Mathematical Modeling of Quantum Repeaters Chains

Lorenzo La Corte

November 15, 2024

Table of Contents

Chapter 1

Mathematical Model for Waiting Time and Fidelity

The objective of a protocol for a quantum repeater chain is to establish an entangled link between the two endpoints. The process is denoted as *entanglement distribution* and the output is a pair of entangled qubits, one at each endpoint, referred to as the *end-to-end link*.

In this chapter, we present a **mathematical model** to quantify the waiting time needed to achieve the end-to-end link, and its fidelity to the ideal (maximally entangled) state—a measure of its quality.

The model described in this section is entirely based on the work of Li et al. [LCE21], which builds upon the work of Brand et al. [BCE20].

We merge and simplify the discussion by not considering cutoffs, introduced by Li *et al.*, and by including examples and visualizations to explain concepts that are underlined in the original works (see Appendix B). The objective of this chapter is to provide a clear and comprehensive explanation of the mathematical model behind the algorithm used to simulate a single repeater chain protocol. It takes in input the hardware parameters of the chain, and returns the probability mass function of the waiting time needed to produce the end-to-end link, and the associated distribution of values for its quality (see, for example, Figure B.1).

1.1 Introduction

We derive expressions for the waiting time and fidelity of the first generated end-to-end link in the repeater chain protocol.

The random variable T_n **represents the waiting time** necessary to achieve a link covering $(2^n + 1)$ nodes. It is derived recursively in the following sections.

The random variable W_n **represents the quality of the entanglement** produced, quantified by the Werner parameter of the output link. It is related to its fidelity F_n by

$$F_n = \frac{1 + 3W_n}{4} \tag{1.1}$$

and thus represents **the quality of the entanglement**.

Class of Repeater Protocols. The operations studied in this class of repeater protocols are

- entanglement generation over a single hop (GEN),
- entanglement swapping ($SWAP$) of two short-distance links into a single long-distance one,
- entanglement distillation ($DIST$) of two low-quality links into a single high-quality one.

These processes are denoted in the following as **PROTOCOL-UNITs**. They all take a duration that is a multiple of L_0/c, the time to send information over a single segment. For this reason we denote the waiting time in **discrete units** of L_0/c. This will also be the unit of time for the discrete simulation algorithm based on the model [LCE21].

These are all *heralded probabilistic* processes, i.e. they are successful with a certain probability, and the outcome (success or fail) will be classically communicated to the parties involved. If the process fails, all the links are lost and the protocol is restarted. Thus, all the PROTOCOL-UNITs involved in the protocol distributing entanglement have to succeed for the end-to-end link to be established. The waiting times to achieve each step are stacked on top of each other, leading to the total waiting time needed, which here we try to model as a random variable (see Appendix A.2).

Cutoffs ($CUT\text{-}OFF$) are also considered in [LCE21], but are not under consideration in this work. They represent conditions imposed on the quality of the entanglement. If not met, the entanglement is discarded and the protocol is restarted.

1.2 Heraldeld Entanglement Generation

The first step in the repeater chain protocol is the generation of elementary entanglement between two nodes. In the following, we derive the waiting time and Werner parameter for the generation protocol.

1.2.1 Waiting Time for Elementary Entanglement

With T_0 we denote the **waiting time for the generation of elementary entanglement**.

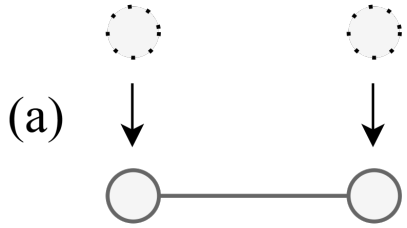

Figure 1.1: Elementary entanglement is (a) generated probabilistically.

Since we model the generation of single-hop entanglement by attempts which **succeed with a fixed probability** p_{gen}, the waiting time T_0 is a discrete random variable (in units of L_0/c) which follows a geometric distribution (see Appendix A.3) with probability distribution given by

$$\Pr(T_0 = t) = p_{\text{gen}}(1 - p_{\text{gen}})^{t-1} \quad \text{for} \quad t \in \{1, 2, 3, \ldots\} \tag{1.2}$$

where p_{gen} is a hardware parameter, i.e. depends on the quality of the physical devices used in the experiments. All the hardware parameters are an input for the model and the simulation algorithm.

1.2.2 Werner Parameter for Elementary Entanglement

The output state of the entanglement generation protocol is a Werner state with Werner parameter w_0. This value represents the quality of the freshly-generated input pair, and depends on the quality of the underlying hardware. No matter how much time is need to generate the entanglement, the quality of the entanglement is fixed at w_0. Thus, the distribution of the Werner parameter is constant (see Appendix B.1).

1.3 Entanglement Swapping

Once we have generated elementary entanglement, we can use it to **create entanglement over longer distances** by entanglement swapping. This UNIT-PROTOCOL takes in input two entangled pairs, say between nodes $A-B$ and $B-C$, and outputs a longer link $A-C$ **of reduced quality**.

1.3.1 Waiting Time for Entanglement Swapping

The waiting time for the generation of elementary entanglement T_0 is the base for the induction presented in the following. We define our inductive step assuming that we have found an expression for T_n and we want to construct T_{n+1}.

In order to perform the entanglement swap to produce a single (2^{n+1})-hop link, a node needs to wait for the production of two (2^n)-hop links, one on each side.

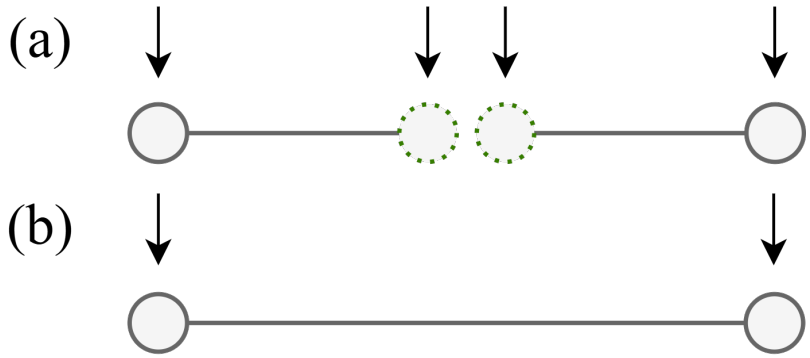

Figure 1.2: Two entangled links are (a) generated probabilistically, and then (b) swapped probabilistically into a single longer link of reduced quality.

