npj | quantum information

Article

Published in partnership with The University of New South Wales

https://doi.org/10.1038/s41534-025-01161-3

Entanglement buffering with multiple quantum memories

Álvaro G. Iñesta[1,2,3,4], Bethany Davies[1,2,3,4] ✉, Sounak Kar[1,2,3] & Stephanie Wehner[1,2,3]

Entanglement buffers are systems that maintain high-quality entanglement, ensuring it is readily available for consumption when needed. We study the performance of a two-node buffer, where each node has one long-lived quantum memory for entanglement storage and multiple short-lived memories for generation. Freshly generated entanglement may be used to purify stored entanglement, which degrades over time. Stored entanglement may be removed due to consumption or failed purification. We derive analytical expressions for the entanglement availability and the average fidelity upon consumption. Our solutions are computationally efficient and provide fundamental bounds to the performance of purification-based entanglement buffers. We also show that purification must be performed as frequently as possible to maximise the average fidelity of entanglement upon consumption, even if this often leads to the loss of high-quality entanglement due to purification failures. Moreover, we obtain heuristics for the design of good purification policies in practical systems.

Entanglement is a fundamental resource for many quantum network applications, including some quantum key distribution protocols[1,2], distributed quantum sensing[3–6], and coordination tasks where communication is either prohibited or insufficiently fast[7,8]. Pre-distributing entanglement between remote parties would eliminate the need to generate and distribute entangled states on demand, saving time and resources[9–12]. However, entanglement degrades over time due to decoherence, preventing long-term storage.

Entanglement buffers are systems that store entanglement until it is needed for an application. Passive buffers, which store entanglement in quantum memories, are constrained by the coherence time of these memories[13]. To overcome this limitation, purification-based entanglement buffers have been proposed[14,15]. These systems store entangled states and employ purification protocols to ensure the states remain high quality, mitigating the effects of decoherence. Purification protocols take m low-quality entangled states as input and produce n higher-quality states as output, typically with $m > n$[16–19]. These protocols often involve some probability of failure, in which case all the input states are lost and no entanglement is produced. Here, we focus on purification-based buffers.

As proposed in ref. 14, the performance of an entanglement buffer can be measured with two quantities: the availability (probability that entanglement is available for consumption when requested, see Definition 2) and the average consumed fidelity (average quality of entanglement at the time of consumption, see Definition 3). As well as having practical utility, entanglement buffers are a useful theoretical tool in order to understand the impact of several important interacting processes that occur in a quantum network: ongoing generation, purification, and consumption of entanglement. Of major interest is the impact of the entanglement purification protocol on the performance of the system. Since the success probability of entanglement purification typically depends on the fidelity of the input states, any rate and fidelity metrics are inherently coupled in systems making use of purification. This coupling adds complexity to analytical calculations. Consequently, most analytical studies on the performance of quantum networking systems exclude purification, and its impact on performance is typically explored with numerical methods[20,21]. Nevertheless, as is a main result in this work, for entanglement buffering systems closed-form solutions are obtainable for a fully general purification protocol. One may then efficiently compute the performance of a particular purification policy, as well as make formal statements about how often purification should be applied to the buffered entanglement.

Here, we study the *1GnB system*: a purification-based entanglement buffer with one good (long-lived) memory and n bad (short-lived) memories. The good memory can store entanglement, which can be consumed at any time by an application. In contrast, bad memories can generate entanglement concurrently but cannot store it; they act as communication qubits. For instance, carbon-13 nuclear spins in diamond can serve as good memories with coherence times up to 1 min[22], while electron spins in

[1]QuTech, Delft University of Technology, Delft, South Holland, The Netherlands. [2]EEMCS, Quantum Computer Science, Delft University of Technology, Delft, South Holland, The Netherlands. [3]Kavli Institute of Nanoscience, Delft University of Technology, Delft, South Holland, The Netherlands. [4]These authors contributed equally: Álvaro G. Iñesta, Bethany Davies. ✉e-mail: bethany.davies@unibas.ch

