximized in $1/\tau_\mathrm{d}$, so the nominator here always stays positive.

3) Fluctuating λ: The actual value of λ may fluctuate due to various physical imperfections. Since $H_\infty(D)$ is monotonic in λ we can give a lower bound $H_\infty(D)_L$ for min-entropy if we know a $\lambda_{\max}$ upper bound for λ, such that

$$\begin{aligned} H_\infty(D)_L &= -\log_2\left(1-\mathrm{e}^{-\lambda\tau}\right) \le H_\infty(D) \\ &= -\log_2\left(1-\mathrm{e}^{-\lambda\tau}\right). \end{aligned} \tag{11}$$

$$p_{\max} = 1-\mathrm{e}^{-\lambda\tau} \tag{12}$$

This also means, that by giving an upper bound for τ_d in Eq. (10), we also upper bound λ and lower bound the min-entropy, so this way we can also account for unknown dead time distributions as long as a maximum value for τ_d is known.

4) Attenuation and detector efficiency: The μ quantum efficiency of detectors (the probability of successfully detecting an incoming photon) is analogous to the T_a transmissivity of attenuators. Due to the memoryless property of the exponential distribution, we can account for these effects so that our detected photons arrive according to an $\mathrm{Exp}(\lambda T_\mathrm{a}\mu)$ distribution.

D. Framework for extractor parameter selection

To give a combined min-entropy lower bound for a case containing all the previously investigated noise effects, we can use the following steps:

1) Apply methods from Sections II-C2 and II-C3 to account for the effects of dead time and fluctuations in λ to get a lower bound for min-entropy and, therefore, an upper bound for $p_{\max}$ of the individual intervals corresponding to the ideal operation of the source.[5]
2) Use Eq. (4) with this $p_{\max}$ and appropriately selected $C_\mathrm{noise}, C_\mathrm{exp}$ values to calculate the final $\underline{H_\infty(D)}$ for the interval series corresponding to the n_e bit long extractor input.

Note that in the second step, we use the overestimation of $p_{\max}$ of the intervals corresponding to C_exp. The reasoning presented in Sec. II-C1 is still valid as potential dependence between intervals due to non-ideal effects during measurement of photons of the ideal source is already accounted for in the overestimated $p_{\max}$ (see Sec. II-C2), therefore, the ability to sum interval min-entropies in Eq. 4 remains.

Also note, that counts from additive noise sources raise the experimentally detected λ, but this overestimation of λ does not lead to any additional security weaknesses, as presented in Sec. II-C3.

Other than min-entropy, we need to choose n_e to fully parameterize the extractor. Generally, choosing n_e to be larger is advantageous, as the scheme becomes more robust against bursty noise, as well as providing a better output ratio for a given ϵ according to Eq. (3). This comes at a cost of increased computational need, however.

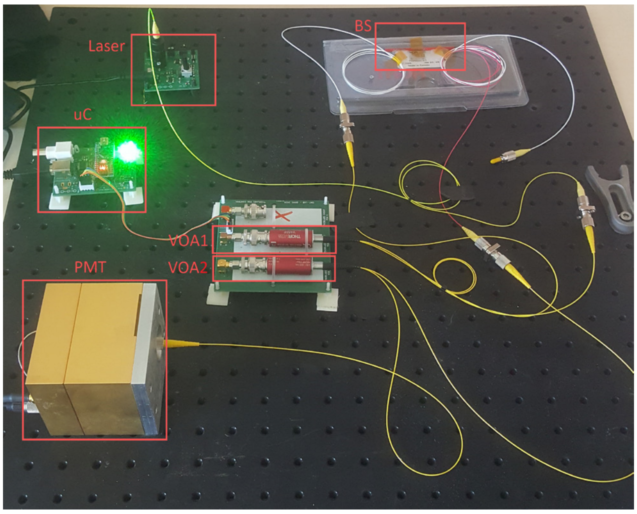

Fig. 3. Photo of the physical setup. uC: microcontroller controlling VOAs, BS: beam splitter, VOA: variable optical attenuator, PMT: photomultiplier tube. Photons travel along the Laser-VOA1-BS-VOA2-PMT optical path.