We denote the waiting time for the two 2^n-hop links to be ready for swapping by $T_n^{(A)}$ and $T_n^{(B)}$, i.i.d. copies of T_n (see Appendix A.2).

Time until both pairs are available

We introduce a new random variable M_n, modeling the time *before* both pairs are available

$$M_n = g_T(T_n^{(A)}, T_n^{(B)}), \tag{1.3}$$

where the function g_T is generally defined as the maximum of the two waiting times

$$g_T(t_A, t_B) = \max(t_A, t_B). \tag{1.4}$$

Note that this is because **as soon as the two pairs are available**, the entanglement swap is attempted, thus the waiting time necessary is just the maximum of the two input links' waiting times.

Thus, M_n represents the waiting time for a single attempt of entanglement swapping, and is distributed according to its PDF

$$\Pr(M_n = t) = \sum_{\substack{t_A, t_B: \\ t = \max(t_A, t_B) + t_c}} \Pr(T_n^{(A)} = t_A \text{ and } T_n^{(B)} = t_B). \tag{1.5}$$

Classical Communication Time

What denoted as t_c is the time needed for classical communication and local operations, which is a fixed time that is added to the waiting time for the entanglement generation.

In particular this is the time needed for the *heralding signal*, i.e. the classical signal that is sent to the nodes to inform them that the entanglement is ready. This signal is sent as soon as the entanglement is generated, and the nodes need to wait for it to arrive before they can know the outcome of the process and proceed with the rest of the protocol.

Number of steps required

We introduce now K_n, the random variable that models **the number of steps k until the first successful swap** at level n. It follows a geometric distribution

$$\Pr(K_n = k) = p_{\text{swap}}(1 - p_{\text{swap}})^{k-1} \tag{1.6}$$

where p_{swap} is the success probability of the entanglement swapping, which is independent of the state of the two input links, i.e. is a constant hardware parameter for the model.

Derivation of T_{n+1}

The derivation of T_{n+1} requires us to combine the random variable for the number of steps required K_n and the random variable for the waiting time for one attempt M_n.

In order to find the relation between M_n and T_{n+1}, note that when entanglement swap fails, the two input are lost and need to be regenerated. The regeneration of entanglement after each failing entanglement swap **adds to the waiting time**.

Thus, T_{n+1} is a *compound random variable*: it is the sum of K_n copies of M_n.

Since the number of entanglement swaps K_n is geometrically distributed, we say that T_{n+1} is a *geometric compound sum* of K_n copies of M_n, denoted as

$$T_{n+1} = \sum_{k=1}^{K_n} M_n^{(k)}. \tag{1.7}$$

Derivation of the PDF of T_{n+1}

The probability distribution of the waiting time T_{n+1}

$$\Pr(T_{n+1} = t) = \Pr\left[\left(\sum_{k=1}^{K_n} M_n^{(k)}\right) = t\right]$$

is computed as the marginal of the waiting time conditioned on a fixed number of attempts

$$\Pr(T_{n+1} = t) = \sum_{k=1}^{\infty} \Pr(K_n = k) \cdot \Pr\left[\left(\sum_{j=1}^{k} M_n^{(j)}\right) = t\right]. \tag{1.8}$$

Plugging in the expressions for $\Pr(K_n = k)$ (Equation 1.6) and recalling that the sum of k copies of M_n can be expressed as a convolution (see Appendix A.1), we get

$$\Pr(T_{n+1} = t) = \sum_{k=1}^{\infty} p_{\text{swap}}(1 - p_{\text{swap}})^{k-1} \left(*_{j=1}^{k} m\right) \tag{1.9}$$

where, from Equation 1.5

$$m(t) = \Pr(M_n = t) = \sum_{\substack{t_A, t_B: \\ t = \max(t_A, t_B) + t_c}} \Pr(T_n^{(A)} = t_A \text{ and } T_n^{(B)} = t_B). \tag{1.10}$$

For simplicity, in the following we may use interchangeably the notation M_n and M, as well as $T_n^{(A)}$ and T_A, omitting that we are considering the n-th level of the protocol to derive the $n+1$-th level.

1.3.2 Werner Parameter for Entanglement Swapping

Considering two entangled pairs, respectively with Werner parameters w_A and w_B, the output Werner parameter, if we do not consider decoherence, will be

$$w_{\text{out}} = w_A \cdot w_B. \tag{1.11}$$

Since $w \in [0, 1]$, the output Werner parameter will be less than the input ones. In addition, when the first of the two pairs is generated, it has to wait for the elementary generation of the other; **during this time the first generated pair decoheres**. In particular, a Werner state $\rho(w)$ residing in memory for a time $\Delta(t)$ will transform into the Werner state $\rho(w_{\text{decayed}})$ with

$$w_{\text{decayed}} = w \cdot e^{-\Delta t / t_{\text{coh}}} \tag{1.12}$$

where t_{coh} is the joint coherence time of the two quantum memories holding the qubits.

Denote by A and B the input links to the entanglement swap and denote by (t_A, w_A) and (t_B, w_B) their respective delivery times and Werner parameters. Without loss of generality, suppose that the link A is produced after link B, i.e. $t_A \geq t_B$.

Link A is produced last, so the entanglement swap will be performed directly after its generation and hence link A will enter the entanglement swap with Werner parameter w_A. Link B is produced earliest and will therefore decohere until production of link A.

It follows that B's Werner parameter decoheres accordingly to Equation 1.12, and therefore is, immediately before the swap, equal to

$$w'_B = w_B \cdot e^{-|t_A - t_B| / t_{\text{coh}}}.$$

Thus, the entanglement swap would produce the (2^{n+1})-hop state with Werner parameter

$$\begin{aligned} w_{\text{out}} &= w_A \cdot w'_B \\ &= w_A \cdot w_B \cdot e^{-|t_A - t_B| / t_{\text{coh}}}. \end{aligned} \tag{1.13}$$

Notice that choosing $t_A \leq t_B$ would have lead to the same result. In the following, we use the prime notation on both w_A and w_B to denote that either of the two links can be the one produced first (and thus the one that decoheres).