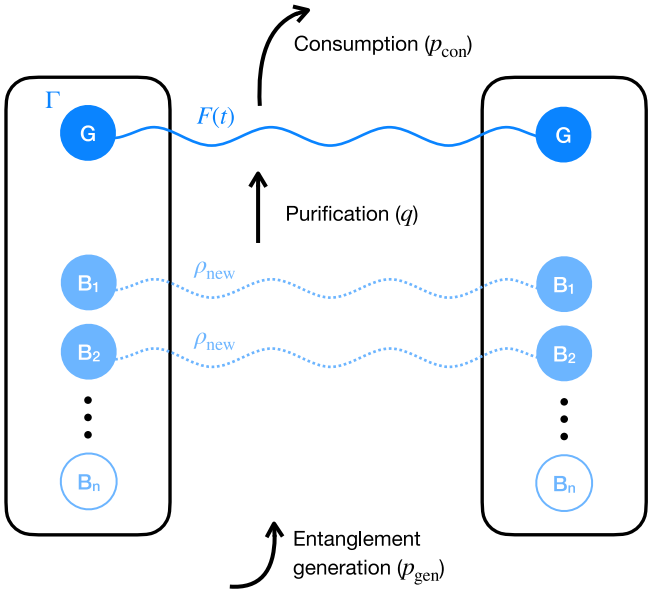

Fig. 1 | Illustration of the 1G*n*B buffering system. Entanglement generation is attempted in every bad memory (B_1,…, B_n) simultaneously in each time slot. Each memory succeeds with probability p_{gen}. The good memory, G, stores entanglement, which decoheres at rate Γ. When G is full, and new entanglement is generated in any of the B memories, a purification subroutine is applied with probability q. Entanglement is consumed from G with probability p_{con} in each time slot.

nitrogen-vacancy centres may function as communication qubits, with coherence times generally below 1 s[23].

Each time entanglement is generated in some of the bad memories, the system may choose to immediately use it to purify the entanglement stored in the good memory. If purification is not attempted, the newly generated entanglement is discarded. We illustrate the 1G*n*B system in Fig. 1. Note that the physical platform must enable easy access to stored entanglement for consumption and purification. However, network activities, such as repeated entanglement generation attempts and purification, may introduce additional noise, reducing memory lifetimes. For example, in ref. 24, even when the carbon-13 nuclear spin used as a storage qubit is protected from network noise by applying stronger magnetic fields, it exhibits a shortened lifetime of approximately 11.6 ms.

The 1G*n*B buffering system is a generalisation of the 1G1B system that was originally proposed in ref. 14. 1G1B is a system with only one good quantum memory and one bad memory. Here, we generalise the work in ref. 14 in three main ways. Firstly, we now consider several (n) bad memories. Including several bad memories in our model now means that there is the possibility of multiplexing. Multiplexing refers to the use of multiple communication qubits to attempt entanglement generation in parallel. This can be achieved, for example, through frequency[13,25,26] or time[27,28] multiplexing, and it effectively boosts the probability of obtaining at least one entangled link per generation attempt. This is a standard strategy to improve the entanglement generation rate in practical platforms[29–31]. Moreover, multiplexing opens up the use of stronger purification protocols, which require the simultaneous presence of multiple links, thereby providing an improvement to system fidelity metrics as well as the rate. Note again that the physical implementation of the buffer must allow for such multiplexing and for purification of the generated entanglement. The second generalisation from previous work is that we now model the system in discrete time rather than continuous time, which is more accurate to real-world systems, as entanglement generation typically happens in discrete attempts (see e.g. refs. 32–35). Finally, we now derive our solutions for a fully arbitrary purification protocol. In particular, the solutions for performance metrics presented in ref. 14 only apply for purification protocols with a constant probability of success (i.e., the success probability must be independent of the fidelity of the buffered quantum state). However, in this work, we remove this assumption and derive closed-form solutions for the availability and the average consumed fidelity of buffers that use arbitrary purification protocols. This is in contrast to ref. 15, where, although performance metrics are derived analytically and the probability of success is not necessarily constant, their computation requires solving a linear system of equations, which has a dimension that scales with system parameters such as the memory lifetime.