III. Experiment

A. Physical setup

Our physical setup presented in Fig. 3 is the same as in [31] and [29]. A Thorlabs LP520-SF15 semiconducting laser (central wavelength 519.9 nm) is attenuated using two successive voltage-controlled variable optical attenuators (Thorlabs V450F) and an optical splitter (Thorlabs TW560R1F1), where the splitter functions as an additional 20 dB attenuator. Photons are then detected by a PicoQuant PMA-175 NANO photomultiplier tube with a $\mu = 21\%$ quantum efficiency. The detector's output voltage pulses are time-tagged by a PicoQuant TimeHarp 260 time-to-digital converter (TDC) card with a base resolution of $\tau = 250$ ps integrated into the PC controlling the measurement and running post-processing. Our detection system (detector and TDC) has a dead time of around 2 ns, very low afterpulsing probability ($\sim 0\%$), and measured dark count rates around 1-10 cps.

B. Parameter selection

We collect and process 2×10^{10} intervals to investigate the validity of our presented framework. During data acquisition, the measured detection rate was around $\lambda_\mathrm{d} = 1.3\times10^6$ cps (counts per second) and between $\lambda_{\min} = 1.08\times10^6$ cps and $\lambda_{\max} = 1.37\times10^6$ cps at all times. We chose not to try mitigating this fluctuation as our goal is to show robustness. We also completely forego using available protective covers made for severely limiting counts from the environment, leading to a $\lambda_\mathrm{n} = 20000$ cps noise rate at our detector. We overestimate our relatively low detector dead time of around 10τ with a conservative $\tau_\mathrm{d} = 50\tau$. In practice, afterpulsing effects are often neutralized by the longer detector dead times compared to them, which is also the case for our hardware, showing negligible afterpulsing probability. To present an example of handling this effect in our framework, we assume a maximum probability of counts caused by afterpulsing of

[5] Since we usually base estimation on measurement results of the detector, attenuation from attenuators and detector efficiency discussed in II-C4 are already included in these.

$P_{\mathrm{after}} = 10^{-4}$ nonetheless. Due to the quality of laser sources, another quantity often considered experimentally negligible is the number of photons created in the source not via stimulated emission. Similarly to the previous case of afterpulsing, this effect could be considered negligible in our setup, but we still assume an exemplary maximum probability for it to be $P_{\mathrm{nonstim}} = 10^{-6}$.

Utilizing our framework presented in Sec. II-D, we can calculate a lower bound for min-entropy using the presented measurement parameters: First, calculate λ from λ_{max} according to Eq. (10), giving $\lambda = 1.3215 \times 10^{6}$ cps. From this, calculate $p_{\mathrm{max}} = 1 - \mathrm{e}^{-\lambda\tau} = 3.3031 \times 10^{-4}$. Utilizing that count numbers in Eq. (4) can be expressed in terms of detection rate over the investigated timeframe, we can use $\lambda_{\mathrm{id}} = \lambda_{\mathrm{min}}(1 - P_{\mathrm{nonstim}})(1 - P_{\mathrm{after}}) - \lambda_{\mathrm{n}}$ to underestimate the number of counts from the ideal source, and $\lambda_{\mathrm{noise}} = \lambda_{\mathrm{n}} + P_{\mathrm{nonstim}}\lambda_{\mathrm{max}} + P_{\mathrm{after}}\lambda_{\mathrm{max}}$ to overestimate noise.

