On noise in swap ASAP repeater chains: exact analytics, distributions and tight approximations

K. Goodenough[1], T. Coopmans[2], and D. Towsley[1]

[1]College of Information and Computer Science, University of Massachusetts Amherst, 140 Governors Dr, Amherst, Massachusetts 01002, USA

[2]Leiden Institute of Advanced Computer Science, Leiden University, Leiden, The Netherlands

Losses are one of the main bottlenecks for the distribution of entanglement in quantum networks, which can be overcome by the implementation of quantum repeaters. The most basic form of a quantum repeater chain is the swap ASAP repeater chain. In such a repeater chain, elementary links are probabilistically generated and deterministically swapped as soon as two adjacent links have been generated. As each entangled state is waiting to be swapped, decoherence is experienced, turning the fidelity of the entangled state between the end nodes of the chain into a random variable. Fully characterizing the (average) fidelity as the repeater chain grows is still an open problem. Here, we analytically investigate the case of equally-spaced repeaters, where we find exact analytic formulae for all moments of the fidelity up to 25 segments. We obtain these formulae by providing a general solution in terms of a *generating function*; a function whose n'th term in its Maclaurin series yields the moments of the fidelity for n segments. We generalize this approach as well to a *global cut-off* policy — a method for increasing fidelity at the cost of longer entanglement delivery times — allowing for fast optimization of the cut-off parameter by eliminating the need for Monte Carlo simulation. We furthermore find simple approximations of the average fidelity that are exponentially tight, and, for up to 10 segments, the full distribution of the delivered fidelity. We use this to analytically calculate the secret-key rate, both with and without binning methods.

1 Introduction

Quantum communication allows for benefits inaccessible with classical communication alone, key examples including long-baseline telescopes [1, 2], secret sharing [3, 4] and distributed quantum computation [5, 6]. These protocols all rely on end-users sharing entanglement. A quantum internet [7] would allow for the distribution of such entanglement over large distances. One way to achieve this distribution is to transmit states over optical fibers; this has the benefit of being able to use pre-existing classical infrastructure [8].

One bottleneck of quantum communication is losses, where the success probability of a single photon arriving end-to-end over a distance L is given by $p = \exp\left(-\frac{L}{L_{\text{att}}}\right)$, $L_{\text{att}} \approx 22$ kilometers. A naive amplification of the signal to overcome this (as one does in the classical case) is prohibited by the no-cloning theorem. Fortunately, *quantum repeaters* [9] provide a solution. That is, by splitting the total length into n segments and generating entanglement over the segments, end-to-end entanglement can be created by performing so-called entanglement swaps [10]. Assuming that entanglement is swapped as soon as possible (*swap ASAP*) one can show that the state delivery rate now scales as $\mathcal{O}\left(\sqrt[n]{p}\right)$ (where we assume here and in the rest of the paper that the swaps are deterministic). Thus, while the rate still decays exponentially with the distance, the delivery rate increases significantly compared to the case without repeaters.

Although repeaters have the ability to increase the delivery rate, they also introduce additional *noise*, lowering the quality of the delivered entanglement. There are two contributions to this additional noise [11, 12]. First, noise arising from imperfections in the devices (such as state

preparation); second, noise arising from memory decoherence. Note that the first type of noise is deterministic[1]. In contrast, the second type of noise is random. This is due to the fact that a node needs to wait for a random amount of time until both its incident links are present before a swap can be performed. This random noise causes the fidelity of the end-to-end state to be a random variable.

The trade-off between increasing the rate and average fidelity complicates understanding what can be done with near-term devices, especially in combination with so-called *cut-off policies*. A cut-off policy is one that sets a threshold for when to discard the present entanglement. After the discard, the entanglement generation is started anew. Discarding generated entanglement reduces the rate at which entanglement is delivered, but, by a clever choice of a cut-off policy, will mitigate the time spent on distributing end-to-end entanglement with too low a fidelity. A cut-off policy thus allows for a trade-off between the rate and quality of the delivered end-to-end entanglement. Cut-off policies and their trade-offs have been studied, for example, in [11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24].

However, analytic expressions for the expected fidelity in the presence or absence of cut-off policies have been derived only up to 8 segments [21]. Here, we analytically calculate the average fidelity of the delivered state for homogeneous swap ASAP repeater chains consisting of up 25 segments. Furthermore, we derive a closed-form expressions for the probability distribution of the fidelity for up to 10 segments. While the analytical expressions grow unwieldy very fast, we find a closed-form expression of a so-called *generating function* — a function whose Maclaurin series coefficients correspond to the average noise parameters. This, in a sense, fully characterizes the average noise for arbitrary length swap ASAP repeater chains. Using tools from analytic combinatorics, we use this generating function to find simple approximations that are provably asymptotically tight, and which in practice are indistinguishable from exact results already for three and more segments. As a caveat, we assume for the exact analytics that the end nodes experience no decoherence, which holds when performing quantum key distribution (QKD) [25] (when the states at the end nodes can be measured directly) or can be justified in the case where the end nodes have significantly more resources than the repeater chain, see also [21] for a similar setup.

We note that a specific type of generating function, the probability generating function, has been used before in the noise analysis for more general quantum repeater protocols, ranging up to $n = 8$ segments for a full analysis of all moments of the fidelity [21]. Our generating function approach is more powerful, since it captures all probability generating functions at once; its n'th term is exactly the probability generating function associated with n segments. This thus provides an explanation for the expressions from [21], where the expressions were found through brute-force calculation. By taking appropriate derivatives/expansions of our generating function, we compute an analytical expression of the average fidelity for up to 25 segments within seconds, and find an analytical expression for the distribution of the fidelity for up to 10 repeater segments.

We furthermore show how most of our tools can be applied to a *global* cut-off policy. This cut-off policy does not require any communication between the nodes, making it especially relevant for near-term quantum networks.

We show how our tools can be used to 1) understand the effect of adding or removing repeaters for a given distance, 2) derive the distribution of the noise and how it relates to the cut-off threshold, 3) calculate the secret-key rate exactly for up to 10 segments, with and without so-called *binning strategies*, 4) understand the interaction between decoherence and losses in a manner that is independent of the number of repeaters.

All of our analytical results can be found in the provided Mathematica files [26].

We leave most of the technical details to the Appendix, and focus on the general concepts in the main text. We start by making explicit our model in Section 2, where we also discuss the global cut-off policy. In Section 3 we give a general overview of how to calculate exact analytics (with and without a global cut-off). Section 4 presents an overview of the derivation of the approximation for the average fidelity, which is based on techniques from analytic combinatorics. For our final result we focus in Section 5,

[1] Or, at the very least, approximately independent of when the different elementary links succeed.

not on the average fidelity, but on the fidelity distribution. We close each of the previous three sections by applying our tools to numerical exploration of repeater chains. We conclude with a discussion on possible extensions and touch upon follow-up work that extends some of our results numerically to both the inhomogeneous and multipartite setting in Section 6.

2 Swap ASAP repeater model

In this section we first present the basic setup of a swap ASAP protocol in the case of qubit depolarizing noise. We then discuss how our results extend to a large set of other noise models. Afterwards, we discuss the cut-off policy relevant for this work. We note that a similar setup was also investigated in [21].

2.1 Model details

The aim of this section is to lay out our model of a swap ASAP repeater chain. A repeater chain consists of n segments, separated by $n-1$ repeater nodes. We order the n segments from left to right by $i = 1, \ldots, n$. In a swap ASAP protocol, elementary link generation attempts are made in parallel for each segment in a given *round*. We assume that rounds are discretized (with duration equal to a single timestep), and label the round in which segment i succeeds by t_i, where $t_i \geq 1$. This is a common model in quantum networking [27, 28, 29, 30][2]. We note that this requires synchronization of all of the nodes in the repeater chain. This synchronization can be challenging when scaling up to larger repeater chains. Furthermore, quantum networks would require the nodes along all flows to synchronize their entanglement generation, which could become impractical for larger networks.

Link generation succeeds with a uniform probability p across all segments. We assume that link generation is attempted again only *after* successful end-to-end entanglement generation or after reaching a *cut-off condition* (see the next subsection). This is not only done to aid in the analysis, but it also prevents congestion[3] and is closer to the capabilities of near-term devices. For future quantum networks however, the idling of segments can lead to an under-utilization of resources, and thus to a lower entanglement throughput.

As soon as a repeater holds two entangled pairs a swap is performed, which we assume to be deterministic. Fig. 1 illustrates a swap ASAP repeater protocol, where $t_1 = 3$, $t_2 = 2$, $t_3 = 4$, $t_4 = 1$ and $t_5 = 2$.

The decoherence experienced in a given run will be determined by the number of rounds each node has to wait for until it can swap, along with the quality of the memory. We model noise in the system as depolarizing noise. That is, our states are of the form $\Lambda |\psi\rangle\langle\psi| + (1-\Lambda)\frac{\mathbb{I}}{4}$, where $\sqrt{2}|\psi\rangle = |00\rangle + |11\rangle$ and $\Lambda \in [0,1]$. From Λ, we calculate the fidelity to be $\Lambda + \frac{1-\Lambda}{4} = \frac{3}{4}\Lambda + \frac{1}{4}$. The usage of the parameter Λ is convenient — swapping two states with parameters Λ' and Λ'' gives a new state with parameter $\Lambda' \cdot \Lambda''$. Furthermore, decoherence over a single time step corresponds to the map $\Lambda \mapsto \lambda\Lambda$ for some $\lambda \in [0,1]$. Here and in what follows, we focus on Λ, since it completely characterizes the quantum state, such that other quality indicators (such as fidelity) can be extracted from it.

Let us consider a concrete example. In Fig. 1 the decoherence experienced by the memory in the repeater holding links 2 and 3 equals $\lambda^{|2-4|} = \lambda^2$, where λ is the noise parameter associated with a single round of decoherence.

Importantly, we ignore the noise that the end nodes incur if they had to keep their states stored until all other links have succeeded. This can be justified exactly when QKD is performed (in which case the states at the end nodes can be measured directly), or when the end nodes are assumed to have significantly better equipment or error correction. See [21] for a treatment that includes this noise.

With knowledge of which swaps are performed and how long certain memories decohere, it is possible to calculate the noise parameter Λ de-

[2] We note that a continuous-time model is also a possible model, which is especially accurate for small p (see e.g. [31]). While this model is out-of-scope for this manuscript, the techniques used in this manuscript (such as the recursion relation and the generating function approach) could be applied to the continuous-time setting as well.

[3] In the setting where there is only a single memory for each repeater node per connection, newly created entanglement may need to idle in memory as older states are kept around until end-to-end entanglement has been established.

scribing the final state. We show in the next section how to calculate the expected value of the final noise parameter. Finally, from linearity of expectation, the average fidelity is $\mathbb{E}[F] = \frac{3}{4}\mathbb{E}[\Lambda] + \frac{1}{4}$.

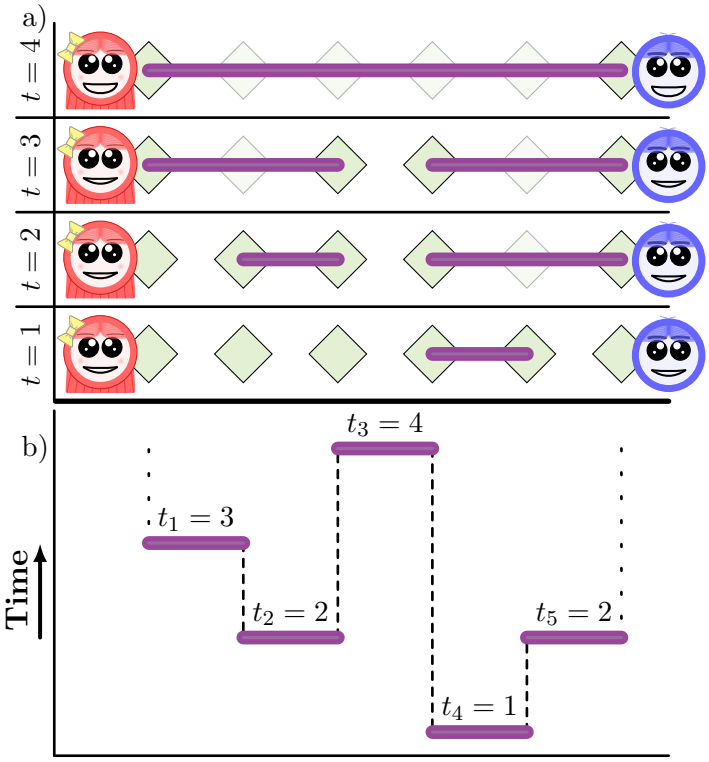

Figure 1: The used model for a swap ASAP repeater chain. In the top figure a) we show the evolution of the present entanglement in the network. At time $t = 1$ the fourth link gets created, at time $t = 2$ both link 2 and 5 are generated. Since we are considering the swap ASAP model, link 4 and 5 are swapped at this time. At time $t = 3$ link 1 is created and swapped with link 2. Finally, at time $t = 4$ the third link is created, which is immediately swapped on both sides, leading to end-to-end entanglement. In b) we summarize the total evolution of a single run. The decoherence experienced corresponds to the waiting time between the different links, indicated by dashed lines. The dotted lines correspond to having to wait until all links have been created, which are not relevant when performing tasks such as QKD.

We note that this model also includes noisy state generation and noisy swaps/measurements. Namely, if Λ is the end-to-end noise parameter for the case of perfect state generation, then the average noise parameter is given by $\Lambda \cdot (\prod_{i=1}^{n} \lambda_{\text{gen},i}) \left(\prod_{i=1}^{n-1} \lambda_{\text{swap},i}\right)$, where $\{\lambda_{\text{gen},i}\}$ are the noise parameters corresponding to the n noisy links, and $\{\lambda_{\text{swap},i}\}$ the noise parameters for the swaps/measurements. This results in a constant multiplicative factor to the Λ parameter. In what follows however, we will assume for simplicity that the only source of noise is decoherence, unless explicitly mentioned.

2.2 Extension to general noise models

Above, we chose the single-qubit depolarizing channel as the noise model. The properties that make this noise model amenable to analysis are the facts that decoherence corresponds to multiplication of the λ parameter, and that swapping two states corresponds to multiplication of the λ parameters. We derive in Appendix C that if the state is a two-qubit maximally-entangled state, then *any* qubit Pauli channel $\mathcal{N}$ has these properties. Consequently, the analysis in this work straightforwardly extends to any qubit Pauli noise model. The parameter update is achieved by a change of variables of $\mathcal{N}$: we start out with the usual parameterization of a Pauli channel,

$$\mathcal{N}(\rho) = p_I \cdot \rho + p_X \cdot X\rho X^\dagger + p_Y \cdot Y\rho Y^\dagger + p_Z \cdot Z\rho Z^\dagger \ ,$$

where $\{p_I, p_X, p_Y, p_Z\}$ is a probability distribution over the Pauli matrices I, X, Y, Z and ρ a single-qubit density matrix. We now re-parameterize $\mathcal{N} = \mathcal{N}_{\vec{\lambda}}$ using four new variables $\vec{\lambda} = (\lambda_1, \lambda_2, \lambda_3, \lambda_4)$, where

$$\begin{aligned}
\lambda_1 &= p_I + p_X + p_Y + p_Z = 1, \\
\lambda_2 &= p_I + p_X - p_Y - p_Z \ , \\
\lambda_3 &= p_I - p_X + p_Y - p_Z \ , \\
\lambda_4 &= p_I - p_X - p_Y + p_Z \ .
\end{aligned}$$

To verify that this change of variables satisfies the above criteria, it is straightforward to calculate that first applying $\mathcal{N}_{\vec{\lambda}}$, followed by applying $\mathcal{N}_{\vec{\lambda}'}$, is the same as applying $\mathcal{N}_{(\lambda_1 \cdot \lambda_1', \lambda_2 \cdot \lambda_2', \lambda_3 \cdot \lambda_3', \lambda_4 \cdot \lambda_4')}$. As a simple example, consider dephasing noise (which was also considered in [21]), where

$$\begin{aligned}
\mathcal{N}(\cdot) &= p(\cdot) + (1-p) Z(\cdot) Z^\dagger && (1) \\
&= \left(\frac{1+\lambda_2}{2}\right)(\cdot) + \left(\frac{1-\lambda_2}{2}\right) Z(\cdot) Z^\dagger && (2) \\
&= \mathcal{N}_{(1,1-2p,1-2p,1)}(\cdot) \ . && (3)
\end{aligned}$$

Multiplicativity under swapping is also satisfied for the general qubit Pauli channel. Denote by $\rho(\vec{\lambda})$ the resultant state after $\mathcal{N}_{\vec{\lambda}}$ has acted on one of the qubits of a maximally entangled

state. An entanglement swap on the states $\rho(\vec{\lambda})$ and $\rho(\vec{\lambda}')$ then yields the state $\rho(\lambda_1 \cdot \lambda_1', \lambda_2 \cdot \lambda_2', \lambda_3 \cdot \lambda_3', \lambda_4 \cdot \lambda_4')$ after correction, independently of the required correction. Since all the λ_i terms are multiplied point-wise, all of the analyses in this paper for the λ parameter for depolarizing noise can be directly translated to the individual λ_i parameters. In Appendix C, we also examine the case where the link between nodes is given by a maximally-entangled state on two *qudits*, i.e. the more general case where the local Hilbert space has dimension ≥ 2. Specifically, we show that the above multiplicativity properties hold for X-symmetric channels. These are channels corresponding to a probabilistic application of d-dimensional qudit Pauli operators $X^a Z^b$ (with $a, b \in \{0, 1, \ldots, d-1\}$), where the probability for $X^a Z^b$ is equal to the probability for $X^{-a} Z^b$. The qubit case follows since any qubit Pauli channel is X-symmetric. Our analysis thus is not only applicable to dephasing (also considered in [21]), but also to (biased) depolarizing noise and relevant noise models for GKP qudits, see [32, 33, 34, 35]. Our derivation is based on interpreting the composition of qudit Pauli channels as a type of convolution, which allows us to use tools from Fourier theory on finite Abelian groups.

2.3 The global cut-off policy

We now describe the cut-off policy considered in this paper, which we refer to as the *global* cut-off policy. This policy does not require any of the nodes to communicate information to non-neighboring nodes, making it particularly appealing for near-term quantum networks. We depict the global cut-off policy visually in Fig. 2. For completeness, we discuss in Appendix A a number of other policies, but which are outside the scope of this paper.

The global cut-off policy imposes a constraint on how long it takes to establish end-to-end entanglement. That is, starting from a chain with no entanglement present, a timer is initialized to zero and increases by one after every elementary link generation attempt. Then the ordinary swap ASAP protocol is followed. Two events can occur. If end-to-end entanglement is established before the timer reaches T_c, the timer is reset and the swap ASAP protocol is performed again. If, however, the timer reaches T_c (i.e. the cut-off is reached), the present entanglement is thrown away and the swap ASAP protocol starts anew.

