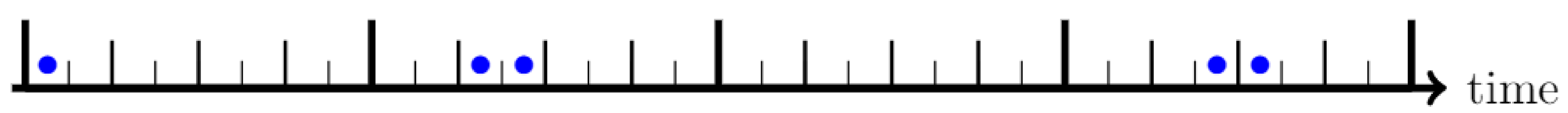
Key Rates with Non Ideal Photon Detectors in Time-Entanglement QKD

Christopher Cheng Dunbar Birnie Emina Soljanin

This project is supported in part by NSF Award No. 2007203

Time-Entanglement Based QKD

Time-entanglement is a promising way to address the photon-starved conditions that limit key rate in polarization-entanglement [2]. With arbitrary precision, we could extract arbitrarily many bits from a single photon arrival time. In practice, precision is limited, so to extract bits from timing information, what we do is divide time into discrete **bins**, which are then grouped into **frames**. Frames can be thought of as encoded binary words, where each bin corresponds to a `1` when occupied by at least one photon and `0` otherwise.



Four "frames", each with $n = 8$ time units.

In the figure above, a blue dot indicates an occupied bin, so the first frame will be interpreted as `10000000`, the second frame is `00110000`, and so on. We can then decode each frame individually (i.e. with binning schemes such as PPM or with compression algorithms) and concatenate the results to produce the final key bits.

Calculating Information Rate

We will focus on two variables which affect how key generation rate is calculated.

The Rate at Which Entangled Photon Pairs are Generated

We model the duration between consecutive entangled photon pair production times as exponential with the rate λ_p. From λ_p, we can find the probability that a bin is occupied, $p = 1 - \exp(-\tau\lambda_p)$, where τ is the bin width in time. Therefore, the maximum number of bits per bin is $R_p = h(p)$, where $h(p) = -p\log_2 p - (1-p)\log_2(1-p)$ is the binary entropy function.

Detector Resolution / Bin Width

Increasing detector resolution means more bits available for us to decode per unit time, which compensates for the resulting decrease in p. The entropy extracted per time is

$$R_t = R_p/\tau = h(p)/\tau = h(1 - \exp(-\tau\lambda_p))/\tau \text{ bits/time}$$

For example, if we halve the bin width and use $\tau' = \tau/2$ instead of τ, we will extract a maximum of $R_t' = h(1-\exp(-\tau'\lambda_p))/\tau' = 2h(1-\exp(-\tau\lambda_p/2))/\tau$ bits per time, which is **strictly greater** than R_t except at $p = 0$. In theory, if we can arbitrarily decrease bin width, we can also arbitrarily increase R_t. Note that while improvements in λ_p also benefit key rates from polarization entanglement, improvements in detector resolution mostly only benefit key rates from time entanglement.

Does Time-Entanglement Live up to its Promise?

The two variables – entangled pair production rate λ_p and bin width τ – as we have described them so far, would, in theory, allow designers to arbitrarily increase the key rate. In practice, decreasing τ too far may overwhelm the system by **jitter errors**, and increasing p too far, reduces the system efficacy because of the detector **recovery-time**. Additionally, notice that by $p = 0.5$, we are back to extracting **at most one bit per photon arrival**. This is predicted by the formula for maximum photon utilization,

$$h(1 - \exp(-\tau\lambda_p))/(\tau\lambda_p) \text{ bits/photon arrival,}$$

which implies that in ideal conditions, polarization entanglement does better when $\tau\lambda_p > h(1 - \exp(-\tau\lambda_p))$ since $\tau\lambda_p$ is the expected number of bits extracted per bin when polarization entanglement is used. With the addition of the effects mentioned above, this flipping point may shift. The advantages of time-entanglement are further reduced by sub-optimal bit extraction methods and non-ideal detectors.

