Experimental Time-of-Arrival Quantum Random Number Generation with Dead Time Overestimation

Ágoston Schranz*†, Balázs Solymos*, Miklós Telek*†
*Department of Networked Systems and Services, Faculty of Electrical Engineering and Informatics
Budapest University of Technology and Economics, Műegyetem rkp. 3., H-1111 Budapest, Hungary
†HUNREN-BME Information Systems Research Group, Budapest, Hungary
Email: {aschranz, solymosb, telek}@hit.bme.hu

***Abstract*—A measurement-based validation of theoretical results on quantum random number generators (QRNGs) is considered in the paper. We have designed an experimental setup built around a single-photon detector to record random datasets of photon arrival time differences with various parameter settings.**

The collected datasets are used to generate random bit sequences with and without dead time overestimation. The statistical properties of the bit sequences are compared with analytical results on the one hand and assessed with four of the most popular statistical test suites on the other hand. The measurement results validate that the dead time overestimation algorithm helps to eliminate unwanted correlations from the generated bit sequences in practice.

***Index Terms*—random number generation, semiconductor lasers, time measurement, time series analysis**

I. Introduction

Truly random numbers are an essential building block for many applications, most notably cryptography. An unpredictable source of uniformly distributed symbols, usually bits, is the prerequisite for the safe adaptation of mathematically unbreakable symmetric-key protocols, such as the one-time pad [1]. Symmetric-key cryptography, enabled by quantum key distribution (QKD), is one of the candidates to provide secrecy even after quantum computers have become a serious threat to asymmetric protocols widely used as of today [2]. While present-day QKD implementations are often using dark fibers, possible integration of QKD into wavelength-division multiplexing systems is being researched actively [3]. Moreover, there have already been successful free-space optical demonstrations of satellite-to-ground and satellite-to-satellite QKD links [4].

Quantum random number generators (QRNGs) are based on the inherent uncertainty of quantum measurements and provide theoretically unpredictable sequences of bits [5], [6]. The randomness in optical phenomena is usually preferred to other alternatives, such as radioactivity, due to the availability of commercial-grade devices and the relative ease of management and maintenance. One group of optical generators creates bits from the uncertainty of times elapsed between photon detections, referred to as time-of-arrival QRNGs [7], [8], [9], [10].

In previous works [11], [12], we focused on the performance analysis of the time-of-arrival QRNG procedure first reported in Ref. [13]. This paper concludes those works with an experimental verification of the analysis results based on measurements.

We first analyzed the scheme and derived its relevant performance indices [11] and later expanded the analysis to properly account for correlations between successive timing samples (and bits) induced by a continuously running measurement clock [12]. Recently, we developed an algorithm that removes these correlations between the samples and deals with dead time in single-photon detection systems [14]. However, the latter algorithm has not yet been tested in QRNGs.

In this paper, we experimentally compare bit sequences: ones generated from raw datasets of time differences by the QRNG scheme and those generated from the datasets after our overestimation algorithm processed them. We assess the changes in the quality of randomness using statistical test suites and the consequential trade-off in terms of the bit generation rate. In Section II, we summarize the theoretical background, including the bit generation method, the formulae for the bit generation efficiency and the bit rate, and the algorithm referred to as *dead time overestimation*. Section III describes the experimental setup used in our measurements and its parameters. Section IV details and interprets the results we obtained, while Section V concludes the paper.

II. Theoretical background

Suppose the QRNG uses a light source emitting photons according to a Poisson point process (PPP). The PPP is characterized by its parameter λ, measured in 1/s, describing the mean number of photons emitted per second, thus being proportional to the mean optical power. The photons are detected by a single-photon detector (e.g. a photomultiplier tube (PMT)), and the detector's output pulses are time-tagged with a resolution τ. The QRNG in question counts the leading impulses of the τ intervals between successive time-of-arrival samples, D_i (c.f. Figure 1). It can be shown that these samples are independent if the reference clock signal is restarted at the same phase at each detection [12]; however, this restartability is not practically feasible at large photon detection rates and fine resolution.