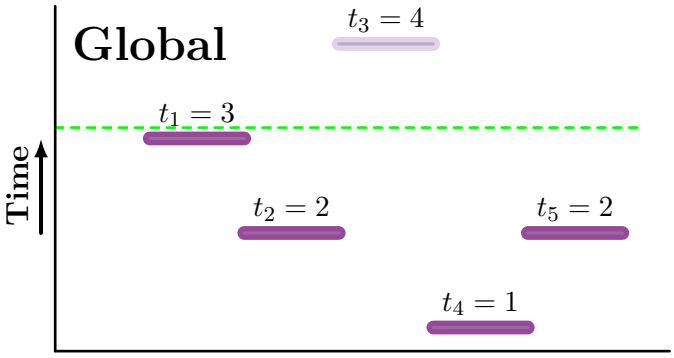

Figure 2: Graphical description of the global cut-off policy. Each of the nodes has agreed on a parameter T_c. For the global cut-off, a reset is performed when after T_c rounds end-to-end entanglement has not been established.

We assume for simplicity that classical communication is instantaneous, such that nodes know when to start entanglement generation again. We note that in general the effect of classical communication delays cannot be disregarded, especially in more complex distribution schemes [36, 37]. In Appendix E we additionally calculate the expected generation time in the case where all links only start generating again when the cut-off timer has been reached (but will not further consider this case in our results). This is especially appealing since it minimizes the classical communication between the nodes.

Fig. 2 shows an example of the global cut-off policy. In this case, the third link did not succeed before the timer $T_c = 3$, and the entanglement is discarded.

3 Recursion relation for the swap ASAP repeater model

With the considered model of the swap ASAP repeater protocol in hand, we will analyze the average end-to-end fidelity. We show that the average end-to-end fidelity can be interpreted as a *probability generating function*, which was shown before in [21]. We find a recursive formulation of the probability generating function, which allows us to go beyond the expressions for swap ASAP chains of length $n = 8$ from [21]. Furthermore, this recursive relation is key for the derivation of the *generating function* of the next section, which

is strictly more powerful than the *probability* generating function discussed in this section and in [21].

Subsection 3.1 considers the no cut-off policy. Subsection 3.1.1 describes how this approach can be extended to a global cut-off. We further discuss why a similar approach fails for the local cut-off policy (see Appendix A), and how to calculate the higher-order moments of Λ (which was also done before in [21]). In Subsection 3.2 we use our results to explore the performance of swap ASAP repeater chains.

3.1 Analytics without a cut-off policy

First, we need to calculate the probability $P(T=\bar{t})$ of observing a specific *instance* $\bar{t}$. Here an instance corresponds to a sequence $\bar{t}=(t_1,t_2,\ldots,t_n)$, where $t_i \in \{1,2,\ldots,\}$ indicate the round in which link i succeeds, see Section 2. A single link i succeeds in round t_i with probability $p(1-p)^{t_i-1} = (1-q)\,q^{t_i-1}$, where $q \equiv 1-p$. An instance $\bar{t}=(t_1,t_2,\ldots,t_n)$ for n links occurs with probability

$$P(T=\bar{t}) = \prod_{i=1}^{n}(1-q)\,q^{t_i-1} \tag{4}$$

$$= \left(\frac{1-q}{q}\right)^n q^{\sum_{i=1}^n t_i}\ , \tag{5}$$

for $\bar{t} \in \{1,2,\ldots,\}^n$. Let us now consider the decoherence for a given instance $\bar{t}$. Each node that is not an end node has two neighboring links, which succeed at times t_i and t_{i+1}. With the swap ASAP protocol, the swap occurs when the latest of the two links got created, i.e. at time $\max(t_i,t_{i+1})$. The link created first was created at time $\min(t_i,t_{i+1})$. Thus, the corresponding node waits $\max(t_i,t_{i+1}) - \min(t_i,t_{i+1}) = |t_i - t_{i+1}|$ rounds, during which the associated memory experiences decoherence.

Since a memory experiences a decay of λ for each time step, every node that is not an end node adds a multiplicative factor of $\lambda^{|t_i-t_{i+1}|}$, leading to a total decay parameter of $\Lambda = \lambda^{\sum_i^{n-1}|t_i-t_{i+1}|} = \lambda^{K(\bar{t})}$, where $K(\bar{t}) \equiv \sum_i^{n-1}|t_i-t_{i+1}|$ is the aggregate waiting time. We will also refer to K as the *roughness* parameter of $\bar{t}$.

The end nodes have to keep their state stored until end-to-end entanglement has been established. By definition, end-to-end entanglement is established at $\max(\bar{t})$, so that the first and last node store entanglement for $\max(\bar{t})-t_1$ and $\max(\bar{t})-t_n$ rounds, respectively. This leads to an additional total decoherence parameter of $\lambda^{2\max(\bar{t})-t_1-t_n}$. The average noise parameter $\mathbb{E}[\lambda_{\text{end}}]$ can then be written as

$$\left(\frac{1-q}{q}\right)^n \sum_{\bar{t}} \left(q^{\sum_i^n t_i}\right)\cdot\left(\lambda^{\left(2\max(\bar{t})-t_1-t_n\right)+\sum_{i=1}^{n-1}|t_i-t_{i+1}|}\right)\ .$$

As mentioned in the previous section, we now drop the $\lambda^{2\max(\bar{t})-t_1-t_n}$ factor.

We are interested in calculating the average noise parameter for n segments, i.e.

$$\begin{aligned}\mathbb{E}[\Lambda_n] &= \left(\frac{1-q}{q}\right)^n \sum_{\bar{t}} \left(q^{\sum_{i=1}^n t_i}\right)\cdot\left(\lambda^{\sum_{i=1}^{n-1}|t_i-t_{i+1}|}\right)\\ &= \sum_{k=0}^{\infty} P(K_n=k)\,\lambda^k\\ &= H_{K_n}(\lambda)\\ &\equiv H_n\ ,\end{aligned} \tag{6}$$

where $H_X(z)$ is shorthand for the *probability generating function* of the random variable X and the subscript indicates the associated number of segments n. It follows that $\mathbb{E}[\Lambda_n]$ is the *probability generating function* of the roughness parameter $K_n = \sum_{i=1}^{n-1}|t_i-t_{i+1}|$, first observed in [21]. This interpretation is useful for two reasons. First, any method to calculate $\mathbb{E}[\Lambda_n]$ (or $\mathbb{E}[\Lambda_n(T_c)]$), allows for the determination of any of the higher-order moments $\mathbb{E}[(\Lambda_n)^m]$. This is done by repeating such a method but with λ replaced by λ^m, since

$$\begin{aligned}\mathbb{E}[(\Lambda_n)^m] &= \sum_{k=0}^{\infty} P(K_n=k)\left(\lambda^k\right)^m\\ &= \sum_{k=0}^{\infty} P(K_n=k)\,(\lambda^m)^k\ ,\end{aligned}$$

noted first in [21]. Second, it is well known that a probability mass function can be reconstructed from the derivatives of its associated probability generating function; we exploit this idea in Section 5. Note that the interpretation of the average noise as the probability generating function of the aggregate waiting time is not specific to swap ASAP repeater chains. This was first noted in [21], where they investigated several

different methods of entanglement distributions for repeater chains. It would be of interest to see similar tools exploited for further variants of entanglement distribution.

To derive an analytical expression of $\mathbb{E}[\Lambda_n] = H_n$, it is convenient to define

$$H_n^t = \left(\frac{1-q}{q}\right)^n \sum_{\substack{\bar{t} \text{ s.t.} \\ t_n = t}} \left(q^{\sum_{i=1}^n t_i}\right) \cdot \left(\lambda^{\sum_{i=1}^{n-1} |t_i - t_{i+1}|}\right) . \tag{7}$$

This is convenient since

$$H_n^t = \frac{1-q}{q} \sum_{t'=1}^{\infty} q^t \lambda^{|t-t'|} H_{n-1}^{t'} . \tag{8}$$

That is, we can write H_n^t in terms of H_{n-1}^t, which in turn can be written in terms of H_{n-2}^t, etc. The approach is as follows: start with $H_1^t = \left(\frac{1-q}{q}\right) q^t$ (since there is only one link, there is only the q term), and then calculate H_n^t recursively using (8). Finally, we use that

$$H_n = \sum_{t=1}^{\infty} H_n^t , \tag{9}$$

to calculate the expectation value of the noise parameter Λ_n for n segments.

The recursion takes a particular simple form; we show in Appendix D that for $n \geq 1$, H_n^t is always a linear combination of terms of the form $q^{at}\lambda^{bt}$, where $a, b \in \mathbb{Z}$. We can thus think of H_n^t as an element of the vector space V consisting of all formal (real) linear combinations of elements of the form $q^{at}\lambda^{bt}$. Furthermore, we show that the recursion taking H_{n-1}^t to H_n^t is a linear map M from V to itself. This allows for a fast determination of H_n^t. Finally, the map (9) taking H_n^t to H_n is a linear form, which can also be conveniently calculated.

We note that the above procedure also works for inhomogeneous repeater chains, i.e. chains with varying entanglement generation success probabilities (due to for example non-equidistant nodes), and varying coherence parameters of the memories. This can be done by allowing the map M_i to vary at each step of the recursion, which we will discuss in more detail in a follow-up work.

While we have explicit analytical expressions for $\mathbb{E}[\Lambda_n]$ up to $n = 25$ segments, the expressions themselves are rather long. Instead of reporting them here, we provide all of our analytical expressions in Mathematica files [26].

3.1.1 Analytics for the global cut-off policy

Let $\mathbb{E}[\Lambda_n(T_c)]$ denote the average of Λ_n with a global cut-off parameter of T_c. To calculate $\mathbb{E}[\Lambda_n(T_c)]$, we sum over all instances that satisfy the global cut-off constraint, divided by the probability that entanglement was generated before the cut-off T_c was reached. That is, we calculate

$$\begin{aligned} &\mathbb{E}[\Lambda_n(T_c)] \\ &= \left(\frac{1-q}{q(1-q^{T_c})}\right)^n \sum_{\substack{\bar{t} \text{ s.t.} \\ \max(\bar{t}) \leq T_c}} \left(q^{\sum_i^n t_i}\right) \cdot \left(\lambda^{\sum_i^{n-1} |t_i - t_{i+1}|}\right) \\ &\equiv \bar{H}_n . \end{aligned}$$

We define $\bar{H}_n^t$ similar to the no cut-off case, i.e.

$$\bar{H}_n^t = \left(\frac{1-q}{q(1-q^{T_c})}\right)^n \sum_{\substack{\bar{t} \text{ s.t.} \\ t_n = t \\ t_i \leq T_c}} \left(q^{\sum_i^n t_i}\right) \cdot \left(\lambda^{\sum_i^{n-1} |t_i - t_{i+1}|}\right) .$$

Now, (8) and (9) generalize straightforwardly,

$$\bar{H}_n^t = \left(\frac{1-q}{q(1-q^{T_c})}\right) \sum_{t'=1}^{T_c} q^t \lambda^{|t-t'|} \bar{H}_{n-1}^{t'} , \tag{10}$$

$$\bar{H}_n = \sum_{t=1}^{T_c} \bar{H}_n^t . \tag{11}$$

In Appendix D we show that the desired properties from the case without a cut-off are preserved. That is, the maps $\bar{H}_n^t \mapsto \bar{H}_{n+1}^t$ and $\bar{H}_n^t \mapsto \bar{H}_n$ are best thought of as a linear map and as a linear form, allowing for the analytical determination of $\mathbb{E}[\Lambda_n(T_c)]$.

For the derivations in the appendices we drop the $\left(\frac{1-q}{q}\right)^n$ term, i.e. we define the quantity

$$Z_n = \sum_{\bar{t}} \left(q^{\sum_i^n t_i}\right) \cdot \left(\lambda^{\sum_i^{n-1} |t_i - t_{i+1}|}\right) . \tag{12}$$

Not only does this simplify some of the calculations, but the Z_n can also be interpreted as the *partition function* of the so-called solid-on-solid model [38, 39, 40, 41]. In follow-up work we exploit this connection, bringing to bear powerful techniques from statistical physics to

numerically calculate noise during (multipartite) entanglement distribution.

There are also other cut-off policies one could consider besides the global cut-off policy; we list several of them in Appendix A. A natural policy is the *local* cut-off policy, where one limits the time a single qubit is allowed to decohere. One is tempted to think that a similar recursive formulation is possible for the local cut-off policy as well. However, such a formulation requires that the sum in (10) starts at $\max(1, t - T_c)$, greatly complicating its calculation.

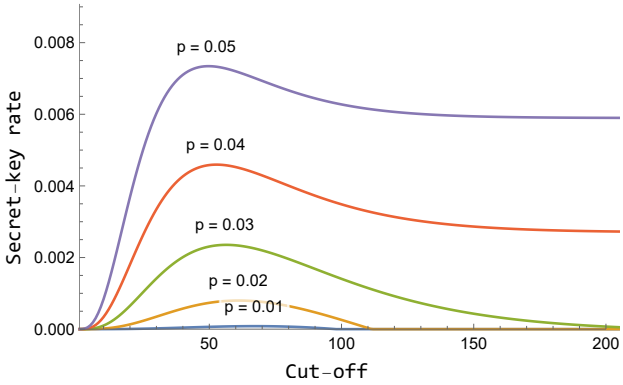

Figure 3: Secret-key rate with a global cut-off policy as a function of the cut-off parameter. Parameters used are $n = 5$ and $\lambda = 0.998$.

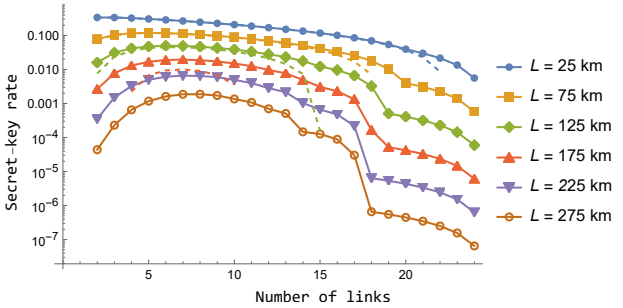

Figure 4: Secret-key rate optimized over the global cut-off parameter for several distances L, for $\lambda = 0.995$ and $\lambda_{\text{gen}} = 0.999$. The dashed lines correspond to the secret-key rate without any cut-off. The abrupt behavior corresponds to the optimal cut-off changing.

3.2 Parameter exploration

The secret-key rate is the rate at which one can generate secret bits between two parties. It is a convenient metric with a clear operational interpretation, which relies on both the (average) quality of the state and the (average) delivery time. In this section, we explore the effect that the cut-off has on the secret-key rate for several different parameter regimes. Appendix E contains details for the calculations of the secret-key rate. We note that for up to 8 repeater segments, the work from [21] includes an extensive numerical analysis of the secret-key rate not only for swap ASAP protocols, but several additional policies as well.

We start by fixing $n = 5$ segments and a decoherence parameter of $\lambda = 0.998$. In Fig. 3 we vary the success probability p and the cut-off. We find that for success probabilities of $p = 0.04 - 0.05$, the cut-off brings a modest increase (where the case of no cut-off corresponds to the cut-off going to infinity). For success probabilities smaller than $p = 0.03$ we find that a cut-off is necessary to be able to generate any amount of secret key [11, 12]. Observe that optimizing the cut-off is vital for maximizing secret-key rate — something that our methods now enable.

Fig. 4 illustrates how such an optimization can be used for parameter exploration. There, we show the secret-key rate optimized over the global cut-off parameter as a function of the number of segments for several total distances. The parameters used are $\lambda = 0.995$ and $\lambda_{\text{gen}} = 0.999$, where the last parameter corresponds to imperfect state generation.

4 Generating function approach

The methods from the previous section provide a recipe to calculate an analytical form of the average noise. However, the expressions grow unwieldy very fast, seemingly without any structure to them [21]. Thus, while the recursive calculation allows for an exact, quantitative understanding of the noise in swap ASAP repeater chains, it does not directly yield a qualitative understanding.

In this section we present a method for overcoming this, based on generating functions (not to be confused with the probability generating function from the previous section). This approach also yields a faster way to compute the analytical formulae found with previous methods. We first introduce generating functions, and how they the understanding of noise in repeater chains. Afterwards we use this approach to develop tight approximations and investigate the performance of repeater chains.

4.1 Outline of the approach

Generating functions are the main object of study in analytic combinatorics [42, 43], and can be used to extrapolate important information about sequences (a_n). This is done by associating a complex function $G(x)$ to the sequence (a_n) in the following manner,

$$G(x) = \sum_{n=0}^{\infty} a_n x^n \ , \tag{13}$$

where $a_n = \mathbb{E}[\Lambda_n]$ in our case of interest. For general sequences (a_n), one first find a closed-form expression for the associated generating function G. Given such a closed-form expression, it is possible to extract *complex analytic* behavior about G. In turn, this complex analytic behavior of G dictates the asymptotic behavior of $\mathbb{E}[\Lambda_n]$.

Let us make this more concrete. Let G be the generating function associated with a sequence (a_n), where $a_n > 0$. Let ρ be the singularity that is both real and closest to the origin (see Pringsheim's theorem [42, 44]). Furthermore, assume that ρ is a simple pole of G, which is always the case in this manuscript. Then

$$\lim_{n\to\infty} \frac{-\mathrm{Res}_\rho(G)\left(\frac{1}{\rho}\right)^{n+1}}{a_n} = 1 \ . \tag{14}$$

In other words, a_n is asymptotically equivalent to $-\mathrm{Res}_\rho(G)\left(\frac{1}{\rho}\right)^{n+1}$, where $\mathrm{Res}_\rho(G) = \lim_{x\to\rho}(x-\rho)G(x)$ is the residue of G at ρ. This last equality holds since ρ is a simple pole [45]. These claims follow from standard techniques in analytic combinatorics, in particular Theorem IV.10 from [42]. More specifically, this follows from expanding $f(z)$ in Theorem IV.10 around $z = \rho$ for the case of a simple pole, see Example IV.7. in [42] for an explicit demonstration. In Appendix B we give an example of this statement when the sequence (a_n) is given by the Fibonacci numbers. We show there how the generating function approach straightforwardly yields both asymptotic approximations and even exact expressions for the Fibonacci numbers.

Let us now return to the generating function associated to swap ASAP repeater chains, i.e. $G = \sum_{n=1}^{\infty} \mathbb{E}[\Lambda_n] x^n$. At this point, it is not clear whether a nice closed-form expression for G should exist, especially since we have argued that the individual coefficients $\mathbb{E}[\Lambda_n]$ become increasingly unwieldy as n grows. This is, however, a powerful feature of generating functions — they often behave much better than their coefficients considered individually [42, 43]. This good behavior is not only what allows us to extract information about the individual coefficients, but also allows us to find G in the first place.