Detector Recovery-Time

Detector recovery-time is a time interval following a photon detection during which the detector is **unresponsive** to any subsequent photon arrivals.

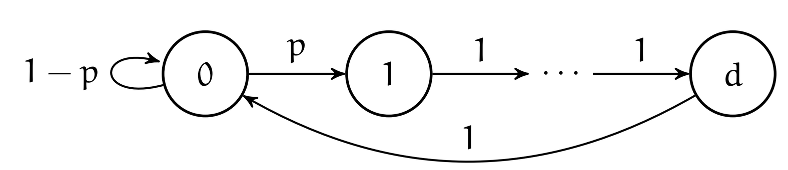
This recovery-time is not restricted by time resolution and may be much greater than the finest distinguishable time unit. This means that photon arrivals can **no longer be modeled as i.i.d.**; rather than as an exponential distribution, photon interarrival times are better modeled as a shifted exponential distribution, where the shift corresponds with the recovery time. This introduction of memory leads to reductions in R_p and R_t.

Markov Chain Representation

Because of memory between the frames, we model the system by a Markov Chain (MC). The raw key rate is equal to the entropy rate of the MC: $\sum_{ij} \boldsymbol{\mu}_i \mathbf{P}_{ij} \log_2 \mathbf{P}_{ij}$ bits, where $\boldsymbol{\mu}$ is the stationary probability vector and $\mathbf{P}$ is the transition matrix.

An Upper Bound on the Key Rate

When we only take into account the detector downtime but not the detector structure, the entropy rate of the corresponding MC represents an upper bound to the key rate. The chain and the bound are shown below.