To be able to determine the C_{noise} counts in a processed data block, we have to first choose n_{e}. As stated before, higher values are beneficial, but since our hashing implementation currently runs on CPU and not on dedicated hardware as its main goal is to serve as proof of concept, we settle for $n_{\mathrm{e}} = 2048$ bits, due to our limited computational resources. With 16-bit long measurement records, 128 records are processed together at once in a block. This means an average C_{noise} of 3 and C_{exp} of 125 (To additionally protect from burst errors C_{noise} can be chosen to be higher if needed.). Using (4) this yields $\underline{H_\infty(D)} = k_{\mathrm{e}} = 649.682$ bits for an $m_{\mathrm{e}} = 544$ bits with $\epsilon = 2^{-52.841} < 2^{-50}$.

For initialization of the Toeplitz hash algorithm (to create the $n \times m$ Toeplitz matrix), we need a $d_{\mathrm{e}} = n_{\mathrm{e}} + m_{\mathrm{e}} - 1 = 2479$ bit long random string, which can come from a different trusted source or can even be a "baked in" string due to its reusability. We used random data collected during a previous different experiment [31] with our setup for initialization.

To further test our framework, we modified our initial measurement record file by artificially inserting counts every 50000 cycles simulating a perfect periodic noise/attacker (and thereby considerably changing the detected distribution too, as there can be no recorded time intervals longer than this inserted periodicity). This accounted for an additional noise source with a 80000 cps rate. Accounting for this (change in λ_{d}, λ_{max}, λ_{min}, λ_{n}), the newly calculated parameters for the hash function, in this case, are: $n_{\mathrm{e}} = 2048$ bits, $k_{\mathrm{e}} = 246.8593$ bits, $m_{\mathrm{e}} = 144$ bits for an $\epsilon < 2^{-51}$ and 1.125 output bits per measurement record accordingly. Note the heavily reduced output efficiency, which is mainly due to the increased unknown noise considered according to Sec. II-C1.

C. Randomness testing

We assess our output files of 8.4 GB and 2.8 GB for the previously mentioned measurement cases with four of the most widely used statistical test suites, namely the NIST STS [32], Dieharder [33], TestU01 [34] and ENT [35] suites. Statistical tests typically try to refute the hypothesis that a source is random, by looking for signs of different kinds of possible non-randomness. Suites are, therefore, composed of batteries of individual tests, each looking for different non-random patterns. Due to the fact that a properly random output contains every possible string, a good generator is also expected to fail some proportion of these tests, so verifying proper operation is tricky and cannot be based on test results alone. To demonstrate this, we also tested unprocessed and not properly parametrized processed versions of our initial measurement data. Still, statistical testing of the output is a handy tool for checking for potential oversights or implementation errors (An uncharacteristically poor performance on tests almost surely indicates some error in operation.).

Results from the NIST STS suite for our first output file are shown in Table I omitting variants of the *NonOverlappingTemplate*, *RandomExcursions* and *RandomExcursionsVariant* tests as these are families of multiple tests producing too many results to be easily presentable in table format. We ran the suite with default settings and 2048 streams to test for both of our files. According to the manual, a case is considered passing if at least 2014 of the streams pass. We found that our data passed all the tests in the assessment.

TABLE I
Results for NIST STS tests

Test Name	p-value	Proportion	Assessment
Frequency	0.4564	2032/2048	Pass
BlockFrequency	0.7979	2031/2048	Pass
CumulativeSums 1	0.9195	2029/2048	Pass
CumulativeSums 2	0.1850	2025/2048	Pass
Runs	0.5862	2025/2048	Pass
LongestRun	0.6920	2026/2048	Pass
Rank	0.7041	2021/2048	Pass
DFT	0.5450	2022/2048	Pass
OverlappingTemplate	0.1885	2031/2048	Pass
Universal	0.1608	2024/2048	Pass
ApproximateEntropy	0.3294	2025/2048	Pass
Serial 1	0.7997	2025/2048	Pass
Serial 2	0.2548	2028/2048	Pass
LinearComplexity	0.1053	2027/2048	Pass

The Dieharder suite is a collection of many tests expanding upon the original Diehard tests [36]. We present results for our first measurement case from these original (Diehard) tests in Table II. [6] The suit additionally contains other tests, which our data also successfully passed for both of the output files.