Using the recursive formulation from the previous section, we show in Appendix F that

$$G = x\cdot\Phi_1 + x^2\left(\frac{(1-q)^2\lambda}{(\lambda-q)(1-\lambda q)}\right)\frac{\Phi_2\cdot\Phi_4}{1-x\cdot\Phi_3} \ ,$$

where $\Phi_i = \Phi_i(x) = \Phi_i(x,\lambda,q)$, $i = 1,2,3,4$ is a particular instance of a so-called q-hypergeometric series [46], and is a function of x, λ and q (see (124) to (129) for a full description). Since Φ_i is an entire function in x for fixed λ, q (with $\lambda \neq q$), the only singularity arises when $1 - x\cdot\Phi_3 = 0$. Although characterizing analytical solutions to equations involving (q-)hypergeometric series is generally a difficult endeavor [47, 48], it is straightforward to find solutions numerically. Let $x = \rho$ be the smallest real solution to $1-x\cdot\Phi_3 = 0$. Using the statement from Eq. (14), we find that

$$\begin{aligned}\mathbb{E}[\Lambda_n] &\sim [-\mathrm{Res}_\rho(G)]\cdot\left(\frac{1}{\rho}\right)^{n+1} \\ &= \left[\left(\frac{(1-q)^2\lambda\rho}{(\lambda-q)(1-\lambda q)}\right)\cdot\left(\frac{\Phi_2(\rho)\cdot\Phi_4(\rho)}{\Phi_3(\rho)+\rho\cdot\Phi_3'(\rho)}\right)\right]\cdot\left(\frac{1}{\rho}\right)^n \\ &\equiv [A(\lambda,q)]\cdot B(\lambda,q)^n \ ,\end{aligned} \tag{15}$$

where $\sim$ means that the ratio between the two sides converges to 1 as $n\to\infty$. The first equality follows from $\mathrm{Res}_\rho(f/g) = \lim_{x\to\rho}(x-\rho)\frac{f(x)}{g(x)} = \frac{f(\rho)}{g'(\rho)}$, which holds because of L'Hôpital's rule and the fact that ρ is a simple zero of g, i.e. $g(\rho) = 0$, while $g'(\rho) \neq 0$.

The approximation in (15) has a clear conceptual interpretation — the quality of the state decays exponentially in the number of segments n, where the decay rate is given by $B(\lambda,q)$ for each added segment. The $A(\lambda,q)$ term, on the other hand, can be thought of as a correction term. Note that both terms are independent of n. In the following subsection we numerically show that this approximation is not only tight in the asymptotic limit; it is already accurate enough for practical purposes for small n.

It is possible to approximate $\mathbb{E}[\Lambda_n]$ by including more than just the singularity closest to the origin, leading to a sum of exponential terms [42]. In practice, we find that taking the first singularity suffices.

All of the above generalizes straightforwardly to the case of a global cut-off, where there exist analogous functions $\overline{A}(\lambda, q, T_c)$ and $\overline{B}(\lambda, q, T_c)$ as well — see Appendices F and F.3 for the derivation of the associated generating function.

We conclude this subsection with two remarks. First, with the generating function G in hand, it is easier to find analytical expressions for $\mathbb{E}[\Lambda_n]$ than with the recursive relation from Section 3. This allows us to analytically calculate $\mathbb{E}[\Lambda_{25}]$ in a matter of seconds with Mathematica — something that was impossible with the recursive approach [26]. Second, we note that the function G is a *multivariate* generating function [43], where the associated combinatorial object can be interpreted as a type of lattice path L. The associated parameters q, λ and $x\left(\frac{1-q}{q}\right)$ count the area $A(L)$ under the path, the roughness $K(L)$ and the width $n(L)$ of the path, respectively. This result may be of independent interest to combinatorialists.

4.2 Parameter exploration

In this subsection we first explore the approximation found in (15), before using it to understand how λ and q affect $\mathbb{E}[\Lambda_n]$ qualitatively.

Fig. 5 shows the difference between the exact average noise and (15) for four sets of values of λ and q. We find that (15) provides an approximation that converges exponentially fast, independent of the parameters used.

From (15) and Fig. 5 we find that $\mathbb{E}[\Lambda_n]$ decays exponentially in the number of segments. In particular, $B(\lambda, q)$ can be thought of as a decay rate — every additional segment added reduces the average Λ parameter by a factor of $B(\lambda, q)$. The decay rate $B(\lambda, q)$ thus allows us to assign a metric to a given repeater chain for a fixed λ and q, *independent of the number of segments.*

As we show in Appendix F.3, a similar approximation works for the global cut-off as well. That is, we find a similar function for the decay parameter rate $\overline{B}(\lambda, q, T_c)$, and analyze how $\overline{B}(\lambda, q, T_c)$ varies as a function of λ and q in Fig. 6, for three values of the cut-off, $T_c = 2$, 6 and 10. We find that this tight approximation allows for a fast qualitative understanding of noise in swap ASAP repeater chains, which would have been impossible through a Monte Carlo simulation.

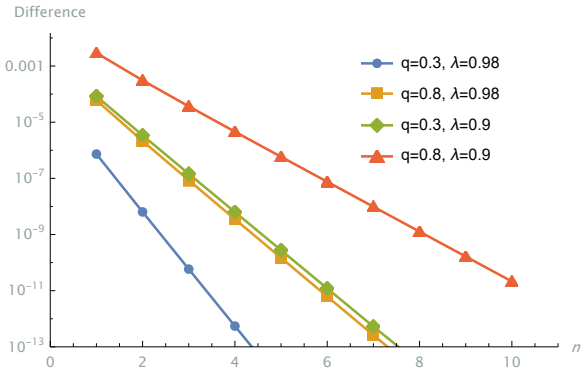

Figure 5: Difference between the approximation of Eq. (15) and the exact average noise parameter $\mathbb{E}[\Lambda_n]$ as a function of n, for four different sets of parameters.

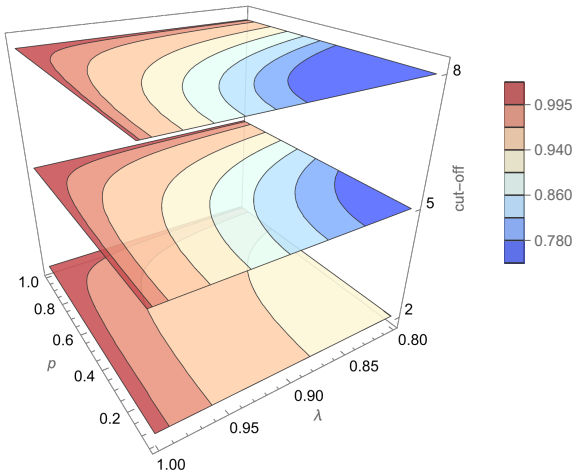

Figure 6: Several slices of the $\overline{B}(\lambda, q, T_c)$ function for $T_c = 2, 6, 10$. The $\overline{B}(\lambda, q, T_c)$ is a proxy for the decay rate of the quality of the state as additional segments get added.

5 Distribution of the noise

In this section we show that knowledge of $\mathbb{E}[\Lambda_n]$ in principle suffices to reconstruct the underlying probability distribution of the Λ_n parameter. We then apply our results to a so-called 'binning' approach to quantum key distribution.

5.1 Derivation

We will use here the found analytical forms of $\mathbb{E}[\Lambda_n]$ to find the actual distribution of the noise. As noted before in the paper and in [21], the key insight is that $\mathbb{E}[\Lambda_n]$ is the probability generating

function (not to be confused with the generating function from the previous section) of the random variable corresponding to the roughness K. That is,

$$\begin{aligned}\mathbb{E}\left[\Lambda_n\right] &= \left(\frac{1-q}{q}\right)^n \sum_{\bar{t}} \left(q^{\sum_i^n t_i}\right) \cdot \left(\lambda^{\sum_i^{n-1} |t_i - t_{i+1}|}\right) \\ &= \sum_{k=0}^{\infty} P\left(K=k\right) \lambda^k \\ &= \sum_{k=0} P\left(\Lambda_n = \lambda^k\right) \lambda^k \ .\end{aligned}$$

From this it follows that

$$k! \cdot P\left(\Lambda_n = \lambda^k\right) = \left(\mathbb{E}\left[\lambda_n\right]^{(k)}\right)\Big|_{\lambda=0} \ . \qquad (16)$$

Here $\mathbb{E}\left[\Lambda_n\right]^{(k)}$ is the k'th derivative of $\mathbb{E}\left[\Lambda_n\right]$ with respect to λ. Using Mathematica, $P\left(\Lambda_n = \lambda^k\right)$ can be calculated analytically for all k for up to $n = 10$ segments [26].

In Fig. 7, we validate our analytical results through Monte Carlo simulation. That is, we compare the exact distribution $P(K = k)$ of the roughness for 6 segments and $q = 0.6$ to an estimate of the distribution from a Monte Carlo simulation over 10^5 timesteps. We observe excellent agreement, and find convergence to the analytical distribution as we increase the number of runs. One interesting feature is the 'non-smooth' behavior of $P(K = k)$ (e.g. $P(K = 6) \geq P(K = 7)$, but $P(K = 7) \leq P(K = 8)$) which becomes more pronounced for larger p.

Noting that $P(K = k) = P(\Lambda_n = \lambda^k)$, we show in Fig. 8 the distribution of Λ_n for $p = 0.1$ and $\lambda = 0.995$ for n ranging from 2 to 10. Note that the 'non-smooth' behavior has disappeared since p is relatively small.

In principle a similar approach will work for the case with a global cut-off; however, due to the associated expressions becoming more complex, it was not possible to find analytical expressions of the k'th derivative.

5.2 Parameter exploration

As an application of the analytical expressions of the distribution on Λ_n, we consider secret key generation using a so-called binning method; see, for example, [49, 50]. That is, the collected classical data are separated into 'bins' according

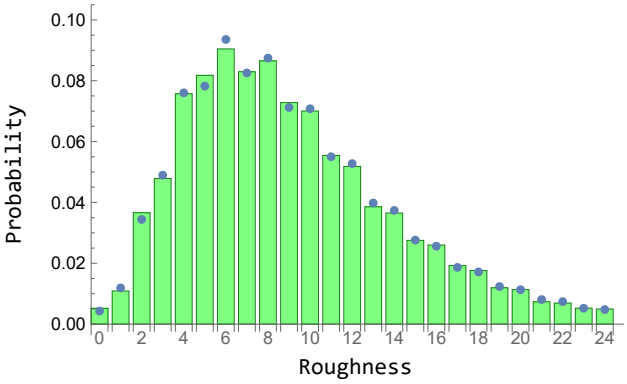

Figure 7: Verification of the analytical distribution of the roughness $K = \sum_{i=1}^{n-1} |t_i - t_{i+1}|$ for $q = 0.6$ and $n = 6$ segments (blue markers), along with an estimation found through a Monte Carlo simulation over 10^5 runs (green bars). The Monte Carlo simulation converges to the exact distribution as the number of runs increases.

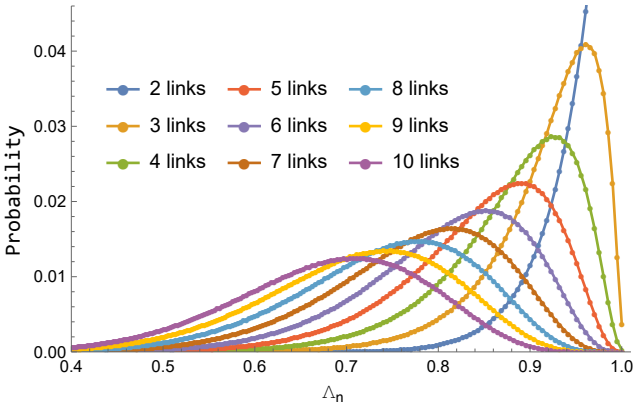

Figure 8: Distribution of the final Λ_n parameter for $q = 0.9$, $\lambda = 0.995$ for several number of segments.

to the available information about the associated generated state. Here, this available information will contain the roughness parameter $K(\bar{t})$, which describes the quality of the state. Note that the data need not be received at the time of measurement, which otherwise would come at the cost of increased decoherence.

How does this binning help? Let $\mathrm{SKF}\left(\Lambda\right)$ be the secret-key fraction, i.e. the fraction of bits that can be extracted from an asymptotic number of copies of a state with parameter Λ (see Appendix E for more details). Without binning, the secret-key fraction is given by $\mathrm{SKF}\left(\mathbb{E}\left[\Lambda\right]\right)$. With binning, the secret-key fraction becomes the weighted average. That is,

$$\begin{aligned}
&\sum_{k=0}^{\infty} P(\Lambda = \lambda^k) \cdot \mathrm{SKF}\left(\lambda^k\right) \qquad (17)\\
&= \sum_{k=0}^{k_{\max}} P(\Lambda = \lambda^k) \cdot \mathrm{SKF}\left(\lambda^k\right)\\
&\geq \mathrm{SKF}\left(\mathbb{E}\left[\Lambda\right]\right) \ ,
\end{aligned}$$

where in the first step we use the fact that the secret-key fraction is zero for $\Lambda \leq \Lambda_{\min} \approx 0.778$, such that $k_{\max} = \left\lceil \frac{\log(\Lambda_{\min})}{\log(\lambda)} \right\rceil$. The second step, relies on the fact that the secret-key fraction is convex as a function of Λ.

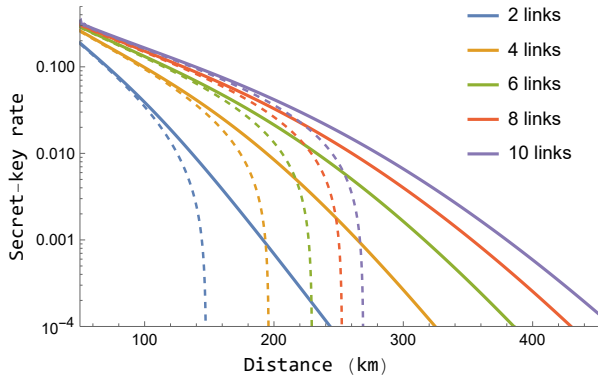

Figure 9: Secret-key rate with (solid) and without (dashed) a binning strategy (see main text) as a function of the distance, for several number of segments n and $\lambda = 0.99$.

Fig. 9 shows the secret-key rate with and without the binning method as a function of distance. The data shown correspond to $\lambda = 0.99$ and several values of n. We observe that the secret-key rate is non-zero for any arbitrary distance. This should not come as a surprise — the probability $P(K=0) = \sum_{t=1}^{\infty} \left(p\,(1-p)^{t-1}\right)^n = \frac{p^n}{1-(1-p)^n} \approx p^n$ is small but strictly positive for any $p > 0$.

6 Discussion

In this work we have characterized the noise in homogeneous swap ASAP repeater chains. We have done so by providing analytical formulae and tight approximations of the average fidelity (even when considering a global cut-off) along with the complete distribution of the noise, for a general class of noise models. The numerical evaluation of the formulae goes beyond existing work and reaches 25 repeater segments if one is interested in the average fidelity of the end-to-end link only, and to 10 repeater segments for a full characterization of the noise distribution. Since near-term quantum repeater experiments will most likely be based on swapping as soon as possible together with a cut-off policy, these results yield an understanding of what to expect for near-term networks. In particular, the optimization of the global cut-off (which would once be a costly Monte Carlo simulation) can now be done efficiently.

Due to the flexibility of the generating function approach, we anticipate that similar techniques as used here can be applied to other entanglement distribution problems, such as perhaps the swap ASAP protocol without idling (see 2), or the setup considered in [51].

In follow-up work, we investigate both the inhomogeneous bipartite setting and the inhomogeneous multipartite setting. This follow-up work hinges on interpreting (12) as the (classical) partition function of a particular physical system — the *solid-on-solid model* [38, 39, 40, 41]. By then exploiting techniques from statistical physics, we find numerically tight approximations on the average noise.

The generating function provides a powerful tool to understand noise in repeater chains, and it is natural to extend this to more complex scenarios. For example, is there a way to extend the generating function approach to incorporate additional noise on Alice's and Bob's memories? Using Jensen's inequality one can provide a lower bound on $\mathbb{E}\left[\Lambda_n\right]$ including the decoherence on Alice and Bob, but an analytical form for the associated generating function seems currently out of reach. Furthermore, other swapping schemes and cut-off policies besides the global cut-off policy have been analytically studied for small n, see [21]. The expressions found in [21] were rational functions in q and λ, similar to the swap ASAP scenario (with a global cut-off policy). It is thus natural to wonder whether these expressions can be obtained using the generating function approach as well.

Acknowledgements

We thank Guus Avis, Aliza Brinton, Patrick Emonts, Vicente Lenz, Filip Rozpędek, Matthew

Skrzypczyk, and Gayane Vardoyan for discussions and feedback on the manuscript. We also thank Mark C. Wilson for discussions on multivariate generating functions and their associated limit laws. Finally, we thank Philippe Flajolet and Robert Sedgewick for the free dissemination of their book.

This research was supported in part by the NSF grant CNS-1955744, NSF-ERC Center for Quantum Networks grant EEC-1941583, the MURI ARO Grant W911NF2110325. Tim Coopmans is supported by the Dutch National Growth Fund, as part of the Quantum Delta NL program.

References

[1] D. Gottesman, T. Jennewein, and S. Croke, "Longer-baseline telescopes using quantum repeaters," *Physical review letters*, vol. 109, no. 7, p. 070503, 2012. doi: 10.1103/PhysRevLett.109.070503 .

[2] A. Sajjad, M. R. Grace, and S. Guha, "Quantum limits of parameter estimation in long-baseline imaging," *Physical Review Research*, vol. 6, no. 1, p. 013212, 2024. doi: 10.1103/PhysRevResearch.6.013212 .

[3] M. Hillery, V. Bužek, and A. Berthiaume, "Quantum secret sharing," *Physical Review A*, vol. 59, no. 3, p. 1829, 1999. doi: 10.1103/PhysRevA.59.1829 .

[4] D. Markham and B. C. Sanders, "Graph states for quantum secret sharing," *Physical Review A*, vol. 78, no. 4, p. 042309, 2008. doi: 10.1103/PhysRevA.78.042309 .

[5] J. Cirac, A. Ekert, S. Huelga, and C. Macchiavello, "Distributed quantum computation over noisy channels," *Physical Review A*, vol. 59, no. 6, p. 4249, 1999. doi: 10.1103/PhysRevA.59.4249 .

[6] A. Serafini, S. Mancini, and S. Bose, "Distributed quantum computation via optical fibers," *Physical review letters*, vol. 96, no. 1, p. 010503, 2006. doi: 10.1103/PhysRevLett.96.010503 .

[7] S. Wehner, D. Elkouss, and R. Hanson, "Quantum internet: A vision for the road ahead," *Science*, vol. 362, no. 6412, 2018. doi: 10.1126/science.aam9288 .

[8] J. Rabbie, K. Chakraborty, G. Avis, and S. Wehner, "Designing quantum networks using preexisting infrastructure," *npj Quantum Information*, vol. 8, no. 1, p. 5, 2022. doi: 10.1038/s41534-021-00501-3 .

[9] H.-J. Briegel, W. Dür, J. I. Cirac, and P. Zoller, "Quantum repeaters: the role of imperfect local operations in quantum communication," *Physical Review Letters*, vol. 81, no. 26, p. 5932, 1998. doi: 10.1103/PhysRevLett.81.5932 .