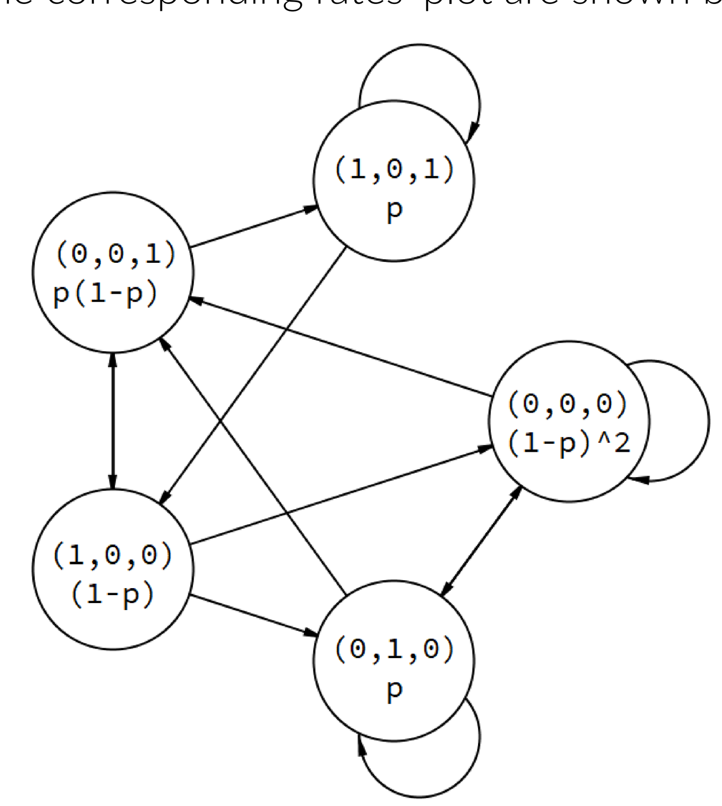


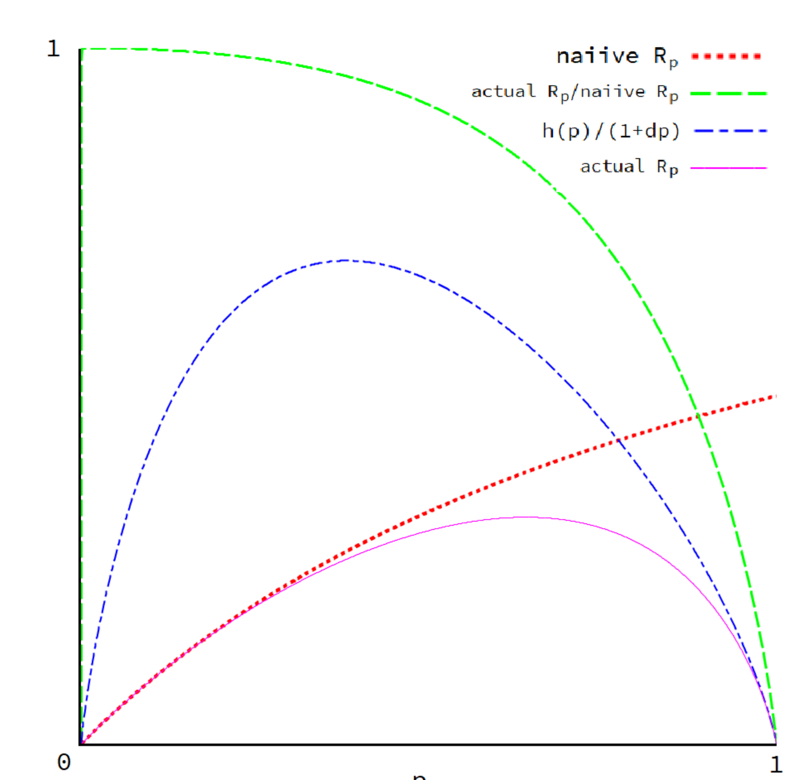
$$R_p \leq \frac{h(p)}{1+dp} \quad \text{bits/bin}$$

Here, d is the recovery-time in bins (so if $k = 1\text{ps}$, and recovery-time is 2ps, then $d = 2$).

In the case of Non-Optimal Schemes

For typical n's (1024, 4096) [2], calculating R_p using an MC state for each of the 2^n frames is impractical. To address this problem, we have developed a way to combine states so that the total number of states increases **polynomially** with n, with negligible loss of accuracy in R_p and R_t. The idea is to combine "similar" states i and j, where states i and j are similar if the distance between $\mathbf{P}_i$ and $\mathbf{P}_j$ *and* between $\mathbf{P}_i^\top$ and $\mathbf{P}_j^\top$ is small. An example MC generated using our method and the corresponding rates' plot are shown below:





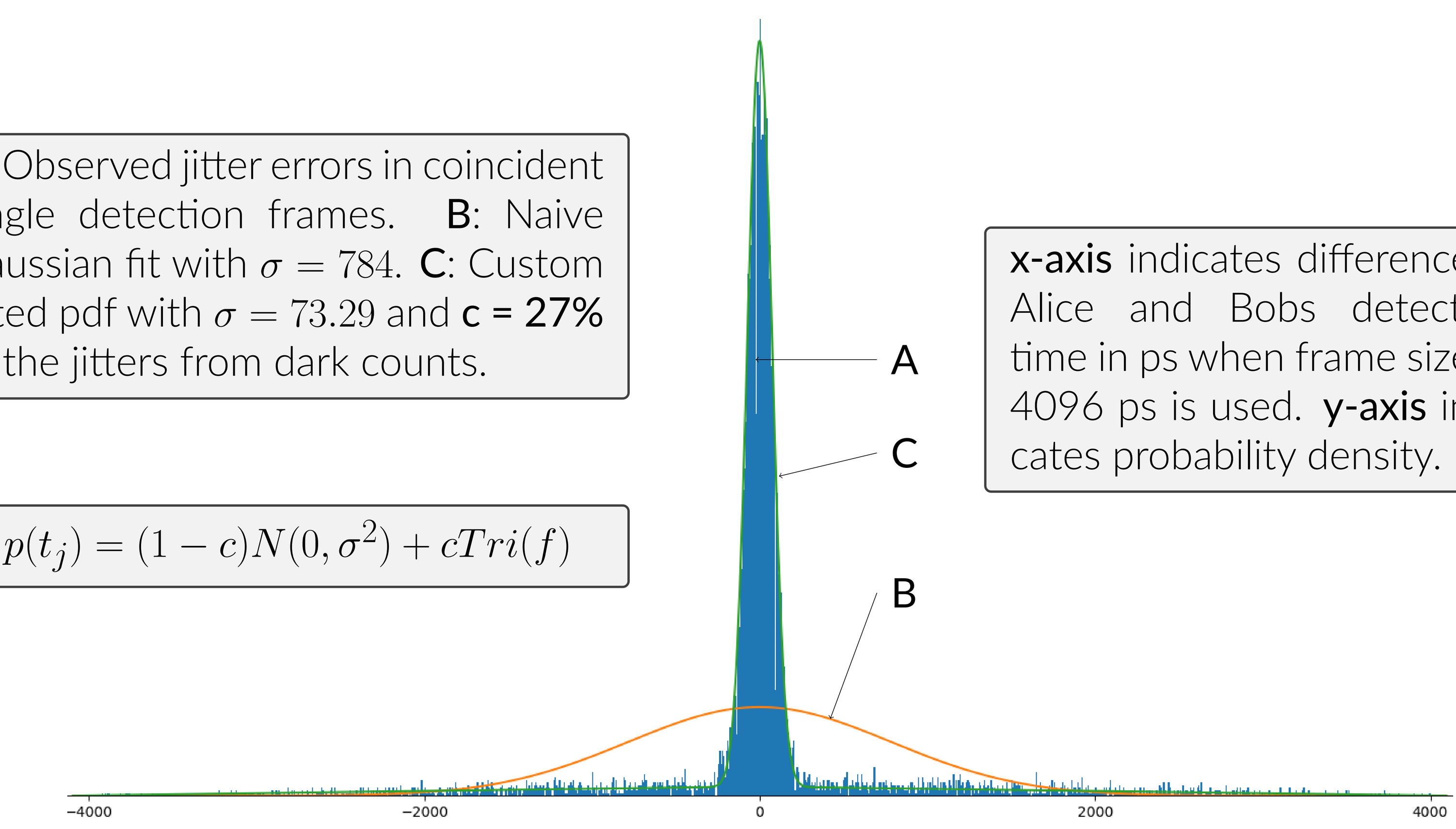
Each state contains an identifier (d_i,n_1,d_o), where d_i and d_o are recovery-time into and out of the frame, respectively, and n_1 is the number of occupied bins inside the frame, minus one if $d_o > 0$. The expression under each identifier is the transition probability of all transitions into that state.

Jitter Error and Associated Rate Cost

Observed jitter error refers to the difference in detection times between the two stations. Two factors contribute to observed jitter errors: detector imprecision and uncorrelated coincident dark counts. In PPM Alice and Bob consider frames where they both have only one occupied bin. In some cases they both declare a frame valid despite only seeing dark counts. Observed jitter errors are shown below using only PPM-valid frames which were extracted from experimental data. We thank Murat Sarihan for providing the raw arrival data [1].

A: Observed jitter errors in coincident single detection frames. B: Naive Gaussian fit with $\sigma = 784$. C: Custom fitted pdf with $\sigma = 73.29$ and c = 27% of the jitters from dark counts.