In this work, we firstly provide analytical expressions for the availability, A, and the average consumed fidelity, $\overline{F}$, of the 1G*n*B system (see model description in "The 1GnB system"). Then, we use these expressions to find fundamental limits to the performance of entanglement buffers. Lastly, we investigate how the 1G*n*B system should be operated: because there is a large amount of freedom in the choice of purification protocols, it is not clear what purification strategies should be employed to maximise A and $\overline{F}$. For example, would it be beneficial to use a purification subroutine that provides a larger fidelity boost (which could increase $\overline{F}$) if this comes at the cost of a higher probability of failure (which means losing high-quality entanglement more frequently, decreasing A and maybe also $\overline{F}$)? Our main findings are the following:

- MONOTONIC PERFORMANCE – We show that, to maximise the average consumed fidelity, purification must be performed as much as possible, i.e., every time entanglement is generated in any of the bad memories. This holds even if the purification protocol has a large probability of failure. Nevertheless, there is a tradeoff between both performance metrics, since the availability decreases when purification is performed more frequently.
- FUNDAMENTAL BOUNDS — We provide upper and lower bounds for the availability and the average consumed fidelity of a 1G*n*B system, which constitute fundamental limits to the impact that a purification policy can have on the performance.
- SIMPLE CAN BE BETTER THAN OPTIMAL — Simple purification protocols can greatly outperform advanced purification protocols that maximise the fidelity of the output entangled state. For example, we find that a buffering system using the 2-to-1 purification protocol from ref. 17 (known as DEJMPS) can outperform a system using the n-to-1 optimal bilocal Clifford protocol from ref. 36, in terms of both availability and average consumed fidelity.

Results

The 1G*n*B system

In this subsection, we describe the entanglement buffering setup (see Fig. 1). The goal of the system is to buffer bipartite entanglement shared between two nodes. These nodes could be, for example, two end users in a quantum network or two processors in a quantum computing cluster. We refer to bipartite entanglement as an *entangled link* between the two nodes. In the 1G*n*B system:

- Each node has *one long-lived memory* (good, G) and *n short-lived memories* (bad, B).
- The G memories are used to store the entangled link. We assume *the link stored in memory is a Werner state* (any bipartite state can be transformed into a Werner state with the same fidelity by applying extra noise, a process known as *twirling*[37,38]). Such a state can be parametrised with its fidelity to the target maximally entangled state, F.
- The entangled link stored in G is subject to *depolarising noise* with memory lifetime $1/\Gamma$, which causes an exponential decay in fidelity with rate Γ. That is, if the link in memory has an initial fidelity F, after time t, this reduces to

$$F \mapsto \left(F - \frac{1}{4}\right) e^{-\Gamma t} + \frac{1}{4}. \quad (1)$$

Table 1 | Parameters of the 1G1B system

Hardware	
n	Number of short-lived memories
p_{gen}	Probability of a successful entanglement generation attempt
ρ_{new}	Bipartite entangled state is produced after a successful entanglement generation
Γ	Rate of decoherence
Application	
p_{con}	Probability of consumption request
Purification policy	
q	Probability of attempting purification immediately after a successful entanglement generation attempt (otherwise, the new links are discarded)
$J_k(F)$	Jump function. Given a buffered link with fidelity F, $J_k(F)$ is the fidelity immediately following a successful purification using k newly generated links. Rational function with coefficients a_k, b_k, c_k, d_k – see Eq. (3).
$p_k(F)$	Probability of successful purification using k newly generated links. Linear function with coefficients c_k, d_k – see Eq. (4).