[10] K. Azuma, S. E. Economou, D. Elkouss, P. Hilaire, L. Jiang, H.-K. Lo, and I. Tzitrin, "Quantum repeaters: From quantum networks to the quantum internet," *arXiv preprint arXiv:2212.10820*, 2022. doi: 10.1103/RevModPhys.95.045006 .

[11] F. Rozpędek, K. Goodenough, J. Ribeiro, N. Kalb, V. C. Vivoli, A. Reiserer, R. Hanson, S. Wehner, and D. Elkouss, "Parameter regimes for a single sequential quantum repeater," *Quantum Science and Technology*, vol. 3, no. 3, p. 034002, 2018.

[12] F. Rozpędek, R. Yehia, K. Goodenough, M. Ruf, P. C. Humphreys, R. Hanson, S. Wehner, and D. Elkouss, "Near-term quantum-repeater experiments with nitrogen-vacancy centers: Overcoming the limitations of direct transmission," *Physical Review A*, vol. 99, no. 5, p. 052330, 2019.

[13] O. Collins, S. Jenkins, A. Kuzmich, and T. Kennedy, "Multiplexed memory-insensitive quantum repeaters," *Physical review letters*, vol. 98, no. 6, p. 060502, 2007. doi: 10.1103/PhysRevLett.98.060502 .

[14] G. Avis, F. F. da Silva, T. Coopmans, A. Dahlberg, H. Jirovská, D. Maier, J. Rabbie, A. Torres-Knoop, and S. Wehner, "Requirements for a processing-node quantum repeater on a real-world fiber grid," *arXiv preprint arXiv:2207.10579*, 2022. doi: 10.1038/s41534-023-00765-x .

[15] B. Li, T. Coopmans, and D. Elkouss, "Efficient optimization of cut-offs in quantum repeater chains," in *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 2020, pp. 158–168. doi: 10.1109/QCE49297.2020.00029 .

[16] S. Khatri, C. T. Matyas, A. U. Siddiqui, and J. P. Dowling, "Practical figures of merit and thresholds for entanglement distribution in quantum networks," *Phys. Rev. Research*, vol. 1, p. 023032, Sep 2019. doi: 10.1103/PhysRevResearch.1.023032 . [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevResearch.1.023032

[17] S. Santra, L. Jiang, and V. S. Malinovsky, "Quantum repeater architecture with hierarchically optimized memory buffer times," *Quantum Science and Technology*, vol. 4, no. 2, p. 025010, mar 2019. doi: 10.1088/2058-9565/ab0bc2 . [Online]. Available: https://doi.org/10.1088%2F2058-9565%2Fab0bc2

[18] B. Davies, T. Beauchamp, G. Vardoyan, and S. Wehner, "Tools for the analysis of quantum protocols requiring state generation within a time window," *arXiv:2304.12673*, 2023. doi: 10.1109/TQE.2024.3358674 .

[19] K. Azuma, S. Bäuml, T. Coopmans, D. Elkouss, and B. Li, "Tools for quantum network design," *AVS Quantum Science*, vol. 3, no. 1, p. 014101, 2021. doi: 10.1116/5.0024062 . [Online]. Available: https://doi.org/10.1116/5.0024062

[20] L. Praxmeyer, "Reposition time in probabilistic imperfect memories," *arXiv preprint arXiv:1309.3407*, 2013.

[21] L. Kamin, E. Shchukin, F. Schmidt, and P. van Loock, "Exact rate analysis for quantum repeaters with imperfect memories and entanglement swapping as soon as possible," *Physical Review Research*, vol. 5, no. 2, p. 023086, 2023. doi: 10.1103/PhysRevResearch.5.023086 .

[22] E. Shchukin, F. Schmidt, and P. van Loock, "Waiting time in quantum repeaters with probabilistic entanglement swapping," *Physical Review A*, vol. 100, no. 3, p. 032322, 2019. doi: 10.1103/PhysRevA.100.032322 .

[23] S. D. Reiß and P. van Loock, "Deep reinforcement learning for key distribution based on quantum repeaters," *Phys. Rev. A*, vol. 108, p. 012406, Jul 2023. doi: 10.1103/PhysRevA.108.012406 . [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevA.108.012406

[24] A. Zang, X. Chen, A. Kolar, J. Chung, M. Suchara, T. Zhong, and R. Kettimuthu, "Entanglement distribution in quantum repeater with purification and optimized buffer time," in *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2023, pp. 1–6. doi: 10.1109/INFOCOMWKSHPS57453.2023.10226122 .

[25] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Theoretical Computer Science*, vol. 560, pp. 7–11, 2014. doi: https://doi.org/10.1016/j.tcs.2014.05.025 ., theoretical Aspects of Quantum Cryptography – celebrating 30 years of BB84. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304397514004241

[26] K. Goodenough, "Swap asap analytics," https://github.com/KDGoodenough/swapASAPAnalytics, 2024.

[27] M. Pant, H. Krovi, D. Towsley, L. Tassiulas, L. Jiang, P. Basu, D. Englund, and S. Guha, "Routing entanglement in the quantum internet," *npj Quantum Information*, vol. 5, no. 1, p. 25, 2019. doi: 10.1038/s41534-019-0139-x .

[28] P. Fittipaldi, A. Giovanidis, and F. Grosshans, "A linear algebraic framework for quantum internet dynamic scheduling," in *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 2022, pp. 447–453. doi: 10.1109/QCE53715.2022.00066 .

[29] E. A. Van Milligen, E. Jacobson, A. Patil, G. Vardoyan, D. Towsley, and S. Guha, "Entanglement routing over networks with time multiplexed repeaters," *arXiv preprint arXiv:2308.15028*, 2023.

[30] E. A. Van Milligen, C. N. Gagatsos, E. Kaur, D. Towsley, and S. Guha, "Utilizing probabilistic entanglement between sensors in quantum networks," *Physical Review Applied*, vol. 22, no. 6, p. 064085, 2024. doi: 10.1103/PhysRevApplied.22.064085 .

[31] G. Vardoyan, S. Guha, P. Nain, and D. Towsley, "On the stochastic analysis of a quantum entanglement switch," *ACM*

SIGMETRICS Performance Evaluation Review, vol. 47, no. 2, pp. 27–29, 2019. doi: 10.1145/3374888.3374899 .

[32] K. Noh and C. Chamberland, "Fault-tolerant bosonic quantum error correction with the surface–gottesman-kitaev-preskill code," *Physical Review A*, vol. 101, no. 1, p. 012316, 2020. doi: 10.1103/PhysRevA.101.012316 .

[33] V. V. Albert, K. Noh, K. Duivenvoorden, D. J. Young, R. Brierley, P. Reinhold, C. Vuillot, L. Li, C. Shen, S. M. Girvin *et al.*, "Performance and structure of single-mode bosonic codes," *Physical Review A*, vol. 97, no. 3, p. 032346, 2018. doi: 10.1103/PhysRevA.97.032346 .

[34] M. H. Shaw, A. C. Doherty, and A. L. Grimsmo, "Logical gates and read-out of superconducting gottesman-kitaev-preskill qubits," *arXiv preprint arXiv:2403.02396*, 2024.

[35] K. Noh, V. V. Albert, and L. Jiang, "Quantum capacity bounds of gaussian thermal loss channels and achievable rates with gottesman-kitaev-preskill codes," *IEEE Transactions on Information Theory*, vol. 65, no. 4, pp. 2563–2582, 2018. doi: 10.1109/TIT.2018.2873764 .

[36] S. Haldar, P. Barge, X. Cheng, K.-C. Chang, B. Kirby, S. Khatri, C. W. Wong, and H. Lee, "Reducing classical communication costs in multiplexed quantum repeaters using hardware-aware quasi-local policies," 2024. doi: 10.1038/s42005-025-02029-w .

[37] J. Li, T. Coopmans, P. Emonts, K. Goodenough, J. Tura, and E. van Nieuwenburg, "Optimising entanglement distribution policies under classical communication constraints assisted by reinforcement learning," *arXiv preprint arXiv:2412.06938*, 2024.

[38] A. L. Owczarek and T. Prellberg, "Exact solution of the discrete (1+ 1)-dimensional sos model with field and surface interactions," *journal of Statistical Physics*, vol. 70, pp. 1175–1194, 1993. doi: 10.1007/BF01049427 .

[39] B. Rózycki and M. Napiórkowski, "The rsos model for a slit with different walls," *Journal of Physics A: Mathematical and General*, vol. 36, no. 16, p. 4551, 2003.

[40] A. Owczarek and T. Prellberg, "Exact solution of the discrete (1+ 1)-dimensional rsos model with field and surface interactions," *Journal of Physics A: Mathematical and Theoretical*, vol. 42, no. 49, p. 495003, 2009. doi: 10.1088/1751-8113/42/49/495003 .

[41] A. L. Owczarek and T. Prellberg, "Exact solution of the discrete (1+ 1)-dimensional rsos model in a slit with field and wall interactions," *Journal of Physics A: Mathematical and Theoretical*, vol. 43, no. 37, p. 375004, 2010. doi: 10.1088/1751-8113/43/37/375004 .

[42] P. Flajolet and R. Sedgewick, *Analytic combinatorics*. cambridge University press, 2009. doi: 10.1017/CBO9780511801655 .

[43] R. Pemantle and M. C. Wilson, *Analytic combinatorics in several variables*. Cambridge University Press, 2013, no. 140. doi: 10.1017/9781108874144 .

[44] R. Remmert, *Theory of complex functions*. Springer Science & Business Media, 1991, vol. 122. doi: 10.1007/978-1-4612-0939-3 .

[45] S. Lang, *Complex analysis*. Springer Science & Business Media, 2013, vol. 103. doi: 10.1007/978-3-642-59273-7 .

[46] G. Gasper and M. Rahman, *Basic hypergeometric series*. Cambridge university press, 2004, vol. 96. doi: 10.1017/CBO9780511526251.004 .

[47] K. Driver and K. Jordaan, "Zeros of the hypergeometric polynomial f (-n, b; c; z)," *arXiv preprint arXiv:0812.0708*, 2008.

[48] D. Dominici, S. J. Johnston, and K. Jordaan, "Real zeros of 2f1 hypergeometric polynomials," *Journal of Computational and Applied Mathematics*, vol. 247, pp. 152–161, 2013.

[49] Y. Jing, D. Alsina, and M. Razavi, "Quantum key distribution over quantum repeaters with encoding: Using error detection as an effective postselection tool," *Physical Review Applied*, vol. 14, no. 6, p. 064037, 2020. doi: 10.1103/PhysRevApplied.14.064037 .

[50] K. Goodenough, S. de Bone, V. L. Addala, S. Krastanov, S. Jansen, D. Gijswijt, and D. Elkouss, "Near-term n to k distillation protocols using graph codes," *arXiv preprint arXiv:2303.11465*, 2023. doi: 10.1109/JSAC.2024.3380094 .

[51] G. Avis, F. Rozpędek, and S. Wehner, "Analysis of multipartite entanglement distribution using a central quantum-network node," *Physical Review A*, vol. 107, no. 1, p. 012609, 2023. doi: 10.1103/PhysRevA.107.012609 .

[52] Á. G. Iñesta, G. Vardoyan, L. Scavuzzo, and S. Wehner, "Optimal entanglement distribution policies in homogeneous repeater chains with cutoffs," *npj Quantum Information*, vol. 9, no. 1, p. 46, 2023.

[53] J. Kepler, *Strena seu de Niue sexangula*. Gottfried Tampach, 1966.

[54] C. Schneer, "Kepler's new year's gift of a snowflake," *Isis*, vol. 51, no. 4, pp. 531–545, 1960. doi: 10.1086/349411 .

[55] R. P. Stanley, "Enumerative combinatorics volume 1 second edition," *Cambridge studies in advanced mathematics*, 2011.

[56] F. Shahbeigi, D. Amaro-Alcalá, Z. Puchała, and K. Życzkowski, "Log-convex set of lindblad semigroups acting on n-level system," *Journal of Mathematical Physics*, vol. 62, no. 7, p. 072105, 2021. doi: 10.1063/5.0009745 .

[57] H. Weyl, "Quantenmechanik und gruppentheorie," *Zeitschrift für Physik*, vol. 46, no. 1-2, pp. 1–46, 1927. doi: 10.1007/BF02055756 .

[58] J. Helsen, I. Roth, E. Onorati, A. H. Werner, and J. Eisert, "General framework for randomized benchmarking," *PRX Quantum*, vol. 3, no. 2, p. 020357, 2022. doi: 10.1103/PRXQuantum.3.020357 .

[59] I. M. Isaacs, *Character theory of finite groups*. American Mathematical Soc., 2006, vol. 359. doi: 10.1090/chel/359 .

[60] A. Terras, *Fourier analysis on finite groups and applications*. Cambridge University Press, 1999. doi: 10.1017/CBO9780511626265 .

[61] P. Diaconis, "Group representations in probability and statistics," *Lecture notes-monograph series*, vol. 11, pp. i–192, 1988. doi: 10.1214/lnms/1215467407 .

[62] M. M. Wilde, *Quantum information theory*. Cambridge University Press, 2013. doi: 10.1017/9781316809976 .

[63] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, and M. Peev, "The security of practical quantum key distribution," *Reviews of modern physics*, vol. 81, no. 3, p. 1301, 2009. doi: 10.1103/RevModPhys.81.1301 .

[64] N. K. Bernardes, L. Praxmeyer, and P. van Loock, "Rate analysis for a hybrid quantum repeater," *Physical Review A*, vol. 83, no. 1, p. 012323, 2011. doi: 10.1103/PhysRevA.83.012323 .

A Other cut-off policies

For completeness, we mention here several other cut-off policies, but which will not be considered in this paper. As previously mentioned, decoherence is a prime obstacle to quantum communication. A cut-off policy is a condition for discarding the present entanglement. Discarding the generated entanglement will reduce the rate at which entanglement is delivered, but will mitigate the time spent on distributing end-to-end entanglement which is of too low quality. A cut-off policy thus allows for a trade-off between the rate and the quality of the delivered end-to-end entanglement. Cut-off policies and their trade-offs have been studied in for example [11, 12, 13, 14, 15, 21, 22, 23], where they were shown to be indispensable for near-term devices.

The *local cut-off policy* limits the time a single memory is allowed to store entanglement for. As soon as a memory not belonging to Alice or Bob holds entanglement, a timer is started. As before, this timer increases by 1 every round, and is reset to 0 after discarding. A reset occurs when any timer exceeds T_c, where we use the same notation for the global and local cut-off. In Fig. 10 we show an example of the local cut-off. Here, the left memory of the fourth link had to store entanglement for more than $T_c = 2$ rounds.

The policy considered in [20] is similar to the global cut-off policy in this paper. However, instead of starting a timer at the beginning of all attempts, a timer is started at the time the first link succeeds.

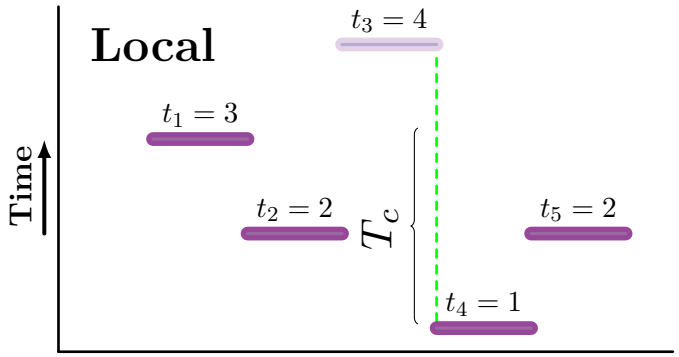

Figure 10: Graphical description of the local cut-off policy. Each node has agreed on a parameter T_c. For the local cut-off, a reset is performed when any node has to store entanglement for more than T_c rounds. We consider the local cut-off in follow-up work.

Another policy would be based on predicting the delivery time (see [52]) and quality of the delivered state, conditioned on the current state of the repeater chain [52]. Note that *all* entanglement is reset after a cut-off condition has been reached in the policy of [20] and both the global and local cut-off.

We have numerically observed through Monte Carlo simulations that the local cut-off performs slightly better than the global cut-off, but not by a significant amount.

B Example of the Fibonacci generating function

We motivate and explain here generating functions through a simple and more familiar example, before applying generating functions to swap ASAP repeater chains. In this example we will estimate the asymptotic behavior of the Fibonacci numbers F_n by first finding the associated generating function $G(x)$ and studying its complex-analytic behavior. To wit, the Fibonacci numbers F_n are defined as $F_0 = 0$, $F_1 = 1$, and $F_n = F_{n-1} + F_{n-2}$, $n \geq 2$. We now show a well-known derivation of the Fibonacci generating function $G(x) = \sum_{n=0}^{\infty} F_n x^n$. Through a judiciously chosen starting expression we find

$$\begin{aligned}
&\left(1 - x - x^2\right) G(x) \\
&= \left(1 - x - x^2\right) \sum_{n=0}^{\infty} F_n x^n \\
&= \sum_{n=0}^{\infty} F_n x^n - \sum_{n=0}^{\infty} F_n x^{n+1} - \sum_{n=0}^{\infty} F_n x^{n+2} \\
&= \sum_{n=0}^{\infty} F_n x^n - \sum_{n=1}^{\infty} F_{n-1} x^n - \sum_{n=2}^{\infty} F_{n-2} x^n \\
&= F_0 + (F_1 - F_0)x \\
&= x \quad \implies G(x) = \frac{x}{1 - x - x^2} \,,
\end{aligned} \tag{18}$$

where we canceled all like terms going from the fourth to the fifth line.

Note that the infinite sum $\sum F_n x^n$ does not converge for all $x \in \mathbb{C}$. Contrast this with $G(x) = \frac{x}{1-x-x^2}$, which is defined for all $x \in \mathbb{C}$, save for the two roots ρ_1, ρ_2 of $1 - x - x^2$. The function $G(x)$ is an example of an analytic continuation of $\sum F_n x^n$, meaning that the original domain can be uniquely extended in a well-defined manner [45]. Such machinery allows us to conveniently ignore issues of convergence. More importantly, under mild conditions a generating function $G(x)$ will be a *meromorphic function*. This means that, save for a set of isolated points $\{\rho_i\}$, $G(x)$ is complex-differentiable on $\mathbb{C}$.

The $\{\rho_i\}$ are called *singularities* of a meromorphic function, and reveal key information about the underlying sequence a_n. For example, a standard result in analytic combinatorics (see theorem IV.10 from [42]) tells us that the ratio between a_n and $\left(\frac{1}{\rho}\right)^{n+1}$ converges to a constant c, where ρ is the singularity closest to the origin[4]. In other words, a_n is asymptotically equal to $c \cdot \left(\frac{1}{\rho}\right)^{n+1}$. Applying this to the Fibonacci generating function, we find that ρ is given by the smallest solution to $1-x-x^2$, i.e. $\rho = -\phi' = \frac{1}{\phi} = \frac{\sqrt{5}-1}{2} = 0.618\ldots$ with ϕ the golden ratio and ϕ' its conjugate. In other words, the limit of the ratio between F_n and $\left(\frac{1}{\rho}\right)^{n+1} = \phi^{n+1}$ converges to some constant. This immediately reclaims the classic result by Kepler that $\lim_{n\to\infty} \frac{F_{n+1}}{F_n} = \phi$ [53, 54].