Induction. In the **base case** (entanglement generation), the **waiting time and Werner parameter are uncorrelated** because we model the attempts at generating single-hop entanglement to be independent and to each take equally long.

At the **recursive step**, we model at the same time the Werner Parameter and the Waiting Time as a joint random variable.

The key idea is the following: if the entanglement swap fails, then the (2^{n+1})-hop link with its Werner parameter will never be produced since both initial 2^n-hop entangled pairs are lost. Thus, **for the waiting time we should consider the sum of the waiting times of all the attempts, but for the Werner parameter we should only consider the last successful attempt**.

Let's explain it further. In order to find how to express the Werner parameter on level $n+1$, consider a sequence (m_j, v_j) of waiting times m_j and Werner parameters v_j, where j runs from 1 to the first successful swap k, with

- m_j representing the waiting time until the end of the entanglement swap that transforms two 2^n-hop links into a single (2^{n+1})-hop link,
- v_j representing the output link's Werner parameter if the swap is successful.

From previous results, we found that the total waiting time is given by $\sum_{j=1}^{k} m_j$, since waiting times add on top of each other. On the other hand, the output link is the last produced link and therefore its Werner parameter equals v_k.

For this reason, the waiting time needed to achieve the first 2^{n+1}-hop link and its Werner parameter are given by:

$$t_{n+1} = \sum_{j=1}^{k} m_j \tag{1.14}$$

$$w_{n+1} = v_k \tag{1.15}$$

or, in a more compact form

$$(t_{n+1}, w_{\mathrm{n+1}}) = \left(\sum_{j=1}^{k} m_j, v_k \right). \tag{1.16}$$

Derivation of W_{n+1}

Taking into account what stated above, we aim to derive the expression for the Werner parameter at level $(n+1)$.

Let $W_s(t)$ be the **average Werner parameter** of the output link of **one attempt**, given that it succeeds and finishes at time t

$$W_s(t) = \sum_{\substack{t_A, t_B: \\ t = \max(t_A, t_B) + t_c}} \Pr(T_A = t_A, T_B = t_B) \cdot [p_{\mathrm{swap}} \cdot w_{\mathrm{swap}}](t_A, t_B). \tag{1.17}$$

The number of swaps k that need to be performed until the first successful one is an instance of the random variable K_n. We thus iterate over all pair of possible input Werner parameters for each k by convolution and get

$$W_{n+1}(t) = \sum_{k=1}^{\infty} p_{\mathrm{swap}} (1 - p_{\mathrm{swap}})^{k-1} \left[\left(*_{j=1}^{k-1} m \right) * (m \cdot W_s) \right] \tag{1.18}$$

where m is defined in Eq. 1.10.

W_{n+1} represents the weighted average of the Werner parameter of the output link, over all possible sequences of failed attempts, followed by a successful one:

- all possible sequences of failed attempts are represented by the $(k-1)$ convolutions of m,
- the successful attempt is represented by the convolution of m and W_s.

1.4 Entanglement Distillation

Entanglement distillation takes in input two low-quality entangled pairs and produces a single high-quality one.

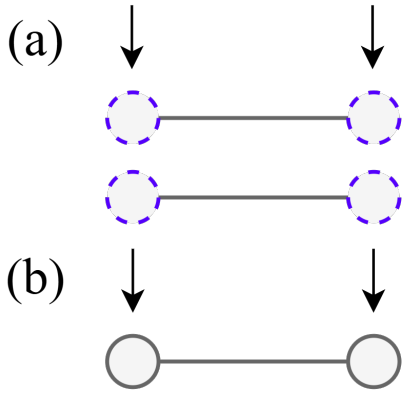

Figure 1.3: Two entangled links are (a) generated probabilistically, and then (b) distilled probabilistically into a single link of higher quality.

For entanglement distillation, since we consider the BBPSW protocol from Bennett et al. [Ben+96], the success probability p_{dist} depends on the Werner parameters of the input links A and B:

$$p_{\text{dist}} = \frac{1 + w'_A w'_B}{2} \tag{1.19}$$

where the primed notation denotes the decoherence of one of the two input links for a time $\Delta t = |t_A - t_B|$, as in Equation 1.12. This means that the success probability of one attempt is **time-dependent**, as it depends on the times (t_A, t_B) at which the two entangled links are ready. Thus, in general, the random variable for the number of attempts K **is correlated** to the one for the waiting times of the input links $M^{(j)}$. To make this correlation between

K and M explicit, we introduce a binary random variable Y describing success (1) or failure (0) of a single attempt, subjected to the success probability for the distillation.

We then **separate the expressions** for the waiting time probability distribution of a successful and failed attempt, respectively denoted as $P_s(t)$ and $P_f(t)$. These are computed by iterating over all possible combinations of the input links' generation times (t_A, t_B) that lead to a waiting time t for a single attempt.

We then express the total waiting time distribution T_{n+1} and the Werner parameter W_{n+1} as those of the successful attempt averaged by all possible sequences of failed attempts, where the weighted average is efficiently computed using convolution (see Appendix A.1).

1.4.1 Waiting Time for Entanglement Distillation

Here, we detail the process described above.

A successful attempt of distillation outputs a Werner state with quality:

$$w_{\text{dist}} = \frac{w'_A + w'_B + 4w'_A w'_B}{6p_{\text{dist}}} \tag{1.20}$$

where the p_{dist} is the success probability of the distillation, as in Equation 1.19.

To compute the waiting time distribution we consider the generation time for a successful or failed attempt separately and use the joint distribution of M and Y.

We define the joint distribution that **one attempt** succeeds or fails and takes time t respectively as

$$P_s(t) = \Pr(M = t, Y = 1) = \sum_{\substack{t_A, t_B: \\ t = \max(t_A, t_B) + t_c}} \Pr(T_A = t_A, T_B = t_B) \cdot p_{\text{dist}}(t_A, t_B) \tag{1.21}$$

$$P_f(t) = \Pr(M = t, Y = 0) = \sum_{\substack{t_A, t_B: \\ t = \max(t_A, t_B) + t_c}} \Pr(T_A = t_A, T_B = t_B) \cdot [1 - p_{\text{dist}}(t_A, t_B)] \tag{1.22}$$

where t_c is the time used for classical communication and local operation.