A. Bit generation method

Random bits are generated by comparing successive pairs of samples, D_{2i-1} and D_{2i}. First, we generate the sequence

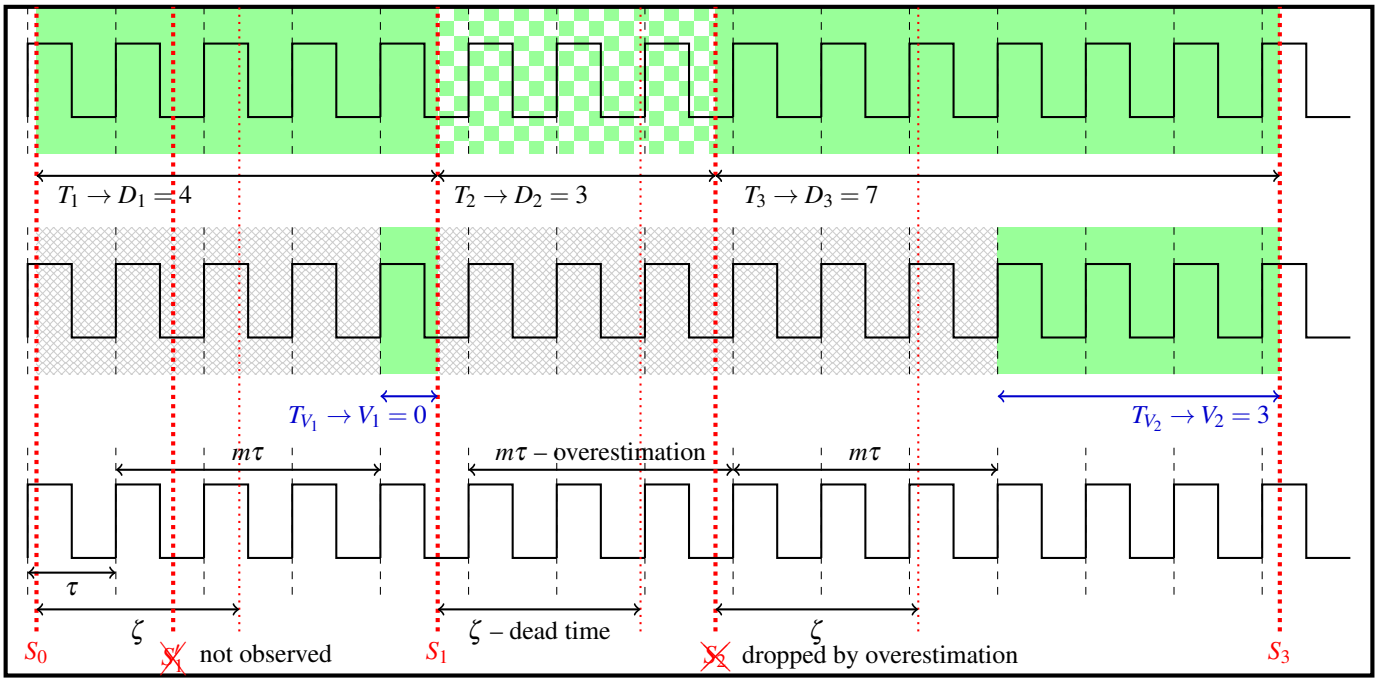

Figure 1. Comparison of time samples obtained with and without overestimation. Dead time is $\zeta = 2.3\,\tau$; the overestimation parameter is $m = 3$. Square waves represent the measurement clock; red dotted lines at S_0 to S_3 denote photon detections, while lighter red lines show the end of the respective dead times. S_1' denotes an undetected photon arrival within a dead time, while S_2 is detected but eventually dropped by overestimation. Top row: time samples D_i obtained by the original, raw method (green and checkered background). Middle row: time samples V_i obtained after the overestimation algorithm (green background; grey hatched background denotes time not used for sample generation). Bottom row: reference time diagram showing dead times and overestimation periods.

consisting of $R_i = \mathrm{sgn}\,(D_{2i-1} - D_{2i})$, where $\mathrm{sgn}(\cdot)$ is the sign function. Then, bits are assigned to the R_i values: if $R_i = -1$, the corresponding bit will obtain the value 0, if $R_i = 1$, we generate a bit 1 and we discard cases of $R_i = 0$. This method ensures that the bit sequence has a uniform distribution if the PPP is time-homogenous (with a time-independent λ), and the D_i values were measured using a restartable clock [13], [11].

B. Performance indices

We focus on two important performance indicators regarding the bit generation scheme: the bit generation efficiency η_g, defined as the mean number of bits generated per photon detection, and the bit generation rate R, the mean number of bits generated per unit time. Assuming an idealistic case, where we are able to use a restartable clock, and the detection system has no dead time—meaning it can distinguish between detections arbitrarily close to each other—, these metrics are a function of λ and τ [11]:

$$\eta_\mathrm{g} = \frac{1}{\mathrm{e}^{\lambda\tau}+1} \quad \text{and} \quad R = \frac{\lambda}{\mathrm{e}^{\lambda\tau}+1}. \tag{1}$$

For the particular scheme, η_g is always less than 0.5, as at least two detections are necessary for the creation of a single bit. Including dead time, the expressions become more complicated but retain much of their characteristics.