The location of the closest singularity ρ dictates the exponential growth/decay, but not the constant c. Fortunately, the constant c depends on the closest singularity as well, see theorem IV.10 of [42]. Specifically, c is given by minus the *residue* of the function G at ρ. As noted in the main text, this can be derived more explicitly by expanding $f(z)$ in Theorem IV.10 of [42] around $z = \rho$ when ρ is a simple pole. An explicit demonstration can be found in Example IV.7. of [42]. For the cases of interest to this manuscript (i.e. where all singularities are simple poles), we have that $\mathrm{Res}_{\rho_i}(G) = \lim_{x\to\rho_i}(x-\rho_i)\,G(x)$ [45]. A straightforward calculation then shows that the residue of G at ρ is given by $-\frac{1}{\sqrt{5}\phi}$, from which we find that

$$F_n \sim [-\mathrm{Res}_\rho(G)] \cdot \left(\frac{1}{\rho}\right)^{n+1} \tag{19}$$

$$= \frac{\phi^n}{\sqrt{5}}\ . \tag{20}$$

Indeed, $F_{10} = 55$, while the approximation in Eq. (20) yields the excellent agreement of $55.0036\ldots$. Theorem IV.10 of [42] tells us that the approximation can be further refined by including the other singularity ρ'. In fact, doing so would recover a closed-form expression of the Fibonacci numbers, known as Binet's formula [42].

To further highlight the insight generating functions provide, we conclude with one more fact that can be gleaned from G. Formally rewriting we have

$$G(x) = \frac{x}{1-x-x^2} \tag{21}$$

$$= x \cdot \frac{1}{1-(x+x^2)} \tag{22}$$

$$= x \cdot \left((x+x^2)^0 + (x+x^2)^1 + (x+x^2)^2 + \ldots\right) \tag{23}$$

$$\equiv \sum_{n=0}^{\infty} F_n x^n\ . \tag{24}$$

Expanding the penultimate line, we find that F_n is the number of ways all the different terms $(x+x^2)^i$ contribute a x^{n-1} term. In other words, this provides a purely algebraic proof of the classic fact that F_n is the number of ordered sums of 1 and 2 adding up to $n-1$ [55], e.g. $F_5 = 5$ since

$$4 = 1+1+1+1 \tag{25}$$

$$= 1+1+2 = 1+2+1 \tag{26}$$

$$= 2+1+1 = 2+2\ , \tag{27}$$

[4]For technical reasons the singularities are implicitly required to be *simple poles*, see [45]. This can be easily verified to be the case for the cases considered here.

and $F_6 = 8$ since

$$\begin{aligned} 5 &= 1+1+1+1+1 = 1+1+1+2 && (28)\\ &= 1+1+2+1 = 1+2+1+1 && (29)\\ &= 2+1+1+1 = 1+2+2 && (30)\\ &= 2+1+2 = 2+2+1\ . && (31) \end{aligned}$$

C Generalization to X-symmetric qudit Pauli noise

Our analysis in the main text is based on the depolarizing channel as noise model for memory decoherence (see Section 2). There were two key features of this noise model that made it amenable for analysis:

1. *The key parameter multiplies when composing two noise channels.* The depolarizing noise model from Section 2 is parameterized by two values: the probability of preserving the state perfectly (i.e. λ) and the probability of transforming the state into the maximally mixed state. Due to normalization, this last probability is just $1-\lambda$. Applying a noise map with a given λ_{noise} corresponds to the transformations $\lambda \to \lambda \cdot \lambda_{\text{noise}}$ and $1-\lambda \to 1-\lambda\cdot\lambda_{\text{noise}}$. The fact that the first map corresponds to multiplication was a key part of our analysis.

2. *The key parameter multiplies when performing the entanglement swap.* Next, we used the fact that performing a swap between two states described by noise parameters λ and λ', respectively, leads to a new state with noise parameter $\lambda\cdot\lambda'$.

In this Appendix, we will consider the more general scenario where the nodes hold two-*qudit* entangled states. We will show for a large class of qudit channels that we call X-symmetric channels, how to find a parameterization into parameters $\vec{\lambda}$ such that the two above properties still hold. To be precise, an X-symmetric channel $\mathcal{N}$ is a d-dimensional qudit Pauli channel (definition below in Section C.1) for which $p_{X^aZ^b} = p_{X^{-a}Z^b}$ for all $a, b \in \{0, 1, \ldots, d-1\}$. We will give a procedure how to find a parameterization $\mathcal{N} = \mathcal{N}_{\vec{\lambda}}$ into at most d^2 parameters $\vec{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_{d^n})$, such that both composition (item 1 below) and entanglement swapping (item 2) are represented by pointwise multiplication of the entries of $\vec{\lambda}$:

1. for any two $\vec{\lambda}, \vec{\lambda}'$, we have $\mathcal{N}_{\vec{\lambda}} \circ \mathcal{N}_{\vec{\lambda}'} = \mathcal{N}_{\vec{\lambda}\cdot\vec{\lambda}'}$.

2. let $\psi_d = |\psi_d\rangle\langle\psi_d|$ be a maximally entangled state on two d-dimensional qudits. Then for any two $\vec{\lambda}, \vec{\lambda}'$, an entanglement swap on states $(\mathbb{I}\otimes\mathcal{N}_{\vec{\lambda}})[\psi]$ and $(\mathbb{I}\otimes\mathcal{N}_{\vec{\lambda}'})[\psi_d]$ yields the state $(\mathbb{I}\otimes\mathcal{N}_{\vec{\lambda}\cdot\vec{\lambda}'})[\psi_d]$.

where $\vec{\lambda}\cdot\vec{\lambda}' = (\lambda_1\lambda_1', \ldots, \lambda_{d^2}\lambda_{d^2}')$ represents entrywise multiplication and $\mathbb{I}$ is the d-dimensional qudit identity channel.

Consequently, the analysis in the main text is straightforwardly extendible to qudit links as long as the noise model is an X-symmetric channel. The X-symmetric channels form a subset of qudit Pauli channels, and include (biased) depolarizing noise as well as *any* qubit Pauli channel (i.e. Pauli channels for $d = 2$).

In what follows we first recall the definition of qudit Pauli channels. We then study composition of qudit Pauli channels, and show that qudit Pauli channels admit a parameterization so that channel composition is represented by multiplication of the parameters. Finally, we investigate how maximally entangled states that have undergone qudit Pauli channels are transformed under swap operations. From the latter, we infer that in general, the entanglement swap is not represented by parameter multiplication, but for X-symmetric channels, it is.

C.1 Qudit Pauli channels

A qudit Pauli channel is any channel $\mathcal{N}$ corresponding to a probabilistic mixture of operators from the set $\mathcal{U} = \{X^a Z^b\}_{a,b=0}^{d-1}$ [56], i.e.

$$\mathcal{N}(\cdot) = \sum_{U \in \mathcal{U}} p_U U(\cdot) \ . \tag{32}$$

Here $0 \leq a \leq d-1$, $0 \leq b \leq d-1$ [56], and X and Z are generalized Pauli operators defined by their action on the standard basis of $\mathbb{C}^d$,

$$X|j\rangle = |j+1 \mod d\rangle \ , \tag{33}$$
$$Z|j\rangle = \omega^j |j\rangle \ , \tag{34}$$

where $\omega = e^{\frac{2\pi i}{d}}$ [57].

Note that $X^a Z^b$ and $X^k Z^l$ commute up to a phase. Since $ABA^\dagger = \left(e^{i\theta}A\right) B \left(e^{i\theta}A\right)^\dagger$ for any angle $\theta \in [0, 2\pi)$, we have that $\left(X^a Z^b\right)\left(X^k Z^l\right)\rho\left(\left(X^a Z^b\right)\left(X^k Z^l\right)\right)^\dagger = \left(X^k Z^l\right)\left(X^a Z^b\right)\rho\left(\left(X^k Z^l\right)\left(X^a Z^b\right)\right)^\dagger$; qudit Pauli channels commute. Now equip the set $\mathcal{U}$ with a group structure, where the group operation is given by matrix multiplication but where phases are ignored. Formally, this is the group $\langle X, Z\rangle \Big/ \langle \omega I\rangle$, of which $\{X^a Z^b\}_{a,b=0}^{d-1}$ forms a transversal of the underlying equivalence relation. This group is isomorphic to $C_d \times C_d$, the direct product of the cyclic group of order d with itself.

C.2 Mapping composition of qudit Pauli channels to multiplication

We first investigate the composition of two qudit Pauli channels, where in particular we are interested in reducing the composition to multiplication of real numbers, i.e. the analogues of the λ parameter for qubit depolarizing noise. The composition of two channels $\mathcal{N}_1, \mathcal{N}_2$ with distributions $p_U \equiv f(U)$ and $p_U \equiv g(U)$, respectively, can be written as

$$(\mathcal{N}_2 \circ \mathcal{N}_1)[\cdot] = \sum_{U \in \mathcal{U}} g(U) U \left[\sum_{V \in \mathcal{U}} f(V) V[\cdot]\right] \tag{35}$$
$$= \sum_{W \in \mathcal{U}} \left(\sum_{UV=W} g(U) f(V)\right) W[\cdot] \ . \tag{36}$$

We recognize eq. (36) as a convolution over finite Abelian groups[5]. The Fourier transform of a function $h : \mathcal{G} \to \mathbb{C}$, where $\mathcal{G}$ is a finite Abelian group, is defined as the function $\hat{h} : \hat{\mathcal{G}} \to \mathbb{C}$ such that

$$\hat{h}(\chi) = \sum_{U \in \mathcal{G}} h(U) \chi(U)^* \ .$$

The inverse Fourier transform is given by

$$h(U) = \frac{1}{|\mathcal{G}|} \sum_{\chi \in \hat{\mathcal{G}}} \hat{h}(\chi) \chi(U) \ . \tag{37}$$

[5]Fourier transforms on (non-Abelian) groups have been applied to quantum information before, e.g. in randomized benchmarking [58].

Here, the χ are also called the (unitary) characters of $\mathcal{G}$, i.e. the distinct homomorphisms from $\mathcal{G}$ to the group of unit complex numbers $\mathbb{T} = \{z \in \mathbb{C} \mid |z| = 1\}$. That is, the set of all functions $\chi : \mathcal{G} \to \mathbb{T}$ such that $\chi(U)\chi(V) = \chi(UV)$. We denote by $\hat{\mathcal{G}}$ the set of all characters of $\mathcal{G}$.

The procedure to find the desired parameterization now is as follows. First, we recall that the coefficients of the channel are specified through the noise coefficient map f, defined as $f(U) = p_U$, where U is an element of the phaseless qudit Pauli group $\mathcal{U}$. The first step is to compute the Fourier transform of f. The second step is to determine the characters χ of $\mathcal{U}$. These are the same as the characters of $C_d \times C_d$, to which the phaseless Pauli group $\mathcal{U}$ is isomorphic; in turn, we can derive the characters of $C_d \times C_d$ from the characters of C_d [59]. Finally, we evaluate the Fourier transform $\hat{f}$ at the characters χ. We will choose the resulting $\hat{f}(\chi)$ as our new parameters $\lambda_1, \lambda_2, \ldots$, as convolution of two functions becomes pointwise multiplication of their Fourier transforms [60, 61].

As special case, we consider $d = 2$, i.e. we consider an arbitrary qubit Pauli channel. In that case, $\mathcal{U}$ is isomorphic to $C_2 \times C_2$ (also known as the Klein four-group), which has characters

$$\begin{aligned}
&\chi^1(I) = 1, \quad \chi^1(X) = 1, \quad \chi^1(Y) = 1, \quad \chi^1(Z) = 1\ , \\
&\chi^2(I) = 1, \quad \chi^2(X) = 1, \quad \chi^2(Y) = -1, \quad \chi^2(Z) = -1\ , \\
&\chi^3(I) = 1, \quad \chi^3(X) = -1, \quad \chi^3(Y) = 1, \quad \chi^3(Z) = -1\ , \\
&\chi^4(I) = 1, \quad \chi^4(X) = -1, \quad \chi^4(Y) = -1, \quad \chi^4(Z) = 1\ .
\end{aligned}$$

From normalization of the probability, we see that λ_1 is always equal to 1. The remaining parameters are

$$\lambda_2 = p_I + p_X - p_Y - p_Z\ , \tag{38}$$
$$\lambda_3 = p_I - p_X + p_Y - p_Z\ , \tag{39}$$
$$\lambda_4 = p_I - p_X - p_Y + p_Z \tag{40}$$

which is indeed the parameterization given in the main text in Section 2.

Let us specialize to the case of depolarizing noise, where $p_X = p_Y = p_Z = \frac{1-p_I}{3}$. We then see that $\lambda^2 = \lambda^3 = \lambda^4 = p_I - \frac{1-p_I}{3} = \frac{4p_I-1}{3} = \frac{4F-1}{3}$, which we see is the original λ treated in the main text.

C.3 Swapping of states

Let $|\psi_d\rangle \equiv \frac{1}{\sqrt{d}} \sum_{i=0}^{d-1} |ii\rangle$ be a qudit maximally entangled state and set $|\psi_d\rangle\langle\psi_d| \equiv \psi_d$. Let $\mathcal{N}$ be a qudit Pauli channel acting on one-half of a qudit maximally entangled state. We will first show that the side the channel acts on is immaterial (i.e. $\mathbb{I} \otimes \mathcal{N}[\psi_d] = \mathcal{N} \otimes \mathbb{I}[\psi_d]$) if $\mathcal{N}$ is X-symmetric (which we define shortly). Thus even if X-symmetric noise has acted on both sides of $|\psi_d\rangle$, we are free to pretend there is one qudit Pauli channel $\mathcal{N}$ acting on an arbitrary side, yielding a state $\mathbb{I} \otimes \mathcal{N}[\psi_d] = \mathcal{N} \otimes \mathbb{I}[\psi_d] \equiv \rho_{\mathcal{N}}$. Secondly, we will show that swapping two states $\rho_{\mathcal{N}_1}, \rho_{\mathcal{N}_2}$ yields a state $\rho_{(\mathcal{N}_1 \circ \mathcal{N}_2)}$.

C.3.1 Transpose trick for X symmetric channels

The well-known transpose trick [62] states that if A is any square matrix (of the appropriate size), the following holds

$$\mathbb{I} \otimes A\,|\psi_d\rangle = A^{\mathsf{T}} \otimes \mathbb{I}\,|\psi_d\rangle\ . \tag{41}$$

Here A^{T} is the transpose of A with respect to the basis $\{|i\rangle\}_{i=1}^{d-1}$. Applying the transpose trick to $\mathbb{I} \otimes \mathcal{N}[\psi_d]$ yields

$$\mathbb{I}\otimes\mathcal{N}\left[\psi_d\right]=\sum_{a,b=0}^{d-1}p_{X^aZ^b}\left(\mathbb{I}\otimes X^aZ^b\right)\left[\psi\right] \tag{42}$$

$$=\sum_{a,b=0}^{d-1}p_{X^aZ^b}\left(\left(X^aZ^b\right)^{\mathsf{T}}\otimes\mathbb{I}\right)\left[\psi\right] \tag{43}$$

$$=\sum_{a,b=0}^{d-1}p_{X^aZ^b}\left(X^{-a}Z^b\otimes\mathbb{I}\right)\left[\psi\right]\ , \tag{44}$$

where we used that $\left(X^aZ^b\right)^{\mathsf{T}}=Z^bX^{-a}$, which is equal to $X^{-a}Z^b$ up to a phase.

The above implies that the side a qudit Pauli channel is applied to is immaterial if and only if $p_{X^aZ^b}=p_{X^{-a}Z^b}$. This means that the condition holds when the probability of applying X^aZ^b is equal to the probability of applying $X^{-a}Z^b$. This property holds for several relevant classes of channels, such as arbitrary qubit Pauli channels, arbitrary qudit dephasing noise (since the probability of applying X^aZ^b is zero when $a\neq 0$) and qudit depolarizing noise.

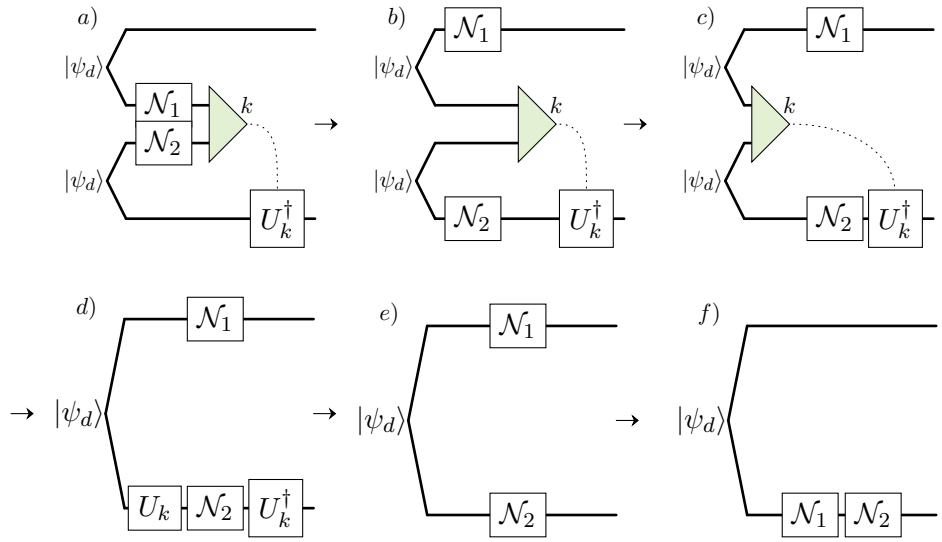

Figure 11: Graphical description of the proof of the desired properties of X-symmetric channels. From a) to b) we use the transpose trick to show that the side on which the noise is applied is immaterial. From b) to c) we use that the application of the noise and the Bell state measurement commute. From c) to d) we performed the projection into one of the (generalized) Bell states. From d) to e) we use the commutativity of qudit Pauli operators (up to a phase). From e) to f) we use the transpose trick again to move the noise all to one side.

C.3.2 Swapping of X-symmetric noisy states

In the previous section we have shown that we have the freedom to move X-symmetric channels from one side of the maximally entangled state to the other. With this tool in hand, let us perform a swap on two such noisy states, see Fig. 11. In a) we show the complete procedure of swapping, including the measurement and classical correction. We have assumed without loss of generality that the channels $\mathcal{N}_1$ and $\mathcal{N}_2$ act on the qudits on which the Bell state measurement (green triangle) is performed. After the Bell state measurement, a classical outcome k is recorded, and is used to specify which correction $U_k^\dagger$ needs to be performed on one of the sides after measuring. Note that the corrections $U_k^\dagger$ are generalized Pauli operators.