See main text for further details

- Before each entanglement generation attempt, the system checks if a new *consumption request* has arrived. The arrival of a new consumption request in each time step occurs with probability p_{con}. If there is a link stored in memory G when a consumption request arrives, the link is immediately consumed and therefore removed from the memory. This takes up the entire time step. If there is no link available, the request is discarded, and the system proceeds with the entanglement generation attempt.
- The B memories are used to generate new entangled links. In the literature, these are usually called communication or broker qubits[39]. This communication qubit can be, for example, the electron spin in a nitrogen-vacancy centre[34,40,41]. Every time step that is not taken up by consumption, *entanglement generation* is attempted in all n bad memories simultaneously, e.g., using frequency or spatial multiplexing, and each of them independently generates an entangled link with probability p_{gen}. This means that, after each multiplexed attempt, the number of successfully generated links follows a binomial distribution with parameters (n, p_{gen}). Each of these new links is of the form ρ_{new}, which is an arbitrary state that depends on the entanglement generation protocol employed (see e.g. refs. 32,33,42,43). We assume that every entanglement generation attempt is heralded: if it fails, a failure flag is raised. Heralded entanglement generation has been proposed theoretically[32] and also demonstrated experimentally[34].
- When $k \geq 1$ entangled links are generated in the B memories and the G memory is empty, one of the links is *transferred* to the G memory with fidelity $F_{\text{new}} = \langle \Phi_{00} | \rho_{\text{new}} | \Phi_{00} \rangle$. Here, it is assumed that the swap operation that transfers the link to the G memory is perfect. We note that it is also possible to consider a noisy swap operation that results in a lower transferred fidelity $F'_{\text{new}} < F_{\text{new}}$, but for simplicity in this work, we consider a perfect initial swap operation. If the G memory is occupied, the new links may be used to *purify* the link in memory. The system decides to attempt purification with probability q. If the system does not decide to purify, the new links are discarded. If the system decides to attempt purification and this succeeds, then the resultant link in the G memory is twirled, converting it into the form of a Werner state with the same fidelity.

Table 1 summarises all variables of the system. Next, we discuss how to model the purification strategy.

The main degree of freedom in the 1GnB system is the choice of purification protocol. This is given by the purification policy.

Definition 1. The *purification policy* π is a function that indicates the purification protocol that must be used when k links are generated in the B memories,

$$\pi : k \in \{1, \ldots, n\} \mapsto \pi(k) \in \mathcal{P}_{k+1}, \tag{2}$$

where $\mathcal{P}_m$ is the set of all m-to-1 purification protocols.

Protocol $\pi(k)$ of purification policy π is the $(k+1)$-to-1 purification protocol that is used when k new links are generated in the B memories (examples of basic protocols can be found in refs. 16,17,44; see ref. 45 for a survey). The purification protocol updates the fidelity of the buffered link from F to $J_k(F)$, where

$$J_k(F) = \frac{1}{4} + \frac{a_k(\rho_{\text{new}})\left(F - \frac{1}{4}\right) + b_k(\rho_{\text{new}})}{c_k(\rho_{\text{new}})\left(F - \frac{1}{4}\right) + d_k(\rho_{\text{new}})}. \tag{3}$$

We call J_k the *jump function of protocol* $\pi(k)$. The protocol succeeds with probability

$$p_k(F) = c_k(\rho_{\text{new}})\left(F - \frac{1}{4}\right) + d_k(\rho_{\text{new}}), \tag{4}$$

otherwise all of the links (including the buffered one) are discarded, and the G memory becomes empty. In Appendix B of ref. 14, the forms (3) and (4) for the output fidelity and success probability are justified, given that the buffered link is a Werner state with fidelity F and any other input state is given by the same arbitrary density matrix ρ_{new}. As shown in ref. 14, a noisy purification protocol (e.g., with noisy gates) is also described by (3) and (4). We therefore see that the action of any purification protocol on the fidelity of the buffered link is determined by the four parameters $a_k(\rho_{\text{new}})$, $b_k(\rho_{\text{new}})$, $c_k(\rho_{\text{new}})$, $d_k(\rho_{\text{new}})$. In Supplementary Note C, we discuss the values that these coefficients can take. As an example, we also provide the explicit form of these coefficients for the well-known 2-to-1 DEJMPS protocol[17]. Throughout this work, we consider *heralded* purification protocols, where measurement outcomes determine success or failure. This is the standard setting in entanglement purification (see e.g., refs. 16,17,44). Lastly, note that purification policy π employs protocol $\pi(k)$ when k new links are generated. However, this does not mean that all the new links are used in the protocol. For example, a policy may simply replace the link in memory with a newly generated link and ignore the rest of the new links.

Given the system description, we now view 1GnB as a discrete-time stochastic process. In particular, at time t the state of the system is the fidelity $F(t)$ of the buffered link, as this is the only quantity that can change over time. If there is no link in the buffered memory at time t, we let $F(t) = 0$. This is for notational convenience, as recalling the decoherence (1), one can never reach zero fidelity if there is a link present.