The total waiting time for **all attempts** can be obtained by

$$\Pr(T_{n+1} = t) = \sum_{k=1}^{\infty} \left[\left(*_{j=1}^{k-1} P_f \right) * P_s \right] \tag{1.23}$$

where the sum over k considers all the possible numbers of attempts by convoluting, for each k, the single success with the $(k-1)$ failures.

Neither P_f or P_s characterizes a random variable since they are joint distributions including Y, that is to say, P_s and P_f do not sum up to 1. Instead, we have

$$\sum_t P_s(t) + \sum_t P_f(t) = 1. \tag{1.24}$$

The convolution here cannot be simply interpreted as a sum of two random variables. Instead, it is the summed waiting time conditioned on the success/failure of each attempt.

To summarize, the full computation goes as follows:

- we compute P_f and P_s: the joint distribution that one attempt succeeds or fails and takes time t.
- once these are computed, we can compute T_{n+1} and W_{n+1} as convolutions of $(k-1)$ unsuccessful attempts and one successful one, summing over all possible numbers of attempts k.

1.4.2 Werner Parameter for Entanglement Distillation

We use the separate expressions for the successful and failed attempts to derive the Werner parameter of the end-to-end link.

First, we compute the **average** Werner parameter for a **single attempt**, given that entanglement distillation succeeds at time t

$$W_s(t) = \frac{\sum_{\substack{t_A,t_B: \\ t=\max(t_A,t_B)+t_c}} \Pr(T_A = t_A, T_B = t_B) \cdot [p_{\text{dist}} \cdot w_{\text{dist}}](t_A, t_B)}{P_s(t)}. \tag{1.25}$$

Next, we take a weighted average of W_s over all possible sequences of failed attempts, followed by a single successful attempt:

$$W_{n+1} = \frac{\sum_{k=1}^{\infty} \left[\left(*_{j=1}^{k-1} P_f^{(j)} \right) * (P_s \cdot W_s) \right](t)}{\Pr(T_{n+1} = t)}. \tag{1.26}$$

This give us the Werner parameter of the link in output of the distillation protocol.

An important remark. Li *et al.* [LCE21] achieves a more complex model with respect to Brand *et al.* [BCE20] through the technique of separating the expressions for a success and a failure. Thus, the *more general* model in [LCE21] can be also used to derive, in an

alternative way, the expressions for the Werner parameter and the waiting time for entanglement swapping (Equations 1.9 and 1.18). As a matter of fact, the model in [LCE21] uses Equations 1.23 and 1.26 for both entanglement distillation and entanglement swapping, simply plugging the corresponding success probability p_{swap} or p_{dist}, and Werner parameter w_{swap} or w_{dist}.

In the following, when converting these equations in the Fourier space, we will use the more general model from [LCE21], and thus we will not need to distinguish between the two protocols.

1.5 Closed Expressions in the Fourier Domain

Fourier (and Laplace) transforms are employed in the study of quantum repeater chains as powerful tools to analyze stochastic processes [KV21]. They **simplify the modeling and computation** of quantum network performances under the effect of memory decay. As a matter of fact, some of the expressions derived in the previous sections can be expressed in the Fourier domain, leading to more efficient and elegant computations.

One example of operation that can be efficiently computed in the Fourier domain is the convolution of two or more probability distributions (see Appendix A.1), which can be expressed as a product of their Fourier transforms. This is a direct consequence of the linearity of the Fourier transform, which converts convolutions into element-wise multiplication.

The Fourier transform acts on a finite sequence of numbers, which corresponds to a sampled version of the original continuous function. This discretization is necessary for practical computations and is typically achieved by using the Fast Fourier Transform (FFT) algorithm.

1.5.1 Waiting Time

Since the discrete Fourier transform acts on a finite sequence of numbers we first truncate the probability distribution at a fixed time.

The Fourier transform is a linear map, converting convolutions into element-wise multiplication. As a consequence, taking the Fourier transform of both sides of Equation 1.23, and then taking the inverse Fourier transform, yields

$$\Pr(T_{n+1} = t) = \mathcal{F}^{-1}\left[\frac{\mathcal{F}[P_s]}{1 - \mathcal{F}[P_f]}\right](t), \tag{1.27}$$

a more elegant and efficient way to compute the waiting time distribution. In the following, we describe in more details how this is achieved.

Full Derivation. We start from Equation 1.23, the probability distribution of the total waiting time T_{n+1} for all attempts, which we restate here for convenience:

$$\Pr(T_{n+1} = t) = \sum_{k=1}^{\infty} \left[\left(*_{j=1}^{k-1} P_f^{(j)} \right) * P_s \right](t). \tag{1.28}$$

Applying the Fourier transform $\mathcal{F}$ to both sides, we get:

$$\mathcal{F}[\Pr(T_{n+1} = t)] = \sum_{k=1}^{\infty} \left[\left(\prod_{j=1}^{k-1} \mathcal{F}[P_f^{(j)}] \right) \cdot \mathcal{F}[P_s] \right](t). \tag{1.29}$$

For k attempts, the productory $\prod_{j=1}^{k-1} \mathcal{F}[P_f^{(j)}]$ represents the convolution of the failure distributions in the Fourier domain. From the assumption made, these distributions $P_f^{(j)}$ are identical copies (see Appendix A.2), so:

$$\prod_{j=1}^{k-1} \mathcal{F}[P_f^{(j)}] = (\mathcal{F}[P_f])^{k-1} \tag{1.30}$$

which means that the equation simplifies to

$$\mathcal{F}[\Pr(T_{n+1} = t)] = \sum_{k=1}^{\infty} \left((\mathcal{F}[P_f])^{k-1} \cdot \mathcal{F}[P_s] \right). \tag{1.31}$$