In step b) we use the result from the previous section that we are free to move X-symmetric channels to the other side. In step c), we commute the Bell state measurement and the channels, which is allowed

since the operations act on different qubits. In step d) we use that measuring outcome k projects the state unto $\mathbb{I} \otimes U_k |\psi_d\rangle$. From d) to e) we use that the qudit Pauli operators commute (up to an immaterial phase), and finally we use that X-symmetric channels can be moved freely between the two sides.

We close with one final remark. The average values for the λ_i parameters can be interpreted as the λ_i parameters of the average channel acting on the entangled state shared between Alice and Bob. This is because convex combinations of qudit Pauli channels map exactly to convex combinations of the $\vec{\lambda}$, since the Fourier transform is linear.

D Recursion calculations

Here we provide the details for the recursive calculation for the analytics of the average noise parameter.

D.1 Recursion calculation without a cut-off policy

As discussed in Section 3, calculating the average Λ_n parameter requires us first to express H_n^t recursively, from which $H_n \equiv \mathbb{E}[\Lambda_n]$ can be found. The main idea is to express the map from H_n^t to H_{n+1}^t in terms of a linear transformation on the real vector space spanned by $q^{at}\lambda^{bt}$, $a, b \in \mathbb{Z}$. That is, all finite linear combinations of the form $\sum_m c_{a,b} q^{at}\lambda^{bt}$, where $c_{a,b} \in \mathbb{R}$.

As a warm-up, let us consider the first recursion step. Here and in what follows, we will drop the $\frac{1-q}{q}$ prefactor, i.e. we set

$$Z_n = \sum_{\vec{t}} \left(q^{\sum_{i=1}^n t_i}\right) \cdot \left(\lambda^{\sum_{i=1}^{n-1} |t_i - t_{i+1}|}\right) \tag{45}$$

$$Z_n^t = \sum_{\substack{\vec{t} \text{ s.t.} \\ t_n = t}} \left(q^{\sum_{i=1}^n t_i}\right) \cdot \left(\lambda^{\sum_{i=1}^{n-1} |t_i - t_{i+1}|}\right) \ , \tag{46}$$

such that $H_n = \left(\frac{1-q}{q}\right)^n Z_n$. Note that Z_1^t is given by q^t, since there is only the associated probability of succeeding at time t and no decoherence.

Let M correspond to the map in the recursion, i.e. the map $Z_i^t \xrightarrow{M} Z_{i+1}^t$. We then have for the first step that

$$Z_1^t \xrightarrow{M} Z_2^t = \sum_{t'=1}^{\infty} q^t \lambda^{|t-t'|} Z_1^{t'} \tag{47}$$

$$= \left(\sum_{t'=1}^{t-1} q^t \lambda^{t-t'} Z_1^{t'}\right) + \left(\sum_{t'=t}^{\infty} q^t \lambda^{t'-t} Z_1^{t'}\right) \tag{48}$$

$$= \left(\sum_{t'=1}^{t-1} q^t \lambda^{t-t'} q^{t'}\right) + \left(\sum_{t'=t}^{\infty} q^t \lambda^{t'-t} q^{t'}\right) \tag{49}$$

$$= q^t \lambda^t \left(\sum_{t'=1}^{t-1} \left(\frac{q}{\lambda}\right)^{t'}\right) + \left(\frac{q}{\lambda}\right)^t \left(\sum_{t'=t}^{\infty} (q\lambda)^{t'}\right) \tag{50}$$

$$= q^t \lambda^t \left(\frac{\frac{q}{\lambda} - \left(\frac{q}{\lambda}\right)^t}{1 - \frac{q}{\lambda}}\right) + \left(\frac{q}{\lambda}\right)^t \left(\frac{q\lambda - (q\lambda)^t}{1 - q\lambda}\right) \tag{51}$$

$$= \left(-\frac{q\left(\frac{1}{\lambda} - \lambda\right)}{(1 - \lambda^{-1}q)(1 - \lambda q)}\right) \cdot q^{2t}\lambda^{0t} + \frac{q}{\lambda - q} \cdot q^t \lambda^t \ . \tag{52}$$

Note that the limit as q goes to λ and the limit of q goes $\frac{1}{\lambda}$ exists. Observe that we end up with a linear combination of terms of the form $q^{at}\lambda^{bt}$. This is not specific to starting with $Z_1^t = q^t$, as we see in the following,

$$q^{at}\lambda^{bt} \xrightarrow{M} \sum_{t'=1}^{\infty} q^t \lambda^{|t-t'|} \cdot \left(q^{at'}\lambda^{bt'}\right) \tag{53}$$

$$= \left(\sum_{t'=1}^{t-1} q^t \lambda^{t-t'} \cdot \left(q^{at'}\lambda^{bt'}\right)\right) + \left(\sum_{t'=t}^{\infty} q^t \lambda^{t'-t} \cdot \left(q^{at'}\lambda^{bt'}\right)\right) \tag{54}$$

$$= C^{a,b} q^{(a+1)t}\lambda^{bt} + D^{a,b} q^t \lambda^t \ , \tag{55}$$

where

$$C^{a,b} = -\frac{q^a\lambda^b\left(\frac{1}{\lambda} - \lambda\right)}{\left(1-\lambda^{b-1}q^a\right)\left(1-\lambda^{b+1}q^a\right)} \tag{56}$$

$$D^{a,b} = \frac{q^a}{\lambda^{1-b} - q^a} \ . \tag{57}$$

The $C^{a,b}$ and $D^{a,b}$ depend only on the values of $a, b \in \mathbb{Z}$ and q and λ.

We can think of the $q^{at}\lambda^{bt} \equiv |a,b\rangle$ for $a, b \in \mathbb{Z}$ as the basis vectors of a vector space V. The map M defined for an arbitrary $|a,b\rangle$ is, in fact, a linear map from V to V, and so works for all expressions that are sums of terms of the form $q^{at}\lambda^{bt}$.

This allows us to calculate Z_n^t fast. Setting $q^{at}\lambda^{bt} \equiv |a,b\rangle$, the map M takes $|a,b\rangle$ to $C^{a,b}\,|a+1,b\rangle + D^{a,b}\,|1,1\rangle$. As such, we find that

$$Z_1^t = |1,0\rangle \tag{58}$$

$$\xrightarrow{M} Z_2^t = C^{1,0}\,|2,0\rangle + D^{1,0}\,|1,1\rangle \tag{59}$$

$$\xrightarrow{M} Z_3^t = C^{1,0}\left(C^{2,0}\,|3,0\rangle + D^{2,0}\,|1,1\rangle\right) + D^{1,0}\left(C^{1,1}\,|2,1\rangle + D^{1,1}\,|1,1\rangle\right) \tag{60}$$

$$= C^{1,0}C^{2,0}\,|3,0\rangle + D^{1,0}C^{1,1}\,|2,1\rangle + \left(C^{1,0}D^{2,0} + D^{1,0}D^{1,1}\right)|1,1\rangle \ . \tag{61}$$

This already gives us a method to fairly straightforwardly compute Z_n^t, especially when done on a computer.

Finally, we need to extract Z_n^t from Z_n. Fortunately, since

$$\sum_{t=1}^{\infty} q^{at}\lambda^{bt} = \frac{q^a\lambda^b}{1-q^a\lambda^b} \ , \tag{62}$$

this can be seen as a linear form defined on the basis vectors by $|a,b\rangle \mapsto \frac{q^a\lambda^b}{1-q^a\lambda^b}$. Using the above idea and equation (61), we find that the expectation value for Λ_3 for a homogeneous repeater chain of three segments is given by

$$\mathbb{E}\left[\Lambda_3\right] = \left(\frac{1-q}{q}\right)^3 \left(C^{1,0}C^{2,0}\frac{q^3}{1-q^3} + D^{1,0}C^{1,1}\frac{q^2\lambda}{1-q^2\lambda} + \left(C^{1,0}D^{2,0} + D^{1,0}D^{1,1}\right)\frac{q\lambda}{1-q\lambda}\right) \ . \tag{63}$$

D.2 Recursion calculation with a global cut-off policy

We take the same approach as in the previous section; the only difference now is that t' ranges from 1 to T_c. Define

$$\bar{Z}_n^t = \sum_{\substack{\bar{t} \text{ s.t.} \\ t_n = t \\ t_i \leq T_c}} \left(q^{\sum_i^n t_i}\right) \cdot \left(\lambda^{\sum_i^{n-1} |t_i - t_{i+1}|}\right) \ , \tag{64}$$

such that $\bar{Z}_n = \sum_{t=1}^{T_c} \bar{Z}_n^t$ and $\bar{H}_n = \left(\frac{1-q}{q(1-q^{T_c})}\right)^n \bar{Z}_n$.
This gives the following recursion relation,

$$q^{at}\lambda^{bt} \xrightarrow{\bar{M}} \sum_{t'=1}^{T_c} q^t \lambda^{|t-t'|} \cdot \left(q^{at'}\lambda^{bt'}\right) \tag{65}$$

$$= \left(\sum_{t'=1}^{t-1} q^t \lambda^{t-t'} \cdot \left(q^{at'}\lambda^{bt'}\right)\right) + \left(\sum_{t'=t}^{T_c} q^t \lambda^{t'-t} \cdot \left(q^{at'}\lambda^{bt'}\right)\right) \tag{66}$$

$$= C^{a,b} q^{(a+1)t} \lambda^{bt} + D^{a,b} q^t \lambda^t + E^{a,b} q^t \lambda^{-t} \tag{67}$$

$$= C^{a,b} \left|a+1, b\right\rangle + D^{a,b} \left|1, 1\right\rangle + E^{a,b} \left|1, -1\right\rangle \ , \tag{68}$$

where $C^{a,b}$ and $D^{a,b}$ are as before and

$$E^{a,b} = -\frac{q^{a(1+T_c)} \lambda^{(b+1)(T_c+1)}}{1 - \lambda^{b+1} q^a} \ . \tag{69}$$

As in the previous section, by repeating this map $\bar{Z}_n^t$ can be calculated. Finally, since $\sum_{t'=1}^{T_c} q^{at}\lambda^{bt} = \frac{q^a\lambda^b - q^{a(T_c+1)}\lambda^{b(T_c+1)}}{1-q^a\lambda^b}$ we find that the linear form that takes $\bar{Z}_n^t$ to $\bar{Z}_n$ is defined by $|a,b\rangle \mapsto \frac{q^a\lambda^b - q^{a(T_c+1)}\lambda^{b(T_c+1)}}{1-q^a\lambda^b}$.

E Expected delivery time and secret-key rate calculation

The secret-key rate is defined as the ratio between the secret-key fraction and the average delivery time. In the following we discuss how to calculate the secret-key fraction, how to determine the average delivery time when using a global cut-off policy, and finally an upper and lower bound to the average delivery time.

E.1 Secret-key fraction

The secret-key rate is defined as the ratio between the secret-key fraction and the average delivery time. The secret-key fraction is the ratio between the number of secret bits that can be extracted and the number of measured states. The secret-key fraction depends on the used protocol. Here we consider fully asymmetric BB84 [25, 11, 12, 63]. The 'BB84' refers to the original quantum key distribution protocol [25], where both parties measure in two bases (X and Z). The 'fully asymmetric' refers to the fact that only in a vanishingly small subset of cases either of the two parties measures in X. Furthermore, we will also assume that we run the protocol for an asymptotic number of rounds, such that finite-size effects can be ignored.

For fully asymmetric BB84, the secret-key fraction can be expressed in terms of Λ as follows [63],

$$\mathrm{SKF}\left(\Lambda\right) = \max\left(0,\ 1 - 2 \cdot H\left(\frac{1-\Lambda}{2}\right)\right) \ , \tag{70}$$

$$\text{where } H(p) = -p\log_2(p) - (1-p)\log_2(1-p) \ . \tag{71}$$

E.2 Expected delivery time

Calculating the secret-key rate above, in Section E.1, includes the expected delivery time. Without a cut-off, this is given by

$$\mathbb{E}\left[\max\left(T_1, T_2, \ldots, T_n\right)\right] = \sum_{k=1}^{n} \binom{n}{k} \frac{(-1)^{k+1}}{1-(1-p)^k} \ , \tag{72}$$

see [22, 64].

For the cut-off case, we first define an 'attempt' as the consecutive collection of rounds in which entanglement generation was attempted before a success or a reset. In the presence of a global cut-off, the expected delivery time is the sum of two terms: the time that the protocol has spent in attempts that were eventually restarted because the cut-off was met, and the time until successful entanglement generation in the final attempt, which occurred before the cut-off time.

For the first term, we note that the expected number of resets is $\frac{1}{P_{\text{succ}}} - 1$, where $P_{\text{succ}} = \left(1-(1-p)^{T_c}\right)^n$ is the probability that the cut-off is not reached in a single attempt and T_c the cut-off time. Since every time a reset is enforced T_c rounds have been performed, the first term equals $\left(\frac{1}{P_{\text{succ}}} - 1\right) T_c$,

The second term is the expectation value of the delivery time, *conditioned* on the fact that entanglement generation succeeds before T_c:

$$\sum_{t_{\max}=1}^{T_c} t_{\max} \cdot \left(\frac{f(t_{\max})}{\left(1-(1-p)^{T_c}\right)^n}\right) \tag{73}$$

where

$$f(t_{\max}) = \left(1-(1-p)^{t_{\max}}\right)^n - \left(1-(1-p)^{t_{\max}-1}\right)^n$$

is the probability that an attempt succeeds exactly at timestep $t_{\max}$, and the denominator $\left(1-(1-p)^{T_c}\right)^n$ is the probability that entanglement generation succeeds before T_c.

Combining the two terms, we find that the expected delivery time in the presence of a cut-off T_c is given by

$$\left(\frac{1}{\left(1-(1-p)^{T_c}\right)^n} - 1\right) T_c + \sum_{t_{\max}=1}^{T_c} t_{\max} \cdot \left(\frac{f(t_{\max})}{\left(1-(1-p)^{T_c}\right)^n}\right) \ . \tag{74}$$

From here we straightforwardly calculate the generation time when all segments always wait until the cut-off timer has been reached before starting entanglement generation,

$$\frac{T_c}{\left(1-(1-p)^{T_c}\right)^n} \ . \tag{75}$$

F Derivation of the generating function

We will show here the derivation of the function discussed in Section 4, i.e. a closed-form expression for

$$G \equiv \sum_{n=1}^{\infty} \mathbb{E}\left[\Lambda_n\right] x^n \ . \tag{76}$$

It will turn out to be more convenient to consider the sum

$$\overline{G} = \sum_{n=1}^{\infty} Z_n x^n \ , \tag{77}$$

$$\text{where } Z_n = \sum_{\bar{t}} \left(q^{\sum_{i=1}^{n} t_i} \right) \cdot \left(\lambda^{\sum_{i=1}^{n-1} |t_i - t_{i+1}|} \right) \ . \tag{78}$$

Here, $\bar{t}$ runs over all length-n sequences of strictly positive integers, i.e. all possible combinations of times at which a link is generated over the n repeater segments individually. Since $\mathbb{E}\left[\Lambda_n\right] = \left(\frac{1-q}{q}\right)^n Z_n$, we have that

$$G(x) = \overline{G}\left(x \left(\frac{1-q}{q} \right) \right) \ . \tag{79}$$

That is, by replacing x with $x\left(\frac{1-q}{q}\right)$ in $\overline{G}$ we retrieve G, the function of interest.

Before continuing, let us make a few technical remarks. In analytic combinatorics, *combinatorial classes* are studied through the analytical properties of their associated generating function. A combinatorial class can be thought of as a countable collection of objects such that the number of objects with a given 'size' n is finite. However, the number of allowed $\bar{t}$ of a fixed length n is infinite when no cut-off is considered. Fortunately, the analytical and combinatorical tools that we employ remain valid. Furthermore, the function G is technically a *multivariate* generating function [43], since the sum has not only a variable x (corresponding to the number of entries n in $\bar{t}$), but also two auxiliary variables q and λ. Finally, a key idea of analytic combinatorics is to treat generating functions as formal power series, ignoring issues of convergence. This can be justified rigorously; see, for example, [42].

In what follows, we first find an expression of G in terms of so-called q-hypergeometric series, after which we find a similar expression for the case of a global cut-off, corresponding to restricting the sum in Eq. (78) to those $\bar{t}$ satisfying $\max(\bar{t}) \leq T_c$.

F.1 No cut-off

We now give a high-level overview of the proof. First, we will use the recursive relation found in Section D that relates Z_n^t to Z_{n+1}^t. This relation can be thought of as a linear map on the vector space spanned by basis vectors of the form $|a,b\rangle \equiv q^{at}\lambda^{bt}$ to itself. Using the well-known correspondence between linear maps and unnormalized Markov chains, the calculation of Z_n^t can be formulated in terms of walks of length $n-1$ on an unnormalized Markov chain. The possible walks on the Markov chain can be split into two disjoint types. The first type is easy to characterize; the second type is more involved. In particular, the second type is the composition of three separate subwalks, the second of which can repeat an arbitrarily large number of times before moving to the third subwalk. The composition of subwalks is most naturally understood in terms of generating functions. That is, the product of the generating functions corresponding to the three aforementioned subwalks keeps track of the relevant data for the second type, as we shall see.

Let us start the proof by recalling the recursive relation from Section D. As before, setting $q^{at}\lambda^{bt} \equiv |a,b\rangle$, we have $Z_1^t = |1,0\rangle$. Each subsequent Z_n^t can be found by repeatedly applying the linear map M, which is defined on the basis vectors by $|a,b\rangle \xrightarrow{M} C^{a,b}\,|a+1,b\rangle + D^{a,b}\,|1,1\rangle$, with

$$C^{a,b} = -\frac{q^a \lambda^b \left(\frac{1}{\lambda} - \lambda\right)}{\left(1-\lambda^{b-1}q^a\right)\left(1-\lambda^{b+1}q^a\right)} \ , \tag{80}$$

$$D^{a,b} = \frac{q^a}{\lambda^{1-b} - q^a} \ . \tag{81}$$

With the above approach, we found in Sec. D, Eq. (61) for example that

$$Z_1^t = |1,0\rangle \tag{82}$$

$$\xrightarrow{M} Z_2^t = C^{1,0}\,|2,0\rangle + D^{1,0}\,|1,1\rangle \tag{83}$$

$$\xrightarrow{M} Z_3^t = C^{1,0}\left(C^{2,0}\,|3,0\rangle + D^{2,0}\,|1,1\rangle\right) + D^{1,0}\left(C^{1,1}\,|2,1\rangle + D^{1,1}\,|1,1\rangle\right) \tag{84}$$

$$= C^{1,0}C^{2,0}\,|3,0\rangle + D^{1,0}C^{1,1}\,|2,1\rangle + \left(C^{1,0}D^{2,0} + D^{1,0}D^{1,1}\right)|1,1\rangle \ . \tag{85}$$

An approach to determine Z_n^t or Z_n would correspond to understanding each of the allowed terms $|a',b'\rangle$ and their associated *weights*, e.g. the weight of $|1,1\rangle$ after two applications of M is given by $C^{1,0}D^{2,0} + D^{1,0}D^{1,1}$. We can think of the repeated application of M as summing each of the possible walks that one can take in the Markov chain below, see Fig. 12. For each step, an additional multiplicative factor of $C^{a,b}$ or $D^{a,b}$ is incurred. After $n-1$ steps (i.e. $n-1$ applications of M), there will be a number of walks that end up at a given $|a',b'\rangle$. The sum of the total weight of each walk that ends up at a given $|a',b'\rangle$ after $n-1$ steps is exactly the coefficient (i.e. the weight) corresponding to $|a',b'\rangle$, when expanding Z_n^t in the basis $\{|a,b\rangle\}_{a,b}$.