We now outline the characteristic behaviours of $F(t)$ when moving from time t to time $t+1$.

Let us consider first $F(t) = 0$. If entanglement generation is unsuccessful, in the next time step, the fidelity will remain at that value: $F(t+1) = 0$. If entanglement generation is successful, in the next time step the fidelity will be F_{new}, where $F_{\text{new}} = \langle \Phi_{00} | \rho_{\text{new}} | \Phi_{00} \rangle$ is the fidelity of freshly generated links. We will assume that $F_{\text{new}} > 1/4$.

If $F(t) > 0$, then in the next time step this could evolve in one of the following ways: (i) if no purification is attempted then the fidelity simply decoheres by one unit of time according to Eq. (1); (ii) if k new links are generated and purification is successfully performed, the fidelity decoheres by one time step and is then mapped according to the corresponding jump function (3); (iii) if a consumption request has arrived or if purification fails, the link is removed and the system becomes empty.

In Fig. 2, we illustrate an example of how the fidelity may evolve.

In the following subsection, we define the two performance metrics: the availability and the average consumed fidelity. We then present simple closed-form solutions for these two performance metrics in the 1GnB system.

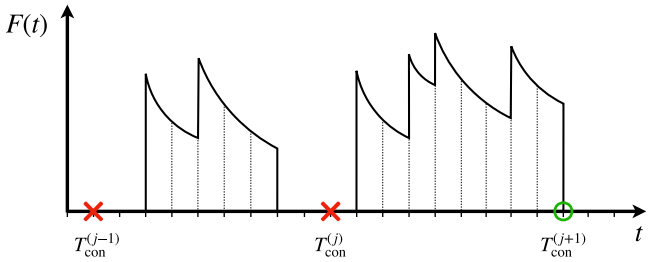

Fig. 2 | Example dynamics of the 1G*n*B system. Here, the fidelity $F(t)$ of the link in the G memory is plotted against time. The vertical lines represent the discretisation of time. The jumps in fidelity occur when the link is purified successfully. In between purifications, the link is subject to decoherence and the fidelity decreases. The link in the G memory is removed due to either failed purification or consumption. When there is no link in memory, $F(t) = 0$. The jth consumption request arrives at time $T_{\text{con}}^{(j)}$. The green tick (red crosses) represent when a consumption request is (is not) served.

Buffering performance

The first step towards the design of useful entanglement buffers is to determine a suitable way to measure performance. Here, we define two performance metrics for entanglement buffers – these quantities were proposed in ref. 14, where they were used to study the 1G1B system. Then, we provide exact, closed-form expressions for these two performance metrics in the 1G*n*B system.

Our first metric is the *availability*. A user is able to consume entanglement only when there is a link available in memory G at the time of requesting the entanglement. Therefore, an important performance measure is the probability that entanglement is available when a consumption request arrives.

Definition 2. (Availability) The availability A is the probability that there is an entangled link present in memory G when a consumption request arrives. This is defined as

$$A = \lim_{m\to\infty} \frac{1}{m} \sum_{j=1}^{m} \mathbb{1}_{\text{link exists}}(T_{\text{con}}^{(j)}), \tag{5}$$

where $T_{\text{con}}^{(j)}$ is the arrival time of the j-th consumption request, and $\mathbb{1}_{\text{link exists}}(t)$ is an indicator function that takes the values one if there is a link stored in memory G at time t, and zero otherwise.

The availability may be seen as a rate metric: it determines the rate at which entanglement can be consumed. The second performance metric is the *average consumed fidelity*, which captures the average quality of consumed entanglement.

Definition 3. (Average consumed fidelity) The average consumed fidelity is the average fidelity of the entangled link upon consumption, conditional on a link being present. More specifically,

$$\bar{F} = \lim_{m\to\infty} \frac{\sum_{j=1}^{m} \mathbb{1}_{\text{link exists}}\left(T_{\text{con}}^{(j)}\right) \cdot F^{-}(T_{\text{con}}^{(j)})}{\sum_{j=1}^{m} \mathbb{1}_{\text{link exists}}(T_{\text{con}}^{(j)})}, \tag{6}$$

where

$$F^{-}(t) = \begin{cases} e^{-\Gamma}\left(F(t-1) - \frac{1}{4}\right) + \frac{1}{4}, & \text{if } F(t-1) > 0, \\ 0, & \text{if } F(t-1) = 0. \end{cases} \tag{7}$$

is the fidelity of the link stored in memory G at the end of the previous timestep at time $t-1$ (and therefore consumed at time t), and $T_{\text{con}}^{(j)}$ is the arrival time of the j-th consumption request.