C. Dead time overestimation

In real-life applications, however, dead time is always present, and the measurement clock runs continuously in the background. Therefore, detections happen at random locations within a clock cycle, and this random *phase* introduces correlations between successive timing samples. Both of these effects cause the resulting bit sequences to deviate from the ideal, uniformly distributed case—leading to serious problems in cryptographic schemes.

The dead time overestimating algorithm, presented in Ref. [14], can eliminate both the distortion in the distribution of D_i samples and the correlations of consecutive samples.

The overestimation method is based on the observation that the unwanted features of the D_i samples are due to the small D_i values. If those small values are discarded, then the obtained time-corrected sample series is independent and geometrically distributed. Thus, the overestimation method discards the D_i samples, which are smaller than or equal to a threshold, m, referred to as the *overestimation parameter*, and assigns $V \leftarrow D - (m+1)$ new corrected output intervals for larger input D_i samples. Intuitively, the larger m is, the more samples are discarded, and the less the bit generation rate is. On the other hand, if m is large enough to hide the effect of the dead time, i.e. $m\tau$ is larger than the dead time, then the obtained samples are independent and geometrically distributed.

Note that the algorithm does not change the underlying PPP's parameter, only the virtual sample generation rate λ_v. See Fig. 1 for a comparison between how the original method and the overestimation-augmented scheme generate samples from the same realization of the photon arrival process.

III. Experimental setup

We implemented the bit generation scheme using the experimental setup shown in Fig. 2. The bit generation, overestimation, and binning processes run on a computer. Since we wanted to obtain raw and pre-processed bit sequences from the same original datasets, overestimation (and binning) happens offline on previously recorded sets of samples.

A. Physical devices

A semiconductor laser (Thorlabs LP520-SF15) operating at 519.9 nm is biased well above the lasing threshold by a driver board. Its light is attenuated using two cascaded variable optical attenuators (VOAs) (Thorlabs V450F) controlled by a microcontroller so that the generator operates at the desired power level and provides the Poisson-distributed stream of photons. The 1% output of a 99/1% power splitter (Thorlabs TW560R1F1) between the two VOAs provides 20 dB additional attenuation, while its other port is reserved for monitoring the power. The remaining photons are detected by a PMT (PicoQuant PMA-175 NANO). The PMT's output pulses are sent to a signal processing unit, including a PicoQuant TimeHarp 260 time-to-digital converter (TDC) time-tagger. The optical fibers (Thorlabs 460HP) connecting the individual devices are specially designed to ensure single-mode operation in the green part of the visible spectrum, having a core diameter of 2.5 µs. The time samples are then processed by a computer, which generates the bits, runs the binning and overestimating algorithms, and tests the quality of randomness.

The parameters of the PMT are very advantageous for this specific application. The device is more sensitive to wavelengths towards the blue end of the visible spectrum; thus, it is characterized by negligible afterpulsing probability and a dark count rate smaller than 50 cps (counts per second), even at room temperature. Assuming the practically available set of τ and ζ parameters, the input photon rates maximizing the bit generation rate are significantly larger than 50 cps for every dataset. Thus, the dark counts of thermal origin barely contribute to the process, and the uncertainty mainly comes from a well-characterized effect of truly quantum origin. The PMT has no explicit dead time other than the finite width of the voltage pulse denoting a detection—which has a FWHM value of 1.5 ns—, the timing accuracy (transit spread) is below 180 ps FWHM, while the quantum efficiency at 520 nm is $\sim 21\%$.

The time-tagger has a base resolution of $\tau_0 = 250$ ps. The total dead time of the system is determined by that of the TimeHarp 260 card. Based on the datasheet, its value is around $\zeta \approx 2\,\text{ns} = 8\tau$, larger than the dead time imposed by the photodetector. However, prior measurements and histograms of interarrival times suggest that the actual dead time might exceed 2 ns, and its value is not constant.

B. Measurement settings and parameter sets

If we assume the dead time to be zero, all performance indicators depend on one parameter, the product $\lambda\tau$ describing the mean number of photon detections per clock period[1]. To analyze the scheme comprehensively, we must thus cover a wide range of possible values of this product, from $\lambda\tau \ll 1$ to $\lambda\tau \simeq 1$.

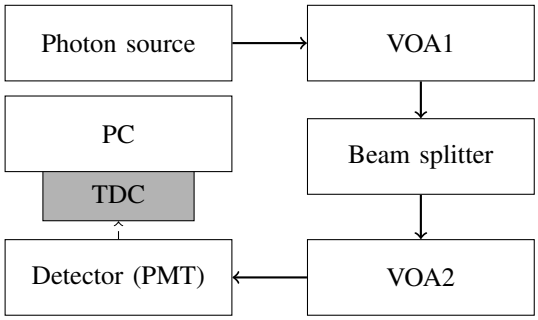

Figure 2. Experimental setup used to record datasets. VOA: variable optical attenuator; PMT: photomultiplier tube; TDC: time-to-digital converter card; PC: computer. (Beam splitter functions as an additional 20 dB attenuator.)