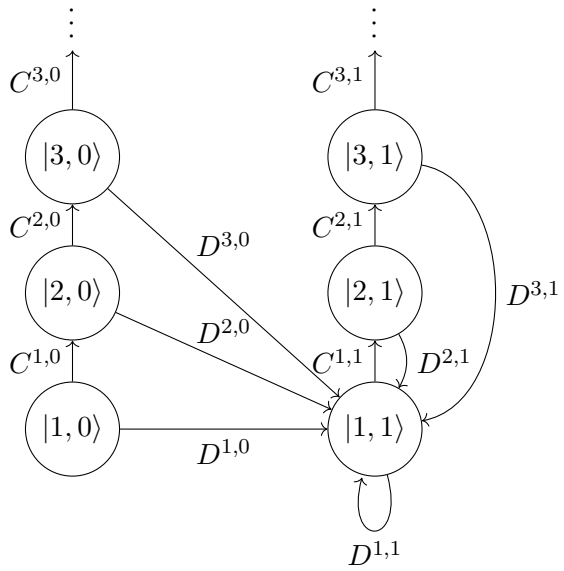

Figure 12: An unnormalized Markov chain with transition 'probabilities' given by the $C^{a,b}$ and $D^{a,b}$. The starting point is $|1,0\rangle$. There are two types of finite walks, those that stay in the left branch (corresponding to $b=0$), and those that transition to the right branch at a certain point. The second type can be decomposed into three parts—a walk staying in the left branch, then any number of ascents followed by a transition back to $|1,1\rangle$, and finally one more ascent.

Thus, doing this for each possible $|a',b'\rangle$ allows us to retrieve the full Z_n^t. At first sight, this seems like a hopeless task—it would require not only understanding the possible walks from $|1,0\rangle$ to $|a',b'\rangle$ (for each a', b') in $n-1$ steps, but also keeping track of the associated weights. Fortunately, such problems can be dealt with using a generating function approach, as we will show in the following. For an extensive overview on generating functions see [42].

Note that the above Markov chain has two distinct 'branches' — one with $b=0$ and one with $b=1$. First, we deal with the case of $b=0$. The only walk that ends in the $b=0$ branch is the one that stays in it for each of the $n-1$ steps, corresponding to multiplying by $C^{a,b}$ for each of the $n-1$ steps. This is because after taking a step corresponding to the $D^{a,b}$ coefficient, the b parameter will remain equal to 1.

The case for $b=1$ is a bit more complicated. Any possible such walk decomposes into three distinct *subwalks*. The first subwalk stays in the $b=0$ branch for a number of steps before moving to the $b=1$

branch. The second subwalk consists of an arbitrary number of 'loops' in the $b = 1$ branch. A 'loop' here corresponds to starting from $|1, 1\rangle$, 'ascending' to some $|a', 1\rangle$, and dropping down to $|1, 1\rangle$ again. Finally, there will be a final ascent to some $|a', 1\rangle$. The final ascent is considered separately, since the weight associated to $|a', 1\rangle$ will acquire a multiplicative factor of $\frac{1-q^{a'}\lambda}{q^{a'}\lambda}$, due to the linear form taking Z_n^t to Z_n, see Eqs. (61) and (63).

The above characterization of the walks allows us to reduce the calculation of $\overline{G}$ to the calculation of the generating functions of the above walks. In other words, we have that

$$\overline{G} = \overline{G}_{0\to\text{end}} + \overline{G}_{b=1} \tag{86}$$

$$= \overline{G}_{0\to\text{end}} + \overline{G}_{0\to1} \cdot \overline{G}^{\text{seq}}_{1\to1} \cdot \overline{G}_{1\to\text{end}} \ . \tag{87}$$

Here $\overline{G}_{0\to1}$ is the generating function corresponding to staying in the $b = 0$ branch, while $\overline{G}_{b=1}$ is the generating function for walks ending up in the $b = 1$ branch. Furthermore, $\overline{G}_{0\to1}$ is the generating function for walks starting in the $b = 0$ branch ending with a transition to the $b = 1$ branch, $\overline{G}^{\text{seq}}_{1\to1}$ is the generating function of all possible walks starting and ending at $|1, 1\rangle$, and $\overline{G}_{1\to\text{end}}$ the generating function of all ascents starting at $|1, 1\rangle$ (including the term corresponding to the linear form). In the first line we used that the generating function of two disjoint sets is the sum of the generating functions. In the second line we used the simple but powerful fact that the generating function of the Cartesian product of two classes equals the product of the generating functions of the two classes [42]. The last statement can be seen as follows. Let there be two walks of two different types, where a_i and b_j are the weights associated to walks of length i and j for the two types, respectively. Summing the product of the weights over every possible path of total length n yields $\sum_{k=0}^{n} a_k b_{n-k}$. The claim then follows since

$$\left(\sum_{n=0}^{\infty} a_n x^n\right)\left(\sum_{n=0}^{\infty} b_n x^n\right) = \sum_{n=0}^{\infty} c_n x^n \tag{88}$$

$$\text{where } c_n = \sum_{k=0}^{n} a_k b_{n-k} \ . \tag{89}$$

Now, it remains to determine the generating functions $\overline{G}_{0\to\text{end}}$, $\overline{G}_{0\to1}$, $\overline{G}^{\text{seq}}_{1\to1}$ and $\overline{G}_{1\to\text{end}}$. First, we consider the $\overline{G}_1$ case. The transition from $|1, 0\rangle$ to $|2, 0\rangle$ corresponds to a weight of $C^{1,0}$. Similarly, the transition from $|2, 0\rangle$ to $|3, 0\rangle$ corresponds to a weight of $C^{2,0}$. This yields a total contribution of $C^{1,0}C^{2,0}$ — exactly the term in front of the $|3, 0\rangle$ term in Eq. (85). Generalizing, a single ascent from $|1, 0\rangle$ to $|m + 1, 0\rangle$ acquires a weight of

$$x^{m+1}\prod_{a=1}^{m} C^{a,0} = x^{m+1}\prod_{a=1}^{m}\left(-\frac{q^a\left(\frac{1}{\lambda}-\lambda\right)}{\left(1-\frac{1}{\lambda}q^a\right)\left(1-\lambda q^a\right)}\right) \tag{90}$$

$$= x^{n+1}\prod_{a=0}^{n-1}\left(-\frac{q^a q\left(\frac{1}{\lambda}-\lambda\right)}{\left(1-\frac{q}{\lambda}q^a\right)\left(1-(\lambda q)\,q^a\right)}\right) \tag{91}$$

$$= x\,(-1)^n\, q^{\binom{n}{2}}\frac{\left[\left(\frac{1}{\lambda}-\lambda\right)qx\right]^n}{\left(\frac{q}{\lambda};q\right)_n\left(\lambda q;q\right)_n} \ . \tag{92}$$

where $(a;q)_m \equiv \prod_{i=0}^{m-1}(1-aq^i)$ is the q-Pochhammer symbol [46]. The x term keeps track of the corresponding number of segments in the repeater chain.

Furthermore, due to the linear form taking Z_n^t to Z_n (see Eq. (62)), each weight corresponding to $|m + 1, 0\rangle$ also acquires a multiplicative factor of $\frac{q^{m+1}}{1-q^{m+1}}$, leading to a total weight of

$$x\,(-1)^m\,q^{\binom{m}{2}}\frac{\left[\left(\frac{1}{\lambda}-\lambda\right)qx\right]^m}{\left(\frac{q}{\lambda};q\right)_m\left(\lambda q;q\right)_m}\cdot\frac{q^{m+1}}{1-q^{m+1}} \tag{93}$$

$$=qx\,(-1)^m\,q^{\binom{m}{2}}\frac{\left[\left(\frac{1}{\lambda}-\lambda\right)q^2x\right]^m}{\left(\frac{q}{\lambda};q\right)_m\left(\lambda q;q\right)_m}\cdot\frac{1}{1-q^{m+1}} \tag{94}$$

$$=\frac{qx}{1-q}\,(-1)^m\,q^{\binom{m}{2}}\frac{(q;q)_m\cdot\left[\left(\frac{1}{\lambda}-\lambda\right)q^2x\right]^m}{\left(\frac{q}{\lambda};q\right)_m\left(\lambda q;q\right)_m\left(q^2;q\right)_m} \tag{95}$$

$$=\frac{qx}{1-q}\,(-1)^m\,q^{\binom{m}{2}}\frac{(q;q)_m\left(q;q\right)_m\cdot\left[\left(\frac{1}{\lambda}-\lambda\right)q^2x\right]^m}{(q;q)_m\left(\frac{q}{\lambda};q\right)_m\left(\lambda q;q\right)_m\left(q^2;q\right)_m}\ . \tag{96}$$

In the first step we took the expression from Eq. (92) and multiplied by the term acquired from the linear map, i.e. $\frac{q^{m+1}}{1-q^{m+1}}$. In the second step we pulled the numerator of $\frac{q^{m+1}}{1-q^{m+1}}$ into the first part. In the third line we use that

$$\frac{1}{1-aq^{m+1}}=\frac{(aq;q)_m}{(1-aq)\left(aq^2;q\right)_m}\ . \tag{97}$$

In the final step we multiply by $\frac{(q;q)_m}{(q;q)_m}$, which will turn out to be convenient in the next step.

The expression in Eq. (96) corresponds to the weight of $|m+1,0\rangle$. The total contribution is then given over all m, i.e.

$$\overline{G}_{0\to\text{end}}=\frac{qx}{1-q}\sum_{m=0}^{\infty}(-1)^m\,q^{\binom{m}{2}}\frac{(q;q)_m\left(q;q\right)_m\cdot\left[\left(\frac{1}{\lambda}-\lambda\right)q^2x\right]^m}{(q;q)_m\left(\frac{q}{\lambda};q\right)_m\left(\lambda q;q\right)_m\left(q^2;q\right)_m} \tag{98}$$

$$=\frac{qx}{1-q}{}_3\phi_3\left[\begin{matrix}q,\ q,\ 0\\ \frac{q}{\lambda},\ \lambda q,\ q^2\end{matrix}\,\middle|\,q,xq^2\left(\frac{1}{\lambda}-\lambda\right)\right]\ , \tag{99}$$

where

$${}_j\phi_k\left[\begin{matrix}c_1,\ c_2,\ \ldots,\ c_j\\ d_1,\ d_2,\ \ldots,\ d_k\end{matrix}\,\middle|\,q,x\right]\equiv\sum_{m=0}^{\infty}\frac{\left(c_1,c_2,\ldots,c_j;q\right)_m}{\left(d_1,d_2,\ldots,d_k,q;q\right)_m}\left((-1)\,q^{\binom{m}{2}}\right)^{1+k-j}x^m \tag{100}$$

is a so-called *q-hypergeometric series* [46]. In the above we used the common shorthand $(c_1;q)_m\,(c_2;q)_m\cdots(c_k;q)_m\equiv(c_1,c_2,\ldots,c_k;q)_m$. Note the additional q in the q-Pochhammer symbol in the denominator of Eq. 100, motivating the introduction of the term $\frac{(q;q)_m}{(q;q)_m}$ in Eq. (96).

We now move on to the $b=1$ case. As mentioned above, any possible walk that ends in the $b=1$ branch consists of three possible *subwalks*, see Eq. (87). Let us first focus on the first subwalk corresponding to staying in the $b=0$ branch before moving to $|1,1\rangle$, i.e. the generating function $\overline{G}_{0\to1}$. The associated weights are similar to before, but with an additional factor of $D^{m+1,0}$. That is,

$$x^{m+1} D^{m+1,0} \prod_{a=1}^{m} C^{a,0} = x^{m+1} \frac{q^{m+1}}{\lambda - q^{m+1}} \prod_{a=1}^{m} \left(- \frac{q^a \left(\frac{1}{\lambda} - \lambda \right)}{\left(1 - \frac{1}{\lambda} q^a \right) \left(1 - \lambda q^a \right)} \right) \tag{101}$$

$$= x^{m+1} \left(\frac{q}{\lambda} \right) \frac{q^m}{1 - \frac{1}{\lambda} q^{m+1}} \prod_{a=0}^{m-1} \left(- \frac{q^a q \left(\frac{1}{\lambda} - \lambda \right)}{\left(1 - \frac{q}{\lambda} q^a \right) \left(1 - (\lambda q)\, q^a \right)} \right) \tag{102}$$

$$= x^{m+1} \left(\frac{q}{\lambda} \right) \frac{q^m}{1 - \frac{1}{\lambda} q^{m+1}} (-1)^m q^{\binom{m}{2}} \frac{\left[\left(\frac{1}{\lambda} - \lambda \right) qx \right]^m}{\left(\frac{q}{\lambda}; q \right)_m (\lambda q; q)_m} \tag{103}$$

$$= x \left(\frac{q}{\lambda} \right) \frac{1}{1 - \frac{1}{\lambda} q^{m+1}} (-1)^m q^{\binom{m}{2}} \frac{\left[\left(\frac{1}{\lambda} - \lambda \right) q^2 x \right]^m}{\left(\frac{q}{\lambda}; q \right)_m (\lambda q; q)_m} \tag{104}$$

$$= x \left(\frac{q}{\lambda} \right) \frac{1}{1 - \left(\frac{q}{\lambda} \right)} (-1)^m q^{\binom{m}{2}} \frac{(q;q)_m \left(\frac{q}{\lambda}; q \right)_m \left[\left(\frac{1}{\lambda} - \lambda \right) q^2 x \right]^m}{(q;q)_m \left(\frac{q^2}{\lambda}; q \right)_m \left(\frac{q}{\lambda}; q \right)_m (\lambda q; q)_m} \tag{105}$$

$$= x \frac{\left(\frac{q}{\lambda} \right)}{1 - \frac{q}{\lambda}} (-1)^m q^{\binom{m}{2}} \frac{(q;q)_m \left[\left(\frac{1}{\lambda} - \lambda \right) q^2 x \right]^m}{(q;q)_m \left(\frac{q^2}{\lambda}; q \right)_m (\lambda q; q)_m} \,. \tag{106}$$

where we used the identity from Eq. (97) and added once more a factor of $\frac{(q;q)_m}{(q;q)_m}$ in Eq. (105). Summing once more over every value of m we find

$$\overline{G}_{0\to 1} = x \frac{\left(\frac{q}{\lambda} \right)}{1 - \frac{q}{\lambda}} \sum_{m=0}^{\infty} (-1)^m q^{\binom{m}{2}} \frac{(q;q)_m \left[\left(\frac{1}{\lambda} - \lambda \right) q^2 x \right]^m}{(q;q)_m \left(\frac{q^2}{\lambda}; q \right)_m (\lambda q; q)_m} \tag{107}$$

$$= x \frac{\left(\frac{q}{\lambda} \right)}{1 - \frac{q}{\lambda}} {}_2\phi_2 \left[\begin{matrix} q, & 0 \\ \frac{q^2}{\lambda}, & \lambda q \end{matrix} \middle| \, q, x q^2 \left(\frac{1}{\lambda} - \lambda \right) \right] \,. \tag{108}$$

We now deal with the $\overline{G}^{\text{seq}}_{1\to 1}$ term. This is the generating function that corresponds to all walks from $|1,1\rangle$ back to $|1,1\rangle$. Note that such a walk can always be decomposed into several 'loops', each consisting of a number of ascents before dropping back down to $|1,1\rangle$. Let $\overline{G}_{1\to 1}$ be the generating function associated to such a single loop. We then have that

$$\overline{G}^{\text{seq}}_{1\to 1} = 1 + \left(\overline{G}_{1\to 1} \right) + \left(\overline{G}_{1\to 1} \cdot \overline{G}_{1\to 1} \right) + \left(\overline{G}_{1\to 1} \cdot \overline{G}_{1\to 1} \cdot \overline{G}_{1\to 1} \right) + \ldots \tag{109}$$

$$= \frac{1}{1 - \overline{G}_{1\to 1}} \,, \tag{110}$$

at least in terms of formal power series. That is, we can safely ignore details of convergence, see [42]. This follows from the fact that the generating function $\left(\overline{G}_{1\to 1} \right)^k$ corresponds to taking exactly k loops, and that walks with different number of loops k are disjoint. As noted in the main

A loop that ascends to $|m+1,1\rangle$ has weight

$$x^{m+1}D^{m+1,1}\prod_{a=1}^{m}C^{a,1} = x^{m+1}\frac{q^{m+1}}{1-q^{m+1}}\prod_{a=1}^{m}\left(-\frac{q^a\lambda\left(\frac{1}{\lambda}-\lambda\right)}{\left(1-q^a\right)\left(1-\lambda^2q^a\right)}\right) \tag{111}$$

$$= x^{m+1}\frac{q^{m+1}}{1-q^{m+1}}\prod_{a=0}^{m-1}\left(-\frac{q^aq\left(1-\lambda^2\right)}{\left(1-qq^a\right)\left(1-q\lambda^2q^a\right)}\right) \tag{112}$$

$$= xq\frac{1}{1-qq^m}\prod_{a=0}^{m-1}\left(-\frac{q^axq^2\left(1-\lambda^2\right)}{\left(1-qq^a\right)\left(1-q\lambda^2q^a\right)}\right) \tag{113}$$

$$= x\frac{1}{1-qq^m}\left(-1\right)^m q^{\binom{m}{2}}\frac{\left[xq^2\left(1-\lambda^2\right)\right]^m}{\left(q;q\right)_m\left(q\lambda^2;q\right)_m} \tag{114}$$

$$= \frac{x}{1-q}\left(-1\right)^m q^{\binom{m}{2}}\frac{\left[xq^2\left(1-\lambda^2\right)\right]^m}{\left(q^2;q\right)_m\left(q\lambda^2;q\right)_m} \tag{115}$$

$$= \frac{x}{1-q}\left(-1\right)^m q^{\binom{m}{2}}\frac{\left(q;q\right)_m\left[xq^2\left(1-\lambda^2\right)\right]^m}{\left(q;q\right)_m\left(q^2;q\right)_m\left(q\lambda^2;q\right)_m}\ , \tag{116}$$

Where we used that $\left(1-qq^m\right)\left(q;q\right)_m = \left(1-q\right)\left(q^2;q\right)_m$ in Eq. (115). Summing over all values of m then gives

$$\overline{G}_{1\to 1} = \frac{xq}{\left(1-q\right)}\sum_{m=0}^{\infty}\left(-1\right)^m q^{\binom{m}{2}}\frac{\left(q;q\right)_m\left[xq^2\left(1-\lambda^2\right)\right]^m}{\left(q\lambda^2;q\right)_m\left(q;q\right)_m\left(q^2;q\right)_m} \tag{117}$$

$$= \frac{xq}{1-q}{}_2\phi_2\left[\begin{matrix}q,\ 0\\ q^2,\ q\lambda^2\end{matrix}\middle|\, q, xq^2\left(1-\lambda^2\right)\right]\ . \tag{118}$$