The indicator function in the numerator of Eq. (6) is included for clarity, but is not necessary: if there is no link in memory at time t, then $F(t) = 0$ by definition.

We note that the Definitions 2 and 3 are presented differently to how they were in ref. 14. This is because the new definitions have a clearer operational meaning, as they are from the viewpoint of the consumer. However, in Supplementary Note A we show that these metrics are equivalent for the 1G*n*B system.

As our first main result, we derive analytical solutions for the availability and the average consumed fidelity in the 1G*n*B system.

Theorem 1. (Formula for the availability) The availability of the 1G*n*B system is given by

$$A = \frac{\mathbb{E}[T_{\text{occ}}]}{\mathbb{E}[T_{\text{gen}}] + \mathbb{E}[T_{\text{occ}}]} \text{ a.s.} \tag{8}$$

where T_{gen} is the time to generate new entangled links, and T_{occ} is the time from when the G memory becomes occupied until it is emptied due to consumption or to failed purification. The expected values are given by

$$\mathbb{E}[T_{\text{gen}}] = \frac{1}{1-(1-p_{\text{gen}})^n} \tag{9}$$

and

$$\mathbb{E}[T_{\text{occ}}] = \frac{1-\widetilde{A}+\widetilde{C}(F_{\text{new}}-\frac{1}{4})}{\left[(1-\widetilde{A})(1-\widetilde{D})-\widetilde{B}\widetilde{C}\right]\widetilde{P}}, \tag{10}$$

with

$$\begin{aligned}
\widetilde{P} &:= p_{\text{con}} + q\left(1-\left(1-p_{\text{gen}}\right)^n\right)(1-p_{\text{con}}), \\
\widetilde{A} &:= \frac{q(1-p_{\text{con}})\widetilde{a}}{e^{\Gamma}-\left(1-q+q\left(1-p_{\text{gen}}\right)^n\right)(1-p_{\text{con}})}, \\
\widetilde{B} &:= \frac{q(1-p_{\text{con}})\widetilde{b}}{p_{\text{con}}+q\left(1-\left(1-p_{\text{gen}}\right)^n\right)(1-p_{\text{con}})}, \\
\widetilde{C} &:= \frac{q(1-p_{\text{con}})\widetilde{c}}{e^{\Gamma}-\left(1-q+q\left(1-p_{\text{gen}}\right)^n\right)(1-p_{\text{con}})}, \\
\widetilde{D} &:= \frac{q(1-p_{\text{con}})\widetilde{d}}{p_{\text{con}}+q\left(1-\left(1-p_{\text{gen}}\right)^n\right)(1-p_{\text{con}})},
\end{aligned}$$

and

$$\begin{aligned}
\widetilde{a} &:= \sum_{k=1}^{n} a_k \cdot \binom{n}{k} (1-p_{\text{gen}})^{n-k} p_{\text{gen}}^k, \\
\widetilde{b} &:= \sum_{k=1}^{n} b_k \cdot \binom{n}{k} (1-p_{\text{gen}})^{n-k} p_{\text{gen}}^k, \\
\widetilde{c} &:= \sum_{k=1}^{n} c_k \cdot \binom{n}{k} (1-p_{\text{gen}})^{n-k} p_{\text{gen}}^k, \\
\widetilde{d} &:= \sum_{k=1}^{n} d_k \cdot \binom{n}{k} (1-p_{\text{gen}})^{n-k} p_{\text{gen}}^k.
\end{aligned}$$

Proof. See Supplementary Note B.