Using the **identity for the sum of a geometric series** $\sum_{k=1}^{\infty} x^{k-1} = \frac{1}{1-x}$, we can state that

$$\sum_{k=1}^{\infty} (\mathcal{F}[P_f])^{k-1} = \frac{1}{1 - \mathcal{F}[P_f]} \tag{1.32}$$

thus, incorporating the success term $\mathcal{F}[P_s]$, we get:

$$\mathcal{F}[\Pr(T_{n+1} = t)] = \frac{\mathcal{F}[P_s]}{1 - \mathcal{F}[P_f]} \tag{1.33}$$

Finally, applying the inverse Fourier transform $\mathcal{F}^{-1}$, we obtain:

$$\Pr(T_{n+1} = t) = \mathcal{F}^{-1} \left[\frac{\mathcal{F}[P_s]}{1 - \mathcal{F}[P_f]} \right](t) \tag{1.34}$$

1.5.2 Werner Parameter

The transformation of Equation 1.26 to Fourier space involves the same concepts and yields

$$W_{n+1} = \mathcal{F}^{-1}\left[\frac{\mathcal{F}[P_s \cdot W_s]}{1 - \mathcal{F}[P_f]}\right] \frac{1}{\Pr(T_{n+1} = t)}(t). \tag{1.35}$$

Again, the convolution has become a productory of identical copies of the same random variable; the resulting term, exponentiated to $(k-1)$, is then transformed through geometric series, yielding the final expression.

Appendix A

Mathematical Background

A.1 Convolution

Convolution is a fundamental mathematical operation used to combine two functions to produce a third function, which represents how the shape of one function is modified by the other.

For two discrete functions a and b, the convolution $a * b$ is defined as:

$$(a * b)(z) = \sum_{x=0}^{z} a(x) \cdot b(z - x) \tag{A.1}$$

Convolution of two Probability Distributions. In the context of probability distributions, if X and Y are independent random variables with probability distribution functions p_X and p_Y, their sum $Z = X + Y$ has a probability distribution function p_Z given by the convolution of p_X and p_Y:

$$p_Z(z) = \sum_{x=0}^{z} p_X(x) \cdot p_Y(z - x) \tag{A.2}$$

Thus, convolution is used to determine the probability distribution of the sum of independent random variables by combining their individual probability distributions.

Example. Consider this simple example. Let X and Y discrete independent random variables.

The probability distribution $\Pr(Z = z)$ of the random variable $Z = X + Y$ is computed as follows:

x	$\Pr(X = x)$	y	$\Pr(Y = y)$	z	$\Pr(Z = z)$	Derivation of $\Pr(Z = z)$
0	0.2	0	0.3	0	0.06	$0.2 \cdot 0.3$
1	0.5	1	0.4	1	0.23	$0.2 \cdot 0.4 + 0.5 \cdot 0.3$
2	0.3	2	0.3	2	0.35	$0.2 \cdot 0.3 + 0.5 \cdot 0.4 + 0.3 \cdot 0.3$
				3	0.27	$0.5 \cdot 0.3 + 0.3 \cdot 0.4$
				4	0.09	$0.3 \cdot 0.3$

Note that $\Pr(Z = z)$ has five elements (all the possible sums), and it is valid as its probabilities sum to 1.

Associativity. The convolution operator $*$ is associative, meaning that for any three functions a, b, and c:

$$(a * b) * c = a * (b * c) \tag{A.3}$$

A.2 Random Variables

In this section, we fix notation on random variables and operations on them.

Most random variables in the context of quantum repeaters

- are discrete,
- have as domain a subset of nonnegative integers.

PDF

Let X be such a random variable, then its probability distribution function is a map

$$p_X : x \mapsto \Pr(X = x) \tag{A.4}$$

which describes the probability that its outcome will be $x \in \{0, 1, 2, \ldots\}$.

CDF

Equivalently, X is described by its cumulative distribution function

$$\Pr(X \leq x) = \sum_{y=0}^{x} \Pr(X = y), \tag{A.5}$$

which is transformed to the probability distribution function as

$$\Pr(X = x) = \Pr(X \leq x) - \Pr(X \leq x - 1). \tag{A.6}$$

Independent Random Variables

Two random variables X and Y are independent if

$$\Pr(X = x \text{ and } Y = y) = \Pr(X = x) \cdot \Pr(Y = y) \tag{A.7}$$

for all x and y in the domain.

Copies of a Random Variable

By a *copy* of X, we mean a fresh random variable which is independent from X and identically distributed (i.i.d.). We will denote a copy by a superscript in parentheses. For example, $X^{(1)}$, $X^{(142)}$ and $X^{(A)}$ are all copies of X.

Expected Value

The mean of X is denoted by

$$E[X] = \sum_{x=0}^{\infty} \Pr(X = x) \cdot x \tag{A.8}$$

and can equivalently be computed as

$$E[X] = \sum_{x=1}^{\infty} \Pr(X \geq x). \tag{A.9}$$

Function of Random Variables

If f is a function which takes two nonnegative integers as input, then the random variable $f(X, Y)$ has probability distribution function defined as

$$\Pr(f(X,Y) = z) = \sum_{\substack{x=0,y=0 \\ f(x,y)=z}}^{\infty} \Pr(X = x \text{ and } Y = y). \tag{A.10}$$

Sum of Random Variables

An example of such a function is addition.

Define $Z = X + Y$ where X and Y are independent, then the probability distribution p_Z of Z is given by

$$p_Z(z) = \Pr(Z = z) = \sum_{\substack{x=0,y=0 \\ x+y=z}}^{\infty} \Pr(X = x \text{ and } Y = y). \tag{A.11}$$

But since $y = z - x$, this is equivalent to

$$p_Z(z) = \Pr(Z = z) = \sum_{x=0}^{z} \Pr(X = x \text{ and } Y = z - x) \tag{A.12}$$

$$= \sum_{x=0}^{z} \Pr(X = x) \cdot \Pr(Y = z - x) \tag{A.13}$$

$$= \sum_{x=0}^{z} p_X(x) \cdot p_Y(z - x) \tag{A.14}$$

which is the convolution of the distributions p_X and p_Y (see A.2), denoted as $p_Z = p_X * p_Y$.