We used the *Alphabit* and *Rabbit* batteries recommended for use with hardware RNGs as well as the *SmallCrush* test battery from the TestU01 software library to assess our data. Results from the *SmallCrush* battery for the first output file are presented in Table III. We found that both of our data files passed all these assessments.

The ENT program can test random files in *byte* and *bit* modes, and calculates statistics like symbol occurrences, entropy, approximation of π, and correlation, to assess the randomness of a bitstream. Our files passed these assessments in both modes.

[6] Due to the occasional expected test failures of proper random operation, the dieharder suite also has a *WEAK* assessment result, where the manual advises further investigation. In our case, the tests *diehard_squeeze* and *diehard_sums* originally produced this result, so we ran them with lengthier than standard input data for a stronger examination to make sure of correctness and found them passing.

TABLE II
RESULTS FOR DIEHARD TESTS

Test Name	p-value	Assessment
diehard_birthdays	0.48117807	Pass
diehard_operm5	0.72724586	Pass
diehard_rank_32x32	0.18749969	Pass
diehard_rank_6x8	0.20228745	Pass
diehard_bitstream	0.13044230	Pass
diehard_opso	0.92784321	Pass
diehard_oqso	0.091542557	Pass
diehard_dna	0.60242889	Pass
diehard_count_1s_str	0.30601543	Pass
diehard_count_1s_byt	0.63839715	Pass
diehard_parking_lot	0.91059716	Pass
diehard_2dsphere	0.18188938	Pass
diehard_3dsphere	0.91144842	Pass
diehard_squeeze	0.51632824	Pass
diehard_sums	0.03411320	Pass
diehard_runs 1	0.90873329	Pass
diehard_runs 2	0.86353873	Pass
diehard_craps 1	0.86019516	Pass
diehard_craps 2	0.39312891	Pass

TABLE III
RESULTS FOR TESTU01 TESTS

Test Name	p-value	Assessment
smarsa_BirthdaySpacings	0.86	Pass
sknuth_Multinomial	0.60	Pass
sknuth_Gap	0.76	Pass
sknuth_SimpPoker	0.74	Pass
sknuth_CouponCollector	0.61	Pass
sknuth_MaxOft 1	0.73	Pass
sknuth_MaxOft 2	0.85	Pass
svaria_WeightDistrib	0.82	Pass
smarsa_MatrixRank	0.13	Pass
sstring_HammingIndep	0.97	Pass
swalk_RandomWalk1 H	0.78	Pass
swalk_RandomWalk1 M	0.47	Pass
swalk_RandomWalk1 J	0.20	Pass
swalk_RandomWalk1 R	0.10	Pass
swalk_RandomWalk1 C	0.80	Pass

To demonstrate the nature of statistical testing and the need for proper analysis in addition to passing test results, we also tested unprocessed data (binary datafile only containing the unprocessed 16-bit records) and an additional test case, where we incorrectly parametrized the hash function with $m_{\text{e}} = 2048$. For the first unprocessed case, the ENT test already showed some weaknesses, with an estimated entropy of 7.247 bits per byte (well above the estimated min-entropy in Sec. III-B, below expected 8 of ideal uniform output), and compressibility of 9 percent, while all the other test suites summarily failed the data (which is expected since raw measurement data correspond to a not uniform distribution). Interestingly, the wrongly parametrized processed data also passed our statistical trials, demonstrating, that passing the tests is not a guarantee for secure randomness in itself. This is probably due to the fact that the hashing operation in itself shows behavior similar to pseudo-random number generators, as its main aim is to produce random-like output from any input. While this wrongly parametrized output is clearly not suitable for quality and security-critical use cases, it may still prove useful in cases with less strict output quality criteria, essentially enabling an operation mode realizing a rapidly reseeded pseudo-random number generator, with higher output efficiency. We leave further investigation of such a scheme up for future study.