From Theorem 1, we see that the availability depends on all the parameters of the system (listed in Table 1), including the noise level Γ. The latter may come as a surprise, since one would expect noise to have an impact on the average consumed fidelity, but maybe not on the availability, which is only affected by processes that fill or deplete the G memory. These processes are entanglement generation, failed purification, and

consumption. In our model, the probability of failed purification depends on Eq. (4) on the fidelity of the buffered link, which is in turn affected by the level of noise. As a consequence, noise has an indirect effect on the availability.

Theorem 2. (Formula for the average consumed fidelity) The average consumed fidelity of the 1GnB system is given by

$$\overline{F} = \frac{w\widetilde{F}_{\text{new}} + \widetilde{x}}{y\widetilde{F}_{\text{new}} + \widetilde{z}} \text{ a.s.} \tag{11}$$

with

$$\begin{aligned}
\widetilde{w} &:= p_{\text{con}} + q(1-p_{\text{con}})\left(p^*_{\text{gen}} + \tfrac{1}{4}\widetilde{c} - \widetilde{d}\right),\\
\widetilde{x} &:= \tfrac{1}{4}\left[e^{\Gamma} - 1 + q(1-p_{\text{con}})\left(-\widetilde{a} + 4\widetilde{b} - \tfrac{1}{4}\widetilde{c} + \widetilde{d}\right)\right],\\
\widetilde{y} &:= q(1-p_{\text{con}})\widetilde{c},\\
\widetilde{z} &:= e^{\Gamma} - 1 + p_{\text{con}} + q(1-p_{\text{con}})\left(p^*_{\text{gen}} - \widetilde{a} - \tfrac{1}{4}\widetilde{c}\right),
\end{aligned}$$

where $p^*_{\text{gen}} = 1-(1-p_{\text{gen}})^n$, and $\widetilde{a}$, $\widetilde{b}$, $\widetilde{c}$, and $\widetilde{d}$ are given in Theorem 1.

Proof. See Supplementary Note B.

We note that both A and $\overline{F}$ have been defined as random variables in Definitions 2 and 3. However, as shown in Theorems 1 and 2, these quantities are almost surely deterministic functions of the system parameters. For clarity and convenience, we will adopt a slight abuse of notation and treat A and $\overline{F}$ as deterministic functions. This convention will be maintained throughout the remainder of the text.

Monotonic performance

Each time a B memory successfully generates entanglement, there is the opportunity to purify the buffered link. This is controlled by the parameter q, which is the probability that, after some fresh links are successfully generated, they are used to attempt purification (otherwise they are discarded). If purification is never attempted ($q=0$), the fidelity of the buffered link will never be increased, although the buffered link will never be lost to failed purification. If purification is always attempted ($q=1$), the availability and average consumed fidelity might be affected as follows:

- Purifying more often means risking the loss of buffered entanglement more frequently, since purification can fail. This suggests availability may be decreasing in q. However, many purification protocols have a probability of success that is increasing in the fidelity of the buffered link, F. This means that, when purification is applied more frequently to maintain a high-fidelity link, subsequent purification attempts are more likely to succeed. Consequently, it is not clear that the availability is decreasing in q.
- The fidelity of the buffered link increases after applying several purification rounds. However, if purification is applied too greedily, we may lose a high-quality link, and we would have to restart the system with a lower-quality link. If a consumption request then arrives, it would only be able to consume low-quality entanglement. Hence, it is not clear that the average consumed fidelity is increasing in q.In the following, we address the previous discussion and show that, if purification is always attempted ($q=1$), the availability is actually minimised, while the average consumed fidelity is maximised. More generally, we show that A and $\overline{F}$ are both monotonic in q, given some reasonable conditions on the jump functions J_k. The following results (Propositions 1 and 2) may be used to answer an important question about the 1GnB system: *how frequently should we purify the buffered state in order to maximise A (or $\overline{F}$)?* That is, *what value of q optimises our performance metrics?*

Proposition 1. The availability is a non-increasing function of q, i.e.

$$\frac{\partial A}{\partial q} \leq 0. \tag{12}$$

Proof. See Supplementary Note E.

As previously explained, the monotonicity of the availability in q is not a trivial result, and it has fundamental implications. It allows us to derive upper and lower bounds that apply to 1GnB systems using *any* purification policy.