Notice that, since the convolution operator $*$ is associative (see A.3), writing $a * b * c$ is well-defined, for functions a, b, c, from the nonnegative integers to the real numbers.

In general, **the probability distribution of sums of independent random variables equals the convolutions of their individual probability distribution functions**.

A.3 Geometric Distribution

The geometric distribution is a discrete probability distribution that models the number of trials needed to achieve the first success in a sequence of independent Bernoulli trials, each with the same success probability p.

A.3.1 Probability Distribution Function (PDF)

The probability distribution function (PDF) of a geometric distribution gives the probability that the first success occurs on the t-th trial. It is defined as:

$$\Pr(T = t) = p \cdot (1 - p)^{t-1} \quad \text{for} \quad t \in \{1, 2, 3, \ldots\}, \tag{A.15}$$

where:

- T is the random variable representing the trial number of the first success,
- p is the probability of success on each trial,
- $(1 - p)$ is the probability of failure on each trial.

This formula expresses that the first $t - 1$ trials must be failures, each occurring with probability $(1 - p)$, and the t-th trial must be a success (with probability p).

A.3.2 Cumulative Distribution Function (CDF)

The cumulative distribution function (CDF) of a geometric distribution gives the probability that the first success occurs on or before the t-th trial. It is defined as:

$$\Pr(T \leq t) = 1 - (1 - p)^t. \tag{A.16}$$

Derivation of the CDF The CDF of a geometric distribution can be derived from its PDF as follows:

$$\begin{aligned}
\Pr(T \leq t) &= 1 - \Pr(T > t) \\
&= 1 - \sum_{k=t+1} \Pr(T = k) \\
&= 1 - \left\{ p(1-p)^t + p(1-p)^{t+1} + p(1-p)^{t+2} + \ldots \right\} \\
&= 1 - p(1-p)^t \sum_{k=0} (1-p)^k \\
&= 1 - (1-p)^t \sum_{k=0} p(1-p)^k \\
&= 1 - (1-p)^t.
\end{aligned}$$

Appendix B

Numerical Examples

This appendix contains a numerical example of the protocols described. We consider the following parameters:

- the probability of success for entanglement generation is $p_{\text{gen}} = 0.5$,
- the probability of success for entanglement swapping is $p_{\text{dist}} = 0.5$,
- the initial Werner parameter of the generated links is $w_{\text{gen}} = 0.9$,

and assume that the system is not subjected to decoherence ($t_{\text{coh}} = \infty$).

B.1 Entanglement Generation

The probability distribution of the waiting time for entanglement generation is a geometric distribution (see Appendix A.3) with $p = 0.5$, shown in Figure B.1.

In the table below, we show the probability distribution of the waiting time for entanglement generation, as well as the corresponding output average Werner parameter, for some selected times.

Time Step	PMF	CDF	Werner
$t = 1$	0.50000	0.50000	0.90000
$t = 2$	0.25000	0.75000	0.90000
$t = 3$	0.12500	0.87500	0.90000
$t = 10$	0.00098	0.99902	0.90000
$t = 50$	0.00000	1.00000	0.90000

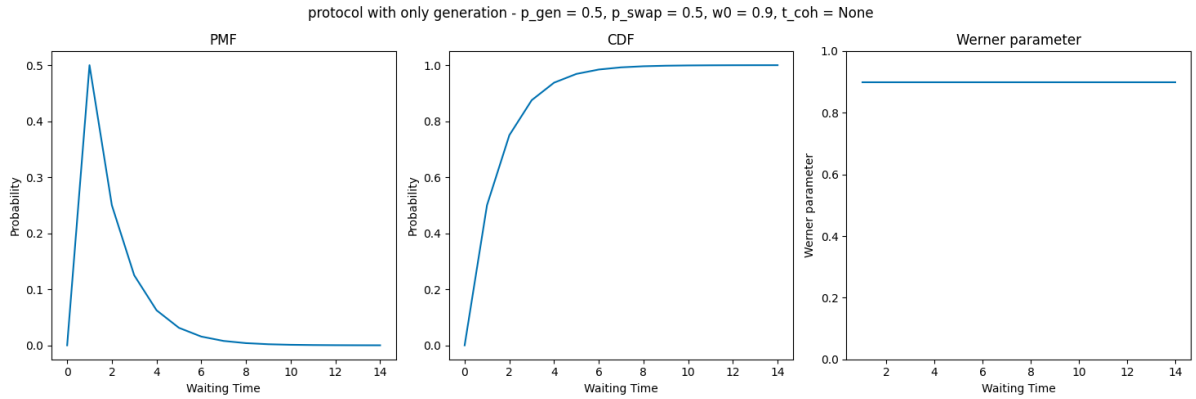

Figure B.1: Plots for entanglement generation. The PMF is a geometric distribution governed by $p_{\text{gen}} = 0.5$; for such high value, the CDF reaches a value close to 1 in few time steps. The Werner parameter is costant.

In this particular example, as $p_{\text{gen}} = (1 - p_{\text{gen}}) = 0.5$, The probability of achieving a link at time $t = i$ is $(0.5)^i$.

For entanglement generation, for all t, the average Werner parameter is the constant $w_0 = 0.9$. As a matter of fact, indipendently from the number of attempts needed to achieve a link, its quality will always be the same.

B.2 Entanglement Distillation

The probability distribution of the waiting time for entanglement distillation is shown in Figure B.2.

In the table below, we show, for some selected times, the PDF, the CDF, and the corresponding Werner parameter for the waiting time of entanglement distillation.

Time Step	PMF	CDF	Werner
$t = 1$	0.22625	0.22625	0.92818
$t = 2$	0.28819	0.51444	0.92818
$t = 3$	0.19739	0.71183	0.92818
$t = 10$	0.00460	0.99370	0.92818
$t = 50$	0.00000	1.00000	0.92819

In the following, we show the calculations for $t = 1, 2$ rows.