D. Achievable output rates

The achievable output efficiency and, consequently, the final output rate are heavily influenced by the magnitude of noise effects. Fig. 4. shows that in our test setup, increasing noise can lead to cases where we can no longer guarantee our goal ϵ for any parameter set (from $\lambda_{\text{noise}} \geq 219352$), or even any secure output at all (from $\lambda_{\text{noise}} \geq 354339$). Furthermore, introducing even small amounts of noise to the system leads to a steep decline in the achievable output rate. For the completely noiseless case, our test setup would have an efficiency of 10.6875 output bits per record, leading to a theoretical max output speed of 13.863 Mbps, while introducing only the example noise from afterpulsing effects and photons not from stimulated emission ($\lambda_{\text{noise}} = 138$) already drops efficiency to 6 output bits per record and output speed to 7.8 Mbps. The two noisy example cases presented before at the start and end of Section III-B have output efficiencies of 4.25 and 1.6875 bits per record and output speeds of 6.598 Mbps and 2.193 Mbps, respectively.

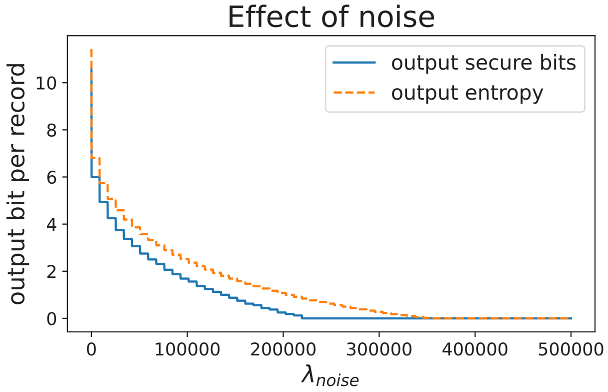

Fig. 4. Effect of different λ_{noise} noise intensities on achievable output bit and entropy rates, with the parameter set presented at the beginning of Sec. III-B, while maintaining $\epsilon < 2^{-50}$.

Unfortunately, our current practical implementation presents a computational bottleneck of $\sim 10^5$ records processed per second, limiting our current practically achievable output speeds. This can likely be overcome in the future with a new implementation utilizing either an FPGA or GPU as it has been demonstrated in the literature [37], [38] and therefore, the implementation of a new post-processing program is our next logical practical goal. Better and stricter characterization of possible noise sources is another worthwhile direction to pursue for possible future development, especially for cases with concrete, well-characterized measurement setups, as it may be possible to find tighter lower bounds than Eq. (4) when using less general assumptions.

IV. CONCLUSION

We presented a post-processing framework for optical QRNGs based on the measurement of photon arrival times, that can be used to safely account for typical distortion effects and hard-to-characterize error sources or attackers given a simple upper limitation on intensity, by strictly underestimating the min-entropy of the measurement results and utilizing this estimate to parameterize a Toeplitz hash-based extractor to provide a guaranteed quality, safe output bitstream. We demonstrated the use of our framework on intentionally non-ideal measurement data, showing its robustness, and assessed the processed outputs with statistical test suites to experimentally verify our proposal's correctness.

We conclude that our method can be used to provide quality output even when paired with noisy and imperfect measurement setups, although at a cost of reduced output efficiency. This drop in efficiency is especially prevalent when adjusting for the effects of error sources considered as unknown, so in practical realizations minimizing or adequately characterizing these should still remain a priority with our framework too.