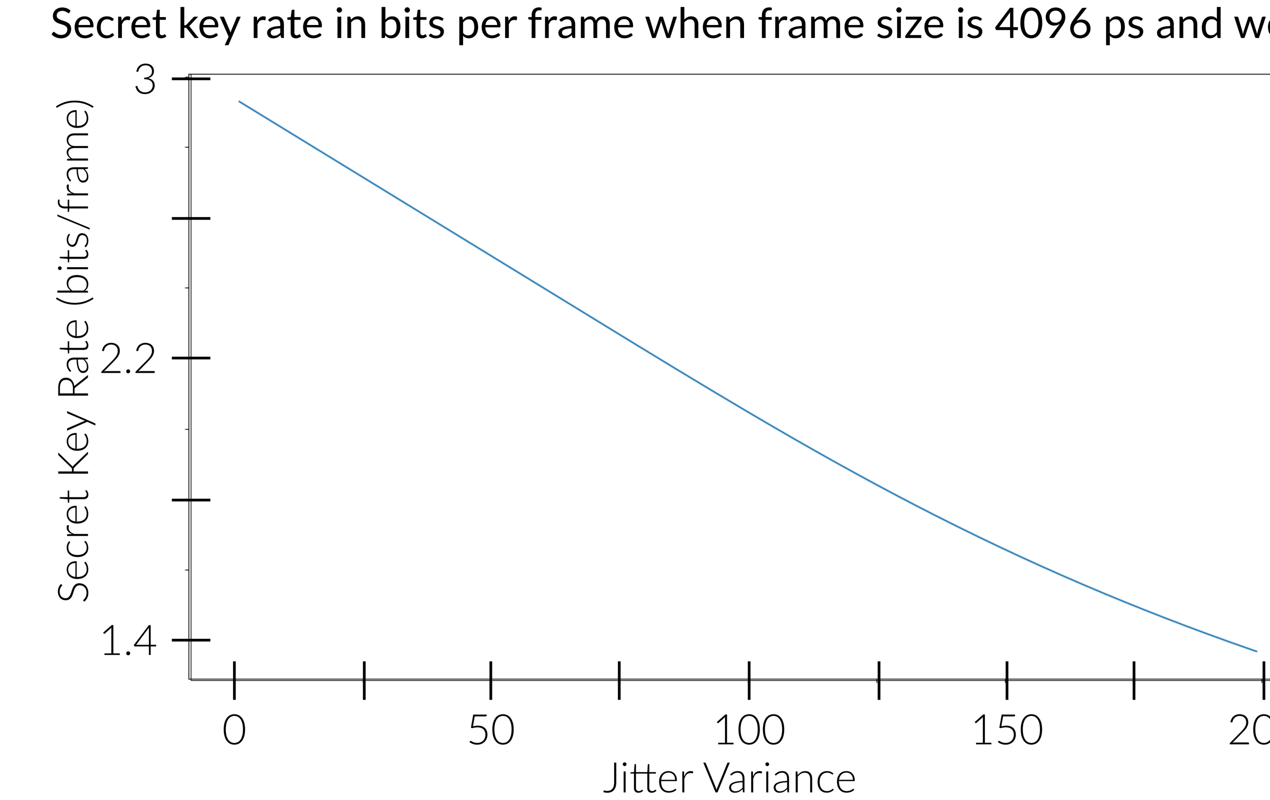
C: $p(t_j) = (1-c)N(0, \sigma^2) + cTri(f)$



x-axis indicates difference in Alice and Bobs detection time in ps when frame size of 4096 ps is used. **y-axis** indicates probability density.

Jitter errors caused by the detector imperfections follow a **Gaussian distribution**. These errors are different from errors introduced through dark counts, which produce a **triangular distribution** (the convolution of two uniform distributions over the frame width). Around 62% of jitters are expected to be less than 100 ps in magnitude for this frame configuration. The plot below shows the direct relation between detector jitter variance and key rate loss (generated using the error channel characterization above).

Secret key rate in bits per frame when frame size is 4096 ps and we use 16 bins per frame



Reed Solomon Codes require t symbols to resolve t errors at known locations. Using this, we calculate the number of parity symbols needed to correct the expected errors and subtract these bits from the raw key. Using larger (or fewer) bins can increase the jitter resilience.

References

[1] Murat Can Sarihan. Private communications, 2021.

[2] Tian Zhong. Photon-efficient quantum key distribution using time–energy entanglement with high-dimensional encoding. `https://iopscience.iop.org/article/10.1088/1367-2630/17/2/022002/pdf`.