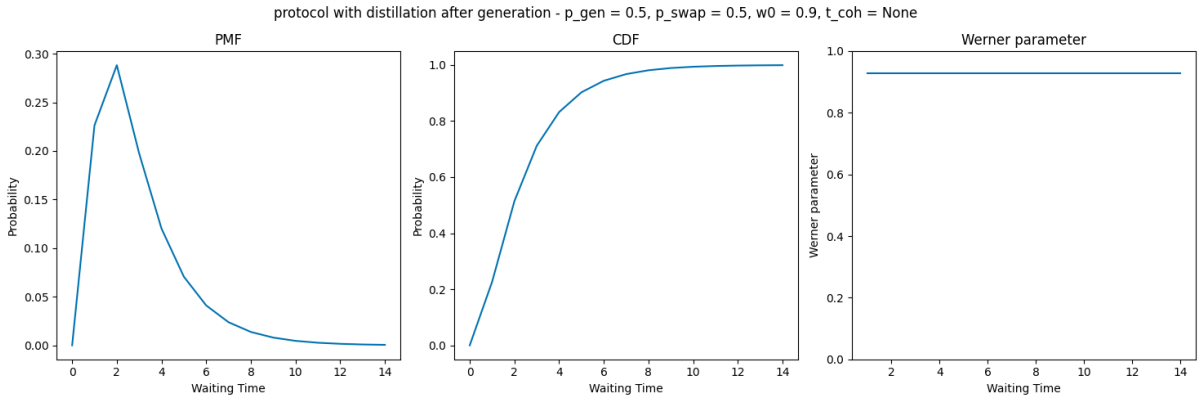

Figure B.2: Plots for entanglement distillation, after generation. The CDF for the waiting time, for all the values of t, is shifted to the right with respect to Figure B.1 of entanglement generation only. This indicates longer waiting times. The Werner parameter is again costant, as no decoherence is considered in this example.

Probability Distribution Function Here, we show the computation of the probability distribution function $\Pr(T_\mathrm{d} = t)$ for the waiting time of entanglement distillation, for some selected times.

For $t = 1$, we consider the case when the distillation suceeds immediately. Hence we take into consideration only the random variable P_s: to achieve the distilled link at time $t = 1$, the protocol has to succeeds at its first attempt:

$$\Pr(T_\mathrm{d} = 1) = P_s(t = 1) = 0.5 \cdot 0.5 \cdot 0.905 = 0.22625 \tag{B.1}$$

where we plugged $t = 1$ in the Equation 1.21 for one successful attempt of distillation.

For $t = 2$, we have to consider both failure and success random variables described in Equation 1.21, which are then used in Equation 1.23. Plugging $t = 2$, and expanding the equation for clarity, we get

$$\Pr(T_\mathrm{d} = 2) = P_s(2) + (P_f * P_s)(2) + (P_f * P_f * P_s)(2) + \dots \tag{B.2}$$

where we have theoretically an infinite number of terms, truncated in practical calculations.

We compute the values for a success and a failure at time $t = 2$ expanding the sum in Equation 1.21

$$P_s(2) = (0.25 \cdot 0.5 \cdot 0.905) + (0.5 \cdot 0.25 \cdot 0.905) + (0.25 \cdot 0.25 \cdot 0.905) = \tag{B.3}$$

$$= 0.25 \cdot (0.5 + 0.5 + 0.25) \cdot 0.905 = 0.28281 \tag{B.4}$$

$$P_f(2) = 0.25\,(0.5 + 0.5 + 0.25) \cdot 0.095 = 0.02968 \tag{B.5}$$

where we considered all the contributions given by both links succeding at times $t = 1, 2$, as the distillation starts when both input links are generated.

To consider distillation success after failure we compute $[P_s * P_f](2)$. To do that **we need to convolve the PDFs of P_s and P_f** (see Appendix A.2):

t	$P_s(t)$	$P_f(t)$	$[P_s * P_f](t)$	$[P_f * P_s * P_f](t)$
0	0	0	0	0
1	0.22625	0.02375	0	0
2	0.28281	0.02968	$5.3734 \cdot 10^{-3}$	0

where we computed $[P_s * P_f](2)$ according to Equation A.2:

$$[P_s * P_f](2) = 0 + 0.22625 \cdot 0.02325 + 0 = 5.3734 \cdot 10^{-3}. \tag{B.6}$$

Thus, the probability distribution function for $t = 2$ is, according to Equation B.2:

$$\Pr(T_\mathrm{d} = 2) = 0.28281 + 0.00537 = 0.28819 \tag{B.7}$$

where we disregarded the higher order terms, as they are all zero.

Werner Parameter We first compute the probability of success for the distillation operation, which is given by

$$p_d = \frac{1 + w_A w_B}{2} = \frac{1 + 2w_0}{2} \tag{B.8}$$

and is equal to

$$p_d = (1 + 0.81)/2 = 0.905. \tag{B.9}$$

The Werner parameter for the output links of the distillation operation is given by

$$w_d = \frac{w_A + w_B + 4w_A \cdot w_B}{6p_d} = \frac{2w_0 + 4w_0^2}{6p_d} \tag{B.10}$$

and is equal to

$$w_d = 0.92818. \tag{B.11}$$

Since we are not considering decoherence in this example, the average ouput Werner parameter of the distilled link is constant for all t.

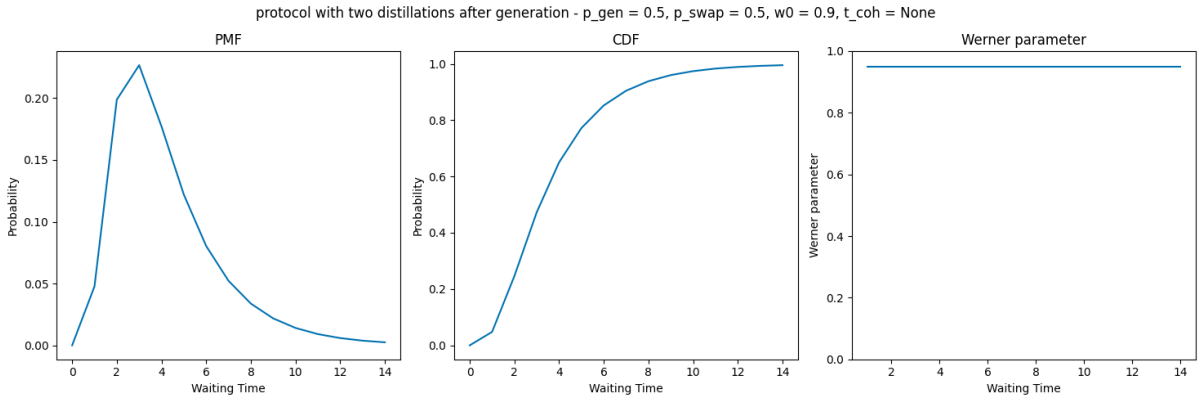

Figure B.3: The probability distribution of the waiting time for two-level entanglement distillation.