References

[1] L. Gyongyosi, L. Bacsardi, and S. Imre, "A survey on quantum key distribution," *Infocommunications Journal*, no. 2, pp. 14–21, 2019. [Online]. Available: **doi**: 10.36244/icj.2019.2.2

[2] D. Chandra, P. Botsinis, D. Alanis, Z. Babar, S.-X. Ng, and L. Hanzo, "On the road to quantum communications," *Infocommunications Journal*, vol. 14, no. 3, pp. 2–8, 2022. [Online]. Available: **doi**: 10.36244/icj.2022.3.1

[3] Y. Dodis, S. J. Ong, M. Prabhakaran, and A. Sahai, "On the (im) possibility of cryptography with imperfect randomness," in *45th Annual IEEE Symposium on Foundations of Computer Science*. *IEEE*, 2004, pp. 196–205. [Online]. Available: **doi**: 10.1109/FOCS.2004.44

[4] N. Heninger, Z. Durumeric, E. Wustrow, and J. A. Halderman, "Mining your ps and qs: Detection of widespread weak keys in network devices," in *Presented as part of the 21st {USENIX} Security Symposium ({USENIX} Security 12)*, 2012, pp. 205–220. [Online]. Available: **doi**: 10.5555/2362793.2362828

[5] M. Herrero-Collantes and J. C. Garcia-Escartin, "Quantum random number generators," *Reviews of Modern Physics*, vol. 89, no. 1, p. 015004, 2017. [Online]. Available: **doi**: 10.1103/RevModPhys.89.015004

[6] T. Jennewein, U. Achleitner, G. Weihs, H. Weinfurter, and A. Zeilinger, "A fast and compact quantum random number generator," *Review of Scientific Instruments*, vol. 71, no. 4, pp. 1675–1680, apr 2000. [Online]. Available: **doi**: 10.1063/1.1150518

[7] A. Stefanov, N. Gisin, O. Guinnard, L. Guinnard, and H. Zbinden, "Optical quantum random number generator," *Journal of Modern Optics*, vol. 47, no. 4, pp. 595–598, mar 2000. [Online]. Available: **doi**: 10.1080/09500340008233380

[8] M. Fürst, H. Weier, S. Nauerth, D. G. Marangon, C. Kurtsiefer, and H. Weinfurter, "High speed optical quantum random number generation," *Optics Express*, vol. 18, no. 12, p. 13029, jun 2010. [Online]. Available: **doi**: 10.1364/oe.18.013029

[9] S. Tisa, F. Villa, A. Giudice, G. Simmerle, and F. Zappa, "High- speed quantum random number generation using CMOS photon counting detectors," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 21, no. 3, pp. 23–29, may 2015. [Online]. Available: **doi**: 10.1109/jstqe.2014.2375132

[10] M. Ren, E. Wu, Y. Liang, Y. Jian, G. Wu, and H. Zeng, "Quantum random-number generator based on a photon-number-resolving detector," *Physical Review A*, vol. 83, no. 2, feb 2011. [Online]. Available: **doi**: 10.1103%2Fphysreva.83.023820

[11] M. Stipčević and B. M. Rogina, "Quantum random number generator based on photonic emission in semiconductors," *Review of scientific instruments*, vol. 78, no. 4, p. 045104, 2007. [Online]. Available: **doi**: 10.1063/1.2720728

[12] H.-Q. Ma, Y. Xie, and L.-A. Wu, "Random number generation based on the time of arrival of single photons," *Applied optics*, vol. 44, no. 36, pp. 7760–7763, 2005. [Online]. Available: **doi**: 10.1364/ao.44.007760

[13] M. A. Wayne, E. R. Jeffrey, G. M. Akselrod, and P. G. Kwiat, "Photon arrival time quantum random number generation," *Journal of Modern Optics*, vol. 56, no. 4, pp. 516–522, 2009. [Online]. Available: **doi**: 10.1080/09500340802553244

[14] N. Massari, L. Gasparini, M. Perenzoni, G. Pucker, A. Tomasi, Z. Bisadi, A. Meneghetti, and L. Pavesi, "A compact tdc-based quantum random number generator," in *2019 26th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2019, pp. 815–818. [Online]. Available: **doi**: /10.1109/icecs46596.2019.8964941

[15] C. R. Williams et al., "Fast physical random number generator using amplified spontaneous emission," *Optics Express*, vol. 18, no. 23, pp. 23 584–23 597, 2010. [Online]. Available: **doi**: 10.1364/oe.18.023584