There are two limiting factors in increasing the product beyond $1.25 \cdot 10^{-3}$. The first is the detector's maximum allowed photon detection rate at 5 million cps. This is a hard limit, as exposing the PMT to stronger irradiation for extended periods could damage the device. The second limitation is the resolution of our TDC, $\tau_0 = 250$ ps. Using a resolution with higher τ in time-tagging mode T2—measuring arrival times without a periodic synchronization signal—is impossible. This problem, however, can be eliminated using a method called *binning*.

Binning is a post-processing method on the measured D_i values, allowing us to realize time resolutions that are integer multiples of the base resolution τ_0. Suppose that we obtained a sequence $\mathbb{D} = \{D_1, D_2, \dots\}$ measured using the base resolution and want to produce the sequence $\mathbb{D}' = \{D'_1, D'_2, \dots\}$—the one we would have obtained if our time-tagger's resolution was $K_\text{b} \cdot \tau_0$, $K_\text{b} \in \mathbb{Z}^+$. Algorithm 1 outputs the desired elements of $\mathbb{D}'$. We form D'_i by dividing a carry-corrected version of D_i by K_b, and taking the floor of the quotient. Algorithm 1 produces the same D'_i values as if the time-tagger had a resolution of $K_\text{b} \cdot \tau_0$.

Algorithm 1 Binning algorithm

Require: $\mathbb{D}$ ▷ Original timing samples
Require: K_b ▷ Integer
$c_1 := 0$ ▷ Carry
for $i = 1$ to length($\mathbb{D}$) **do**
  get D_i
  $D'_i = \lfloor (D_i + c_i)/K_\text{b} \rfloor$
  $c_{i+1} = D_i + c_i \pmod{K_\text{b}}$
end for

During our measurements, we recorded eight sequences of time differences, called *datasets*, indexed from A to H, all corresponding to different values of the product $\lambda\tau$. Note that

[1] Datasets pre-processed by the overestimation algorithm are essentially free of dead time. Even in the case of raw datasets, the $\lambda\tau$ product determines much of the statistical properties, although not completely.

datasets $\mathbb{D}_E$ to $\mathbb{D}_H$ are recounted versions of dataset $\mathbb{D}_D$ using the binning algorithm. Table I lists the relevant parameters for the individual sequences. For each dataset, we decided on the set of overestimation parameters we wanted to examine, denoted as $\mathbb{M}$, to contain values that are too low, overly safe and in between. For example, $m = 10$ corresponds to an overestimation of $m\tau = 2.5\,\text{ns}$ for the datasets measured using the $\tau = 250\,\text{ps}$ resolution, close to the datasheet value of $\zeta = 2\,\text{ns}$. $\mathbb{M}$ differs between binned and non-binned sequences, as binning rescales the necessary overestimation parameters (as would changing the resolution with a fixed dead time).

Table I
DATASETS AND CORRESPONDING PARAMETERS

$\mathbb{D}_{ID}$	λ [cps]	τ [ns]	$\lambda\tau$	K_b	$\mathbb{M}_{ID}$
$\mathbb{D}_A$	$5.080 \cdot 10^5$	0.25	0.000 127	1	$\{10, 50, 100, 500\}$
$\mathbb{D}_B$	$1.238 \cdot 10^6$	0.25	0.000 310	1	$\{10, 100, 500\}$
$\mathbb{D}_C$	$2.199 \cdot 10^6$	0.25	0.000 550	1	$\{10, 100, 500\}$
$\mathbb{D}_D$	$4.111 \cdot 10^6$	0.25	0.001 028	1	$\{10, 100, 500, 1000\}$
$\mathbb{D}_E$	$4.111 \cdot 10^6$	1.25	0.005 139	5	$\{2, 10, 100\}$
$\mathbb{D}_F$	$4.111 \cdot 10^6$	12.5	0.051 387	50	$\{2, 4, 20\}$
$\mathbb{D}_G$	$4.111 \cdot 10^6$	125	0.513 865	500	$\{1, 4\}$
$\mathbb{D}_H$	$4.111 \cdot 10^6$	300	1.233 277	1200	$\{1, 2, 3, 4, 5, 10\}$

To differentiate between bits generated from a certain dataset using overestimation with parameter m, we introduce the notations A_m to H_m for the respective bit sequences (A_0 to H_0 denoting sequences generated without overestimation).

IV. RESULTS AND DISCUSSION

Based on the bit sequences generated from the measured datasets, we validate our prior analysis results for the raw datasets and for the overestimation method.