B.3 Two-Level Entanglement Distillation

Now, on top of the entanglement distillation, we add a second level of distillation. The probability distribution of the waiting time for this two-level entanglement distillation $\Pr(T_{dd} = t)$ is shown in Figure B.3.

In the table below, we show, for some selected times, the PDF, the CDF, and the corresponding Werner parameter for this protocol.

Time Step	PMF	CDF	Werner
$t = 1$	0.04764	0.04764	0.94948
$t = 2$	0.19884	0.24649	0.94948
$t = 3$	0.22670	0.47319	0.94948
$t = 10$	0.01406	0.97442	0.94948
$t = 50$	0.00000	1.00000	0.94948

Probability Distribution Function For $t = 1$, we have again the edge case for which we consider only an attempt succeeding at time $t = 1$, without failures

$$\Pr(T_{dd} = 1) = P'_s(1) = 0.22625 \cdot 0.93075 = 0.04764 \tag{B.12}$$

where we used the prime notation to differentiate this from the previous case.

For $t = 2$,

$$\Pr(T_{dd} = 2) = P'_s(2) + (P'_f * P'_s)(2) + (P'_f * P'_f * P'_s)(2) + \dots \tag{B.13}$$

is computed as before, and yields

$$\Pr(T_{dd} = 2) = 0.19884. \tag{B.14}$$

Werner Parameter The probability of success for the two-level distillation operation is

$$p_{dd} = \frac{1 + w_A w_B}{2} = \frac{1 + w_d^2}{2} = 0.93075. \tag{B.15}$$

The Werner parameter for the output links of the two-level distillation operation is given by

$$w_{dd} = \frac{w_A + w_B + 4w_A \cdot w_B}{6p_{dd}} = \frac{2w_d + 4w_d^2}{6p_{dd}} = 0.94974 \tag{B.16}$$

and, again, it is constant for all the times t as no decoherence is considered.

B.4 Entanglement Swapping

The probability distribution of the waiting time for entanglement swapping is shown in Figure B.4.

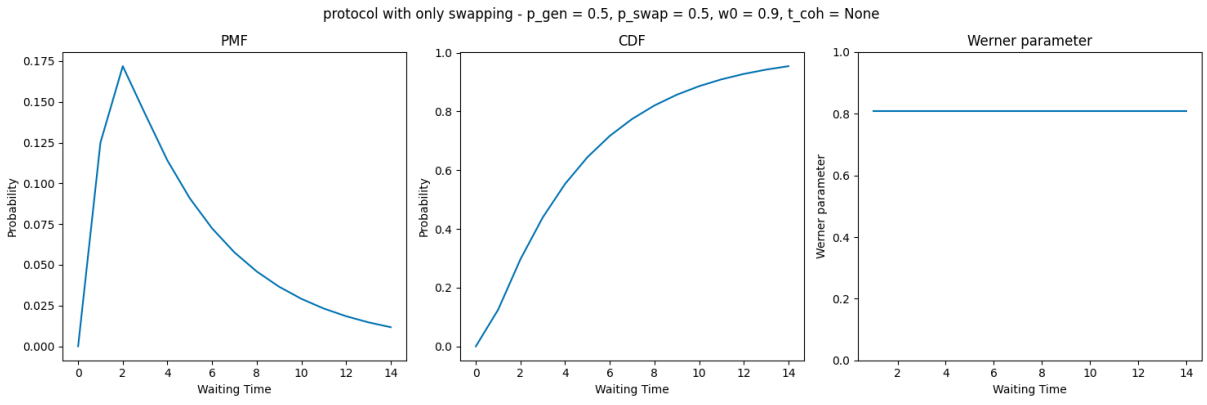

Figure B.4: The probability distribution of the waiting time for entanglement swapping.

In the table below, we show, for some selected times, the PDF, the CDF, and the corre-

sponding Werner parameter for the waiting time of entanglement swapping.

Time Step	PMF	CDF	Werner
$t = 1$	0.12500	0.12500	0.81000
$t = 2$	0.17188	0.29688	0.81000
$t = 3$	0.14258	0.43945	0.81000
$t = 10$	0.02913	0.88595	0.81000
$t = 50$	0.00000	0.99999	0.81000

Bibliography

[Ben+96] Charles H Bennett et al. "Purification of noisy entanglement and faithful teleportation via noisy channels". In: *Physical review letters* 76.5 (1996), p. 722.

[BCE20] Sebastiaan Brand, Tim Coopmans, and David Elkouss. "Efficient Computation of the Waiting Time and Fidelity in Quantum Repeater Chains". In: *IEEE Journal on Selected Areas in Communications* 38.3 (2020), pp. 619–639. ISSN: 1558-0008. DOI: 10.1109/jsac.2020.2969037. URL: http://dx.doi.org/10.1109/JSAC.2020.2969037.

[KV21] Viacheslav V. Kuzmin and Denis V. Vasilyev. "Diagrammatic technique for simulation of large-scale quantum repeater networks with dissipating quantum memories". In: *Physical Review A* 103.3 (Mar. 2021). ISSN: 2469-9934. DOI: 10.1103/physreva.103.032618. URL: http://dx.doi.org/10.1103/PhysRevA.103.032618.

[LCE21] Boxi Li, Tim Coopmans, and David Elkouss. "Efficient Optimization of Cutoffs in Quantum Repeater Chains". In: *IEEE Transactions on Quantum Engineering* 2 (2021), pp. 1–15. ISSN: 2689-1808. DOI: 10.1109/tqe.2021.3099003. URL: http://dx.doi.org/10.1109/TQE.2021.3099003.