[16] Á. Marosits, Á. Schranz, and E. Udvary, "Amplified spontaneous emission based quantum random number generator," *Infocommunications Journal*, vol. 12, no. 2, pp. 12–17, 2020. [Online]. Available: **doi**: 10.36244/icj.2020.2.2

[17] M. Jofre, M. Curty, F. Steinlechner, G. Anzolin, J. P. Torres, M. W. Mitchell, and V. Pruneri, "True random numbers from amplified quantum vacuum," *Optics Express*, vol. 19, no. 21, p. 20665, oct 2011. [Online]. Available: **doi**: 10.1364/oe.19.020665

[18] W. Lei, Z. Xie, Y. Li, J. Fang, and W. Shen, "An 8.4 gbps real-time quantum random number generator based on quantum phase fluctuation," *Quantum Information Processing*, vol. 19, no. 11, nov 2020. [Online]. Available: **doi**: 10.1007/s11128-020-02896-y

[19] P. J. Bustard, D. Moffatt, R. Lausten, G. Wu, I. A. Walmsley, and B. J. Sussman, "Quantum random bit generation using stimulated raman scattering," *Optics Express*, vol. 19, no. 25, p. 25173, nov 2011. [Online]. Available: **doi**: 10.1364/oe.19.025173

[20] "ID Quantique Quantis QRNG chip," https://www.idquantique.com/random-number-generation/products/quantis-qrng-chip/, 2024, (Last accessed 2024/03/05).

[21] P. Keshavarzian, K. Ramu, D. Tang, C. Weill, F. Gramuglia, S. Tan, M. Tng, L. Lim, E. Quek, D. Mandich, M. Stipčević, and E. Charbon, "A 3.3-gb/s spad-based quantum random number generator," *IEEE Journal of Solid-State Circuits*, vol. PP, pp. 1–16, 09 2023. [Online]. Available: **doi**: 10.1109/JSSC.2023.3274692

[22] H. Xu, N. Massari, L. Gasparini, A. Meneghetti, and A. Tomasi, "A SPAD-based random number generator pixel based on the arrival time of photons," *Integration*, vol. 64, pp. 22–28, jan 2019. [Online]. Available: **doi**: 10.1016/j.vlsi.2018.05.009

[23] F. Regazzoni, E. Amri, S. Burri, D. Rusca, H. Zbinden, and E. Charbon, "A high speed integrated quantum random number generator with on-chip real-time randomness extraction," *arXiv preprint arXiv:2102.06238*, 2021.

[24] B. L. Márton, D. Istenes, and L. Bacsárdi, "Enhancing the operational efficiency of quantum random number generators," *Infocommunications Journal*, vol. 13, no. 2, pp. 10–18, 2021. [Online]. Available:**doi**: 10.36244/icj.2021.2.2

[25] X. Ma, F. Xu, H. Xu, X. Tan, B. Qi, and H.-K. Lo, "Postprocessing for quantum random-number generators: Entropy evaluation and randomness extraction," *Physical Review A*, vol. 87, no. 6, jun 2013. [Online]. Available: **doi**: 10.1103/physreva.87.062327

[26] R. Impagliazzo, L. A. Levin, and M. Luby, "Pseudo-random generation from one-way functions," in *Proceedings of the twenty-first annual ACM symposium on Theory of computing - STOC '89*. ACM Press, 1989. [Online]. Available: **doi**: 10.1145/73007.73009

[27] R. J. Glauber, "Coherent and incoherent states of the radiation field," *Physical Review,* vol. 131, no. 6, pp. 2766–2788, 9 1963. [Online]. Available: **doi**: 10.1103/PhysRev.131.2766

[28] M. C. Teich and B. E. A. Saleh, "Effects of random deletion and additive noise on bunched and antibunched photon-counting statistics," *Optics Letters*, vol. 7, no. 8, p. 365, aug 1982. [Online]. Available: **doi**: 10.1364/ol.7.000365