A. Distribution of bit tuples

First, we evaluate the distribution of the bits. We have shown in Ref. [12] that they are always unbiased, independently of λ, τ and ζ. However, the bits are not uncorrelated, resulting in bit tuple distributions (describing the probabilities of successive bits) that deviate from the ideal uniform case. The magnitude of this deviation strongly depends on the physical parameters.

We have proved in Theorem 2 of Ref. [12] that the so-called *inverse-reverse relation* applies to any bit sequence $b_1, b_2, \ldots b_N$ generated by the scheme, even in the presence of dead time. This means that

$$\begin{aligned} &\Pr\left(B_1 = b_1, B_2 = b_2, \ldots B_{N-1} = b_{N-1}, B_N = b_N\right) = \\ &\Pr\left(B_1 = \overline{b_N}, B_2 = \overline{b_{N-1}}, \ldots B_{N-1} = \overline{b_2}, B_N = \overline{b_1}\right), \end{aligned} \tag{2}$$

where $\overline{b_j}$ denotes the inverse of b_j (if $\overline{b_j} = 0$, then $b_j = 1$ and vice versa), and B_j is the random variable describing the jth bit. E.g., the probability of bit triplets 011 and 001 is equal.

Figure 3a) indicates that the deviation from uniformity decreases with decreasing $\lambda\tau$ for bit sequences without overestimation. Figure 3b) shows the bit triplet probabilities for a selected few bit sequences: all of them obtained from dataset $\mathbb{D}_D$, and the non-overestimated sequence from $\mathbb{D}_A$. The pair and triplet distribution of sequence D_0 is noticeably non-uniform; as are that of D_{10}, hinting that the choice of $m = 10$ is insufficient to get rid of the dead time's distorting effects. However, as m increases to 100, 500, and 1000, the probabilities tend to the uniform value of 0.125. Interestingly enough, the distributions of A_0 are closer to uniform than even D_{1000}. This aligns with our previous analysis: smaller values of $\lambda\tau$ yield better-behaved bit sequences than those with a higher product within the interval we are checking. Another noticeable feature of the curves is the indication of a positive lag-1 correlation between bits (e.g. pairs, triplets of the same bit are the most probable), also reported in Ref. [12]. These traits are generally true for the bit sequences of every dataset. The analysis of pair and triplet probabilities provides a way of finding the smallest appropriate overestimation parameter for a given value of τ and ζ.

B. Bit generation rates with overestimation

For each dataset, as m increases, the number of bits generated from the samples decreases as more and more detections fall within the insensitive periods. To quantify this effect, we define the bit retention efficiency η_b as

$$\eta_b = \frac{\text{bits generated from } \mathbb{D} \text{ using overestimation}}{\text{bits generated from } \mathbb{D} \text{ by the raw method}}, \tag{3}$$

to compare the length of bit sequences generated from the same dataset. The bit retention efficiency is a figure of merit that is readily available, and it helps to quantify the losses due to the overestimation algorithm.

Table II summarizes the bit generation rates and efficiencies for unprocessed datasets—without overestimation—and the bit retention efficiencies for the chosen values of m. (The generation rates for overestimated datasets could be calculated as the product of the unprocessed rate and the corresponding η_b. The η_g values, however, are defined.)

Table II
BIT RETENTION EFFICIENCIES, BIT GENERATION RATES AND BIT GENERATION EFFICIENCIES OF DIFFERENT BIT SEQUENCES (ELEMENTS OF CORRESPONDING $\mathbb{M}_{ID}$ IN INCREASING ORDER). NOT EVERY ELEMENT IN A COLUMN CORRESPONDS TO THE SAME m.

$\mathbb{D}_{ID}$	raw data		**Overestimation parameter**			
			$\{\mathbb{M}_{ID}\}_1$	$\{\mathbb{M}_{ID}\}_2$	$\{\mathbb{M}_{ID}\}_3$	$\{\mathbb{M}_{ID}\}_4$
	R [bps]	η_g	η_b	η_b	η_b	η_b
$\mathbb{D}_A$	$2.539 \cdot 10^5$	0.4999	0.9999	0.9959	0.9904	0.9453
$\mathbb{D}_B$	$6.189 \cdot 10^5$	0.4999	0.9999	0.9764	0.8655	–
$\mathbb{D}_C$	$1.099 \cdot 10^6$	0.4999	0.9997	0.9582	0.7718	–
$\mathbb{D}_D$	$2.053 \cdot 10^6$	0.4997	0.9995	0.9221	0.6109	0.3558
$\mathbb{D}_E$	$2.050 \cdot 10^6$	0.4987	0.9986	0.9643	0.6095	–
$\mathbb{D}_F$	$2.005 \cdot 10^6$	0.4876	0.8996	0.8150	0.3460	–
$\mathbb{D}_G$	$1.668 \cdot 10^6$	0.4057	0.4296	0.0868	–	–
$\mathbb{D}_H$	$1.441 \cdot 10^6$	0.3506	0.1010	0.0282	0.0078	0.0022