We now move to the final piece—the calculation of $\overline{G}_{1\to\text{end}}$. This generating function corresponds to a number of ascents m, before applying the linear form. As such, we find

$$x^{m+1}\frac{\lambda q^{m+1}}{1-\lambda q^{m+1}}\prod_{a=1}^{m}C^{a,1} = x^{m+1}\frac{\lambda q^{m+1}}{1-\lambda q^{m+1}}\prod_{a=0}^{m-1}\left(-\frac{q^aq\left(1-\lambda^2\right)}{\left(1-qq^a\right)\left(1-q\lambda^2q^a\right)}\right) \tag{119}$$

$$= x^{m+1}\frac{\lambda q^{m+1}}{1-\lambda q^{m+1}}\left(-1\right)^m q^{\binom{m}{2}}\frac{\left[q\left(1-\lambda^2\right)\right]^m}{\left(q;q\right)_m\left(q\lambda^2;q\right)_m} \tag{120}$$

$$= x\frac{\lambda q}{1-\lambda q}\left(-1\right)^m q^{\binom{m}{2}}\frac{\left(\lambda q;q\right)_m\left[xq^2\left(1-\lambda^2\right)\right]^m}{\left(\lambda q^2;q\right)_m\left(q;q\right)_m\left(q\lambda^2;q\right)_m}\ . \tag{121}$$

Summing over m yields

$$\overline{G}_{1\to\text{end}} = x\frac{\lambda q}{1-\lambda q}\sum_{m=0}^{\infty}\left(-1\right)^m q^{\binom{m}{2}}\frac{\left(\lambda q;q\right)_m\left[xq^2\left(1-\lambda^2\right)\right]^m}{\left(\lambda q^2;q\right)_m\left(q;q\right)_m\left(q\lambda^2;q\right)_m} \tag{122}$$

$$= x\frac{\lambda q}{1-\lambda q}{}_2\phi_2\left[\begin{matrix}\lambda q,\ 0\\ \lambda q^2,\ q\lambda^2\end{matrix}\middle|\, q, xq^2\left(1-\lambda^2\right)\right]\ . \tag{123}$$

We have now all ingredients; using Eqs. (78), (79), (87), (99), (108), (110), (118), (123) and $q\equiv 1-p$ we find that

$$G = \sum_{n=1}^{\infty} \mathbb{E}\left[\Lambda_n\right] x^n \tag{124}$$

$$= x \cdot \Phi_1 + x^2 \left(\frac{(1-q)^2 \lambda}{(\lambda - q)(1 - \lambda q)} \right) \frac{\Phi_2 \cdot \Phi_4}{1 - x \cdot \Phi_3} \ , \tag{125}$$

$$\text{with } \Phi_1 = {}_3\phi_3 \left[\begin{matrix} q, & q, & 0 \\ q^2, & \lambda q, & \frac{q}{\lambda} \end{matrix} \middle| \, q, xpq \left(\frac{1}{\lambda} - \lambda \right) \right] \ , \tag{126}$$

$$\Phi_2 = {}_2\phi_2 \left[\begin{matrix} q, & 0 \\ \frac{q^2}{\lambda}, & \lambda q \end{matrix} \middle| \, q, xpq \left(\frac{1}{\lambda} - \lambda \right) \right] \ , \tag{127}$$

$$\Phi_3 = {}_2\phi_2 \left[\begin{matrix} q, & 0 \\ q^2, & q\lambda^2 \end{matrix} \middle| \, q, xpq \left(1 - \lambda^2 \right) \right] \ , \tag{128}$$

$$\Phi_4 = {}_2\phi_2 \left[\begin{matrix} \lambda q, & 0 \\ \lambda q^2, & q\lambda^2 \end{matrix} \middle| \, q, xpq \left(1 - \lambda^2 \right) \right] \ . \tag{129}$$

In the above we have changed x to $x\left(\frac{1-q}{q}\right)$ (see Eq. (79)), and collected the prefactors in front of the q-hypergeometric series.

F.2 Cut-off case

The goal of this section is to find the generating function for the case of a global cut-off. The construction is the same as the case without a cut-off. That is, the recurrence relation can be mapped unto a Markov chain, and the generating function can be recovered by splitting the possible walks into different subwalks (and finding the generating functions of those subwalks). In this section we will only discuss the splitting into subwalks—the determination of the individual generating functions is similar to the previous section. As such, we do not report these calculations here and instead refer to our Mathematica code [26].

In the case of a cut-off, the linear map taking Z_n^t to Z_n^t is defined by $|a,b\rangle \xrightarrow{M} C^{a,b}|a+1,b\rangle + D^{a,b}|1,1\rangle + E^{a,b}|1,-1\rangle$, with $C^{a,b}$, $D^{a,b}$ as before, and

$$E^{a,b} = -\frac{q^{a(t_c+1)}\lambda^{1+t_c+b(1+t_c)}}{1-q^a\lambda^{b+1}} \ , \tag{130}$$

see Section D.2. This map can also be seen in terms of an unnormalized Markov chain, where there are now three branches, $b = 0, 1, -1$, see Fig. 13. This case is more complicated, since a walk might now alternate between the $b = -1$ and $b = 1$ branch. In what follows, we list all possible subwalks. We change notation slightly; for a generating function of the form G_x^y, x refers to the start of the subwalk, while y refers to the end of the subwalk. Furthermore, instead of referring to the branches by their b parameter, we will use the convention that the $b = 0, 1, -1$ branches correspond to $c = 0, 1, 2$, respectively. Finally, we will drop the overline notation, with the understanding that the distinction between the generating functions with and without the replacement $x \mapsto x\frac{1-q}{q(1-q^{T_c})}$ of is clear.

As before, a walk might stay in the $c = 0$ branch, corresponding to the G_0^{end} generating function. On the other hand, a walk might stay in the $c = 0$ branch for some time, before moving to the $c = 1$ or $c = 2$ branch, corresponding to G_0^1 and G_0^2, respectively. Let us focus on the walks that first go to the $c = 1$ branch and also finish there. As before, the walk can stay for an arbitrary number of attempts in the $c = 1$ branch before falling back to $|1,1\rangle$. A single such 'loop' is described by a generating function G_1^1, such that an arbitrary sequence of such 'loops' is given by $\frac{1}{1-G_1^1}$.

Since the walk needs to end in the $c = 1$ branch, it needs to transition back and forth between the $c = 1$ and $c = 2$ branches several times (including potentially zero times). Let us consider the case where it transitions only once to the $c = 2$ branch. The transition itself is captured by some generating

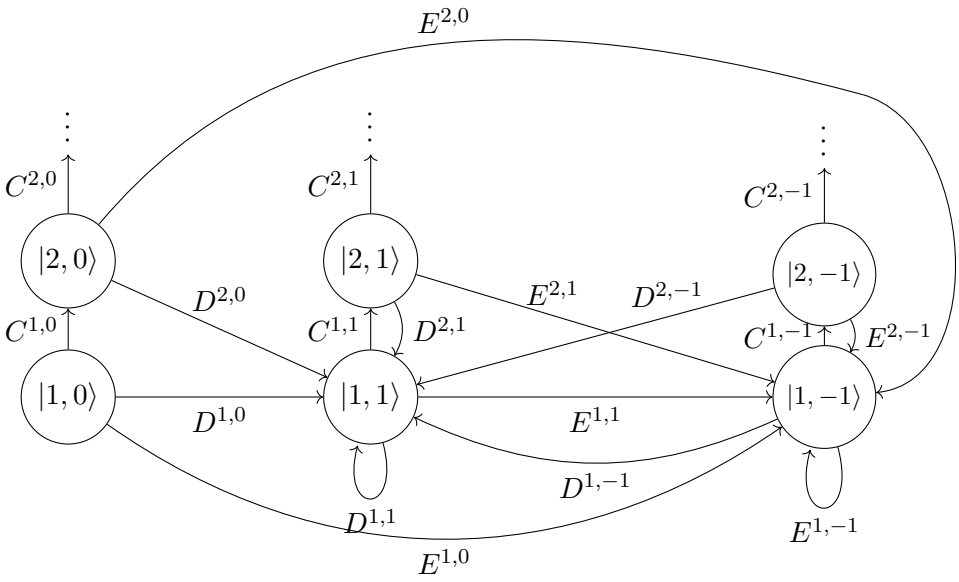

Figure 13: An unnormalized Markov chain with transition 'probabilities' given by $C^{a,b}$, $D^{a,b}$ and $E^{a,b}$. The starting point is $|1,0\rangle$. There are now several different types of walks, explained in the main text.

function G_1^2. As with the $c = 1$ branch, the walk can stay in the $c = 2$ branch for a number of 'loops', corresponding to $\frac{1}{1-G_2^2}$. Afterwards, the walk transitions back to the $c = 1$ branch with associated generating function G_2^1. Up to this point, the generating function of a walk that transitions once back and forth between the $c = 1$ and $c = 2$ branch is given by

$$G_0^1 \cdot \left(\frac{G_1^2}{1 - G_1^1} \cdot \frac{G_2^1}{1 - G_2^2} \right) . \tag{131}$$

The transition from $c = 1$ to $c = 2$ and back can now happen an arbitrary number of times, leading to the following,

$$G_0^1 \cdot \frac{1}{1 - \left(\frac{G_1^2}{1-G_1^1} \cdot \frac{G_2^1}{1-G_2^2} \right)} . \tag{132}$$

Finally, the walk will have an arbitrary number of loops in the $c = 1$ branch, before having one final ascent,

$$G_0^1 \cdot \frac{1}{1 - \left(\frac{G_1^2}{1-G_1^1} \cdot \frac{G_2^1}{1-G_2^2} \right)} \cdot \frac{1}{1 - G_1^1} \cdot G_1^{\text{end}} . \tag{133}$$

In a similar fashion, the generating functions for the walk starting and finishing at $c = 2$ is given by

$$G_0^2 \cdot \frac{1}{1 - \left(\frac{G_2^1}{1-G_2^2} \cdot \frac{G_1^2}{1-G_1^1} \right)} \cdot \frac{1}{1 - G_2^2} \cdot G_2^{\text{end}} . \tag{134}$$

Let us now consider the walk that starts at $c = 1$ and ends at $c = 2$. Note that we can think of this as a path that is very similar to a path that starts and ends at $c = 1$. That is, instead of finishing in the $c = 1$ branch after oscillating between the $c = 1$ and $c = 2$ branch, there is a transition between to the $c = 2$ branch. Finally, there will be a number of 'loops' and one last ascent in the $c = 2$ branch. This yields

$$G_0^1 \cdot \frac{1}{1 - \left(\frac{G_1^2}{1-G_1^1} \cdot \frac{G_2^1}{1-G_2^2} \right)} \cdot \frac{1}{1 - G_1^1} \cdot G_1^2 \cdot \frac{1}{1 - G_2^2} \cdot G_2^{\text{end}} . \tag{135}$$

Similarly, for the walks that start and end at $c = 2$ and $c = 1$, respectively, we find that

$$G_0^2 \cdot \frac{1}{1 - \left(\frac{G_2^1}{1-G_2^2} \cdot \frac{G_1^2}{1-G_1^1}\right)} \cdot \frac{1}{1-G_2^2} \cdot G_2^1 \cdot \frac{1}{1-G_1^1} \cdot G_1^{\text{end}} \,. \tag{136}$$

Adding the generating function of all five types of path and collecting like terms give a generating function of the form

$$G_0^{\text{end}} + \frac{\frac{G_0^1}{1-G_1^1} \cdot \left(G_1^{\text{end}} + \frac{G_1^2 \cdot G_2^{\text{end}}}{1-G_2^2}\right) + \frac{G_1^2}{1-G_2^2} \cdot \left(G_2^{\text{end}} + \frac{G_2^1 \cdot G_1^{\text{end}}}{1-G_1^1}\right)}{1 - \left(\frac{G_1^2}{1-G_1^1} \cdot \frac{G_2^1}{1-G_2^2}\right)} \tag{137}$$

$$= G_0^{\text{end}} + \frac{G_0^1\left(\left(1-G_2^2\right)G_1^{\text{end}} + G_1^2 \cdot G_2^{\text{end}}\right) + G_0^2\left(\left(1-G_1^1\right)G_2^{\text{end}} + G_2^1 \cdot G_2^{\text{end}}\right)}{1 - \left(G_1^1 + G_2^2 + G_1^2 \cdot G_2^1 - G_1^1 \cdot G_2^2\right)} \,. \tag{138}$$

As mentioned previously, the individual generating functions G_x^y themselves are tedious but straightforward to calculate, and we leave them to the Mathematica file [26]. After making the replacement $x \mapsto x\frac{1-q}{q(1-q^{T_c})} \equiv R \cdot x$ we find that the generating function is given by

$$G(x) = \sum_{n=1}^{\infty} \mathbb{E}\left[\Lambda_n\left(T_c\right)\right] x^n \tag{139}$$

$$= \Phi_0^{\text{end}} + Rx \cdot \frac{\Phi_0^1\left(\left(1-\Phi_2^2\right)\Phi_1^{\text{end}} + \Phi_1^2 \cdot \Phi_2^{\text{end}}\right) + \Phi_0^2\left(\left(1-\Phi_1^1\right)\Phi_2^{\text{end}} + \Phi_2^1 \cdot \Phi_2^{\text{end}}\right)}{1 - \left(\Phi_1^1 + \Phi_2^2 + \Phi_1^2 \cdot \Phi_2^1 - \Phi_1^1 \cdot \Phi_2^2\right)} \,, \tag{140}$$

$$\text{with } \Phi_0^{\text{end}} = \sum_{t=1}^{T_c} q^t Rx \cdot {}_2\phi_2\left[\begin{matrix} q,\ 0 \\ \frac{q}{\lambda},\ \lambda q \end{matrix} \middle| q, q^{t+1}Rx\left(\frac{1}{\lambda} - \lambda\right)\right] \,, \tag{141}$$

$$\Phi_0^1 = -\frac{Rx}{1-\frac{\lambda}{q}} {}_2\phi_2\left[\begin{matrix} q,\ 0 \\ \frac{q^2}{\lambda},\ q\lambda \end{matrix} \middle| q, q^2Rx\left(\frac{1}{\lambda} - \lambda\right)\right] \,, \tag{142}$$

$$\Phi_0^2 = -\frac{Rx\left(q\lambda\right)^{T_c+1}}{1-\lambda q} {}_2\phi_2\left[\begin{matrix} q,\ 0 \\ \frac{q}{\lambda},\ q^2\lambda \end{matrix} \middle| q, q^{T_c+2}Rx\left(\frac{1}{\lambda} - \lambda\right)\right] \,, \tag{143}$$

$$\Phi_1^1 = \frac{qRx}{1-q} {}_2\phi_2\left[\begin{matrix} q,\ 0 \\ q^2,\ q\lambda^2 \end{matrix} \middle| q, q^2Rx\left(1 - \lambda^2\right)\right] \,, \tag{144}$$

$$\Phi_2^2 = -\frac{q^{T_c+1}Rx}{1-q} {}_2\phi_2\left[\begin{matrix} q,\ 0 \\ q^2,\ \frac{q}{\lambda^2} \end{matrix} \middle| q, q^{T_c+2}Rx\left(\frac{1}{\lambda^2} - 1\right)\right] \,, \tag{145}$$

$$\Phi_1^2 = -\frac{Rx\left(q\lambda^2\right)^{T_c+1}}{1-q\lambda^2} {}_2\phi_2\left[\begin{matrix} q,\ 0 \\ q,\ q^2\lambda^2 \end{matrix} \middle| q, q^{T_c+2}Rx\left(1 - \lambda^2\right)\right] \,, \tag{146}$$

$$\Phi_2^1 = -\frac{Rx}{1-\frac{\lambda^2}{q}} {}_2\phi_2\left[\begin{matrix} q,\ 0 \\ q,\ \frac{q^2}{\lambda^2} \end{matrix} \middle| q, q^2Rx\left(\frac{1}{\lambda^2} - 1\right)\right] \,, \tag{147}$$

$$\Phi_1^{\text{end}} = \sum_{t=1}^{T_c} \left(q\lambda\right)^t {}_1\phi_1\left[\begin{matrix} 0 \\ q\lambda^2 \end{matrix} \middle| q, q^{t+1}Rx\left(1 - \lambda^2\right)\right] \,, \tag{148}$$

$$\Phi_2^{\text{end}} = \sum_{t=1}^{T_c} \left(\frac{q}{\lambda}\right)^t {}_1\phi_1\left[\begin{matrix} 0 \\ \frac{q}{\lambda^2} \end{matrix} \middle| q, q^{t+1}Rx\left(\frac{1}{\lambda^2} - 1\right)\right] \,. \tag{149}$$

$$\tag{150}$$

F.3 Approximations for the global cut-off

Here we find a tight approximation similar to the case without a cut-off, see Section 4.2. The approximation is given by

$$\begin{aligned}\mathbb{E}\left[\Lambda_n(T_c)\right] &\sim \left[-\mathrm{Res}_\rho\left(G\right)\right]\left(\frac{1}{\rho}\right)^{n+1} \\ &\equiv \left[\overline{A}(\lambda,q,T_c)\right]\cdot\overline{B}(\lambda,q,T_c)^n \quad ,\end{aligned} \tag{151}$$

As with the case without a cut-off, the behavior of the average noise parameter is governed by the simple pole of $G(x)$ closest to $x=0$. Denote by ρ the location of this singularity. As before, ρ is given by the solution to $1-\left(\Phi_1^1+\Phi_2^2+\Phi_1^2\cdot\Phi_2^1-\Phi_1^1\cdot\Phi_2^2\right)=0$. Denote this solution by $x=\rho$. The decay factor $\overline{B}$ is then given by

$$\overline{B}\left(\lambda,q,T_c\right)=\frac{1}{\rho}\ . \tag{152}$$

The other factor $\overline{A}$ is a bit more complicated,

$$\overline{A}\left(\lambda,q,T_c\right)=R\cdot\frac{\left(\Phi_0^1\left(\left(1-\Phi_2^2\right)\Phi_1^{\mathrm{end}}+\Phi_1^2\cdot\Phi_2^{\mathrm{end}}\right)+\Phi_0^2\left(\left(1-\Phi_1^1\right)\Phi_2^{\mathrm{end}}+\Phi_2^1\cdot\Phi_2^{\mathrm{end}}\right)\right)\Big|_{x=\rho}}{\left(\left(\Phi_1^1+\Phi_2^2+\Phi_1^2\cdot\Phi_2^1-\Phi_1^1\cdot\Phi_2^2\right)'\right)\Big|_{x=\rho}}\ . \tag{153}$$

Note that only the $\overline{A}$ function involves a sum over t while $\overline{B}$ does not.