[29] B. Solymos and L. Bacsárdi, "Efficiency improvement of photon arrival time based quantum random number generator with hashing," in *IEEE 17th International Symposium on Applied Computational Intelligence and Informatics SACI 2023 : Proceedings*, 2023, pp. 53–58. [Online]. Available: DOI: 10.1109/SACI58269.2023.10158613

[30] A. Schranz and E. Udvary, "Mathematical analysis of a quantum random number generator based on the time difference between photon detections," *Optical Engineering*, vol. 59, no. 4, p. 044104, 2020. [Online]. Available: DOI: 10.1117/1.OE.59.4.044104

[31] Á. Schranz, "Optical solutions for quantum key distribution transmitters," Ph.D. dissertation, Budapest University of Technology and Economics, 2021. [Online]. Available: http://hdl.handle.net/10890/16991

[32] "NIST SP 800-22: Documentation and Software," https://csrc.nist.gov/projects/random-bit-generation/documentation-and-software, 2024, (Last accessed 2024/03/05).

[33] "dieharder by Robert G. Brown, Duke University Physics Department, Durham, NC 27708-0305 Copyright Robert G. Brown, 2019," https://webhome.phy.duke.edu/~rgb/General/dieharder.php, 2024, (Last accessed 2024/03/05).

[34] P. L'Ecuyer and R. Simard, "TestU01: A c library for empirical testing of random number generators," *ACM Transactions on Mathematical Software*, vol. 33, no. 4, pp. 1–40, aug 2007. [Online]. Available: DOI: 10.1145/1268776.1268777

[35] "ENT: A Pseudorandom Number Sequence Test Program," https://www.fourmilab.ch/random/, 2024, (Last accessed 2024/03/05).

[36] G. Marsaglia, ""the marsaglia random number cdrom including the diehard battery of tests of randomness". Florida State University. 1995. archived from the original on 2016-01-25." https://web.archive.org/web/20160125103112/http://stat.fsu.edu/pub/diehard/, (Last accessed 2024/03/05).

[37] X. Zhang, Y.-Q. Nie, H. Liang, and J. Zhang, "FPGA implementation of toeplitz hashing extractor for real time post-processing of raw random numbers," in *2016 IEEE-NPSS Real Time Conference (RT)*. IEEE, jun 2016. [Online]. Available: DOI: 10.1109/rtc.2016.7543094

[38] M. J. Ferreira, N. A. Silva, and N. J. Muga, "Efficient randomness extraction in quantum random number generators," in Anais do II Workshop de Comunicação e Computação Quântica (*WQuantum 2022*). Sociedade Brasileira de Computação, may 2022. [Online]. Available: DOI: 10.5753/wquantum.2022.223591

Balázs Solymos received his B.Sc. degree in 2018, followed by his his M.Sc. degree in early 2020 in Electrical Engineering from the Budapest University of Technology and Economics (BME). He is currently pursuing his PhD at the Department of Networked Systems and Services, BME. He is involved in a research project aiming to establish a quantum random generator service on campus. His current research interests are quantum communications, quantum internet, and quantum computing.

László Bacsárdi (M'07) received his MSc degree in 2006 in Computer Engineering from the Budapest University of Technology and Economics (BME) and his PhD in 2012. He is a member of the International Academy of Astronautics (IAA). Between 2009 and 2020, he worked at the University of Sopron, Hungary in various positions including Head of Institute of Informatics and Economics. Since 2020, he is associate professor at the Department of Networked Systems and Services, BME and head of Mobile Communications and Quantum Technologies Laboratory. His current research interests are quantum computing, quantum communications and ICT solutions developed for Industry 4.0. He is the past chair of the Telecommunications Chapter of the Hungarian Scientific Association for Infocommunications (HTE), Vice President of the Hungarian Astronautical Society (MANT). Furthermore, he is member of IEEE and HTE as well as alumni member of the UN established Space Generation Advisory Council (SGAC). In 2017, he won the IAF Young Space Leadership Award from the International Astronautical Federation.