Looking at the generation rates of sequences A_0 to D_0, it can be deduced that for the same resolution, an increasing photon arrival rate increases R as well. This is to be expected until reaching a local or global maximum of the function [11], which, for the given τ, would only appear at λ values way past the possible limit of 5 million cps. From D_0 to H_0, however, λ is unchanged, but τ increases, yielding smaller and smaller

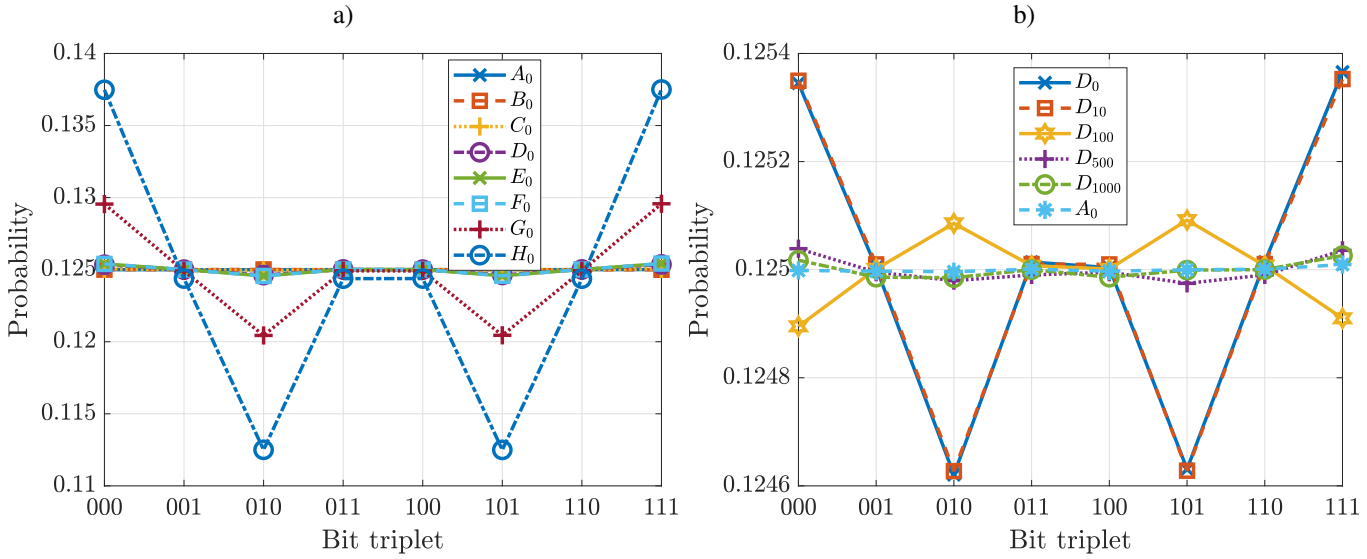

Figure 3. Bit triplet probabilities. a) For sequences with different $\lambda\tau$ and no overestimation. Decreasing the product brings the distributions closer to uniform. b) For sequences generated from dataset $\mathbb{D}_\mathrm{D}$ with different overestimation parameters m. Increasing m brings the distributions closer to uniform, indicating an improvement in the quality of randomness. Probabilities from bit sequence A_0, obtained from dataset $\mathbb{D}_\mathrm{A}$, show that small $\lambda\tau$ yields almost-uniform distributions even without overestimation. In both plots, the probabilities hint at the presence of the inverse-reverse relation. Note the largely different y-axis scaling on subplots a) and b).

generation rates. This result is also in line with the results of the analysis we reported previously.

The losses also behave according to the analytical results. For small λ, when arrivals are relatively rare, even the highest examined overestimation rates result in mild losses. On the other hand, for $\mathbb{D}_\mathrm{H}$, the choice of $m = 1$ reduces the bit sequence's length by almost 90% since chains of detections restarting the overestimation interval are frequent. Note that, with respect to the overestimation time $m\tau$, this $m = 1$ (with $\tau = 300\,\mathrm{ns}$) corresponds to $m = 1200$ (with $\tau = 0.25\,\mathrm{ns}$) for non-recounted cases. That is, photon arrivals with less than $m\tau = 300\,\mathrm{ns}$ interarrival times are discarded when the mean photon interarrival time is 243 ns (the reciprocal of $\lambda = 4.111 \cdot 10^6\,\mathrm{cps}$).

Figure 4 shows the bit retention efficiency for several datasets as a function of the adjusted overestimation parameter (AOP) $m \cdot K_\mathrm{b}$; the AOP allows for a fair comparison between sequences, as the binning algorithm rescales the dead time by $1/K_\mathrm{b}$ in terms of the resolution. The figure contains cases not included in the respective $\mathbb{M}_\mathrm{ID}$ sets.

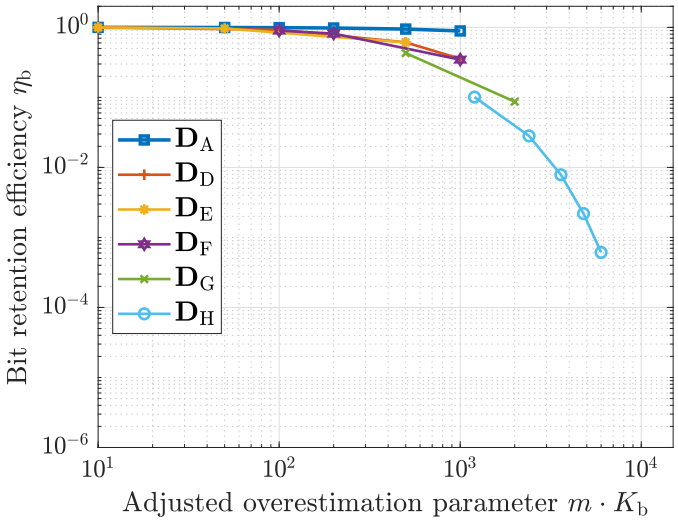

Figure 4. Bit retention efficiency as a function of AOP for selected datasets.

C. Statistical testing

We use statistical test suites to assess and compare the quality of the bit sequences. The four suites used are the following: NIST Statistical Test Suite [15] (default parameters, 1024 teststreams), Dieharder [16] (default parameters), TestU01 [17] (Alphabit and Rabbit batteries) and ENT [18] (in both bit and byte modes).

Each of these suites contains multitudes of different hypotheses tests, looking for different non-random patterns in the tested data. The sequences are evaluated against an ideal uniform sequence in various test statistics, and the resulting p-values are compared to the default significance levels of the particular suites. The results are mostly binary PASS or FAILED (Dieharder has an intermediate WEAK category). Due to our limited data storage capacity per interval and bit sequence, we could not fully satisfy the data requirements of some test cases. Thus, in cases with shorter output bit sequences, the NIST STS and TestU01 suites may not complete successfully, while for some Dieharder tests, the data file gets rewound several times, yielding skewed results. Nonetheless, even after considering this limitation, we observed clear differences between the results corresponding to sequences generated from unprocessed and overestimated cases.

Overestimated sequences are expected to pass the statistical tests if the chosen m overestimation parameter is large enough

to majorate the dead time. We found that for each of our $\mathbb{D}$ measurement cases, sequences with the largest corresponding m parameters passed the applicable statistical tests[2].

For the unprocessed cases, we found that cases with lower $\lambda\tau$ values are still passing the suites successfully, while from $\mathbb{D}_{\mathrm{C}}$ on, all of the sequences corresponding to unprocessed data fail. The primary failing trials for each of the used suites were tests investigating "runs of bits". Calculated lag-1 correlation coefficients also increased with increasing $\lambda\tau$, yielding values of 0.00126, 0.00127, 0.0014, 0.01, 0.04 for cases from D_0 to H_0. These observations align with the theorized effects of the phenomenon investigated in [12].

V. Conclusion

We experimentally compared bit sequences generated from raw datasets and from datasets processed by our previously presented overestimation method [14]. First, we experimentally showed the validity of our previous analytical results [12], by investigating multiple bit sequences generated from unprocessed time difference datasets. We compared these with results from bit sequences generated from overestimated datasets. We found that the unwanted correlation artefacts vanished when using sufficiently large m overestimation parameters, thus experimentally validating the overestimation method's correctness. We collected data points corresponding to multiple m values for each physical measurement case and showed the expected decrease in output efficiency with increasing m parameters. Finally, we also statistically assessed the output bit sequences with popular statistical test suites to verify the correct operation.

In addition to the experimental verification of previous analytical work [12], [14], our presented results also nicely highlight the diminishing effect of the discussed non-idealities with decreasing $\lambda\tau$ product.

Acknowledgment

Miklós Telek was supported by the OTKA K-138208 project of the Hungarian Scientific Research Fund. Balázs Solymos was supported by the Ministry of Culture and Innovation and the National Research, Development and Innovation Office within the Quantum Information National Laboratory of Hungary (Grant No. 2022-2.1.1-NL-2022-00004). Ágoston Schranz was supported by the OTKA K-142845 project of the Hungarian Scientific Research Fund and also received funding from the European Union under grant agreement No. 101081247 (QCIHungary project), which has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund.

References

[1] C. E. Shannon, "Communication theory of secrecy systems," *The Bell System Technical Journal*, vol. 28, no. 4, pp. 656–715, 1949. [Online]. Available: https://doi.org/10.1002/j.1538-7305.1949.tb00928.x

[2] V. Zapatero, T. van Leent, R. Arnon-Friedman, W.-Z. Liu, Q. Zhang, H. Weinfurter, and M. Curty, "Advances in device-independent quantum key distribution," *Npj Quantum Inf.*, vol. 9, no. 1, Feb. 2023. [Online]. Available: https://doi.org/10.1038/s41534-023-00684-x

[3] E. Udvary, "Integration of qkd channels to classical high-speed optical communication networks," *Infocommunications journal*, vol. 15, no. 4, p. 2–9, 2023. [Online]. Available: http://dx.doi.org/10.36244/ICJ.2023.4.1

[4] C.-Y. Lu, Y. Cao, C.-Z. Peng, and J.-W. Pan, "Micius quantum experiments in space," *Rev. Mod. Phys.*, vol. 94, p. 035001, Jul 2022. [Online]. Available: https://link.aps.org/doi/10.1103/RevModPhys.94.035001

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

[6] V. Mannalatha, S. Mishra, and A. Pathak, "A comprehensive review of quantum random number generators: concepts, classification and the origin of randomness," *Quantum Inf. Process.*, vol. 22, no. 12, Dec. 2023. [Online]. Available: https://doi.org/10.1007/s11128-023-04175-y

[7] Y.-Q. Nie, H.-F. Zhang, Z. Zhang, J. Wang, X. Ma, J. Zhang, and J.-W. Pan, "Practical and fast quantum random number generation based on photon arrival time relative to external reference," *Appl. Phys. Lett.*, vol. 104, no. 5, p. 051110, Feb. 2014. [Online]. Available: https://doi.org/10.1063/1.4863224

[8] G. Shaw, S. Sivaram, and A. Prabhakar, "Quantum random number generator with one and two entropy sources," in *2019 National Conference on Communications (NCC)*, 2019, pp. 1–4. [Online]. Available: https://doi.org/10.1109/NCC.2019.8732222

[9] A. Tomasi, A. Meneghetti, N. Massari, L. Gasparini, D. Rucatti, and H. Xu, "Model, validation, and characterization of a robust quantum random number generator based on photon arrival time comparison," *Journal of Lightwave Technology*, vol. 36, no. 18, pp. 3843–3854, 2018. [Online]. Available: https://doi.org/10.1109/JLT.2018.2829210

[10] A. Stanco, D. G. Marangon, G. Vallone, S. Burri, E. Charbon, and P. Villoresi, "Certification of the efficient random number generation technique based on single-photon detector arrays and time-to-digital converters," *IET Quantum Communication*, vol. 2, no. 3, pp. 74–79, 2021. [Online]. Available: https://doi.org/10.1049/qtc2.12018

[11] Á. 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: https://doi.org/10.1117/1.OE.59.4.044104

[12] Á. Schranz, B. Solymos, and M. Telek, "Stochastic performance analysis of a time-of-arrival quantum random number generator," *IET Quantum Communication*, pp. 1–17, dec 2023. [Online]. Available: https://doi.org/10.1049/qtc2.12080

[13] 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: https://doi.org/10.1063/1.2720728

[14] B. Solymos, A. Schranz, and M. Telek, "Correlation avoidance in single-photon detecting quantum random number generators by dead time overestimation," 2024, Research Square, preprint. [Online]. Available: https://doi.org/10.21203/rs.3.rs-3914156/v1

[15] A. L. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, and S. Vo, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," National Institute of Standards & Technology, Gaithersburg, MD, United States, Tech. Rep., 2010, spec. Pub. 800-22, Rev. 1a. [Online]. Available: https://doi.org/10.6028/NIST.SP.800-22r1a

[16] R. G. Brown, D. Eddelbuettel, and D. Bauer, "Dieharder: A random number test suite," Duke University Physics Department, Durham, NC 27708-0305, https://webhome.phy.duke.edu/~rgb/General/dieharder.php, 2024, (Last accessed 2024/04/22).

[17] 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: https://doi.org/10.1145%2F1268776.1268777

[18] J. Walker, "ENT: A pseudorandom number sequence test program," https://www.fourmilab.ch/random/, 2008, (Last accessed 2024/04/22).

[2] Beside file rewinds in some Dieharder trials, $\mathbb{D}_{\mathrm{G}}$ had insufficient data for running the NIST STS, while $\mathbb{D}_{\mathrm{H}}$ had insufficient data for the NIST STS and TestU01 assessments.