Corollary 1. The availability is bounded as

$$\frac{p^*_{\text{gen}}\cdot(\gamma + p_{\text{con}})}{\xi + \xi'\cdot p^*_{\text{gen}} + \xi''\cdot (p^*_{\text{gen}})^2} \leq A \leq \frac{p^*_{\text{gen}}}{p^*_{\text{gen}} + p_{\text{con}}}, \tag{13}$$

with $p^*_{\text{gen}} := 1-(1-p_{\text{gen}})^n$, $\gamma := e^{\Gamma}-1$, $\xi := \gamma p_{\text{con}} + p^2_{\text{con}}$, $\xi' := 1 + 2\gamma + (2-\gamma)p_{\text{con}} - 2p^2_{\text{con}}$, and $\xi'' := 2(1-p_{\text{con}})^2$. Moreover, the upper bound is tight, and for any purification policy is achieved when $q=0$.

Proof. See Supplementary Note E.

We refer to p^*_{gen} as the *effective generation probability*, since it is the probability that at least one new link is generated in a single (multiplexed) attempt.

The upper bound from Eq. (13) only depends on the effective generation probability and the probability of consumption. This bound is achievable with any purification policy: to maximise the availability, it suffices to never purify ($q=0$). A special case is deterministic policies (those with $p_k(F)=1, \forall k$), which achieve this bound for any q. This upper bound coincides with the tight upper bound found in previous work for a 1G1B system[14]. Note that the 1G1B analysis from ref. 14 was done in continuous time, where rates were used instead of probabilities. In this framework, the maximum availability was $\lambda/(\lambda+\mu)$, where λ was the (non-multiplexed) entanglement generation rate, and μ was the consumption rate.

Unlike the upper bound, we note that the lower bound from Eq. (13) has not yet shown to be tight. We believe that the availability at $q=1$ of a policy that always fails purification ($c_k = d_k = 0, \forall k$) constitutes a tight lower bound for any other purification policy. We leave this proof as future work.

Figure 3 shows the upper and lower bounds for the availability from Eq. (13) versus p^*_{gen} for two different noise levels. As discussed, only the lower bound is affected by noise. In particular, we have observed that the gap between the bounds is reduced when the noise level increases. Another remarkable feature is that, when p^*_{gen} approaches zero, both upper and lower bounds are equal to $p^*_{\text{gen}}/p_{\text{con}}$ to first order in p_{gen}. Hence, in the limit of small effective generation probabilities, the availability also satisfies

$$A \approx \frac{p^*_{\text{gen}}}{p_{\text{con}}}. \tag{14}$$

Proposition 2. The average consumed fidelity is a non-decreasing function of q, i.e.,

$$\frac{\partial \overline{F}}{\partial q} \geq 0, \tag{15}$$

if $J_k(F_{\text{new}}) \geq F_{\text{new}}$, $\forall k \in \mathbb{N}$.

Proof. See Supplementary Note F.

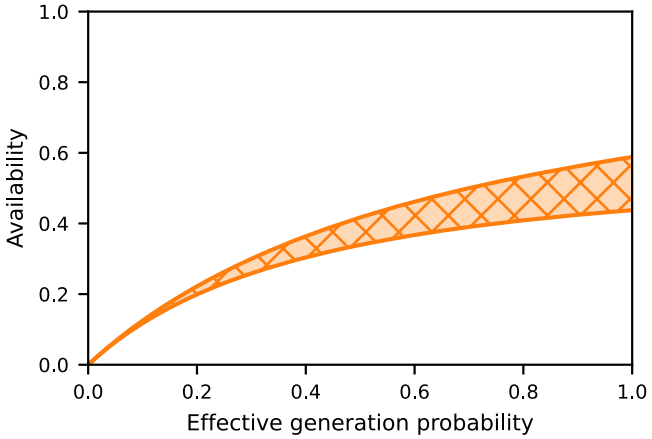

Fig. 3 | The upper bound on the availability is tight, and it converges to the lower bound in the limit of small generation probabilities. Upper and lower bounds on the availability from Eq. (13), versus the effective generation probability $p^*_{\text{gen}} = 1-(1-p_{\text{gen}})^n$. The availability can only take values within the shaded region. In this example, we use $\Gamma = 1$ and $p_{\text{con}} = 0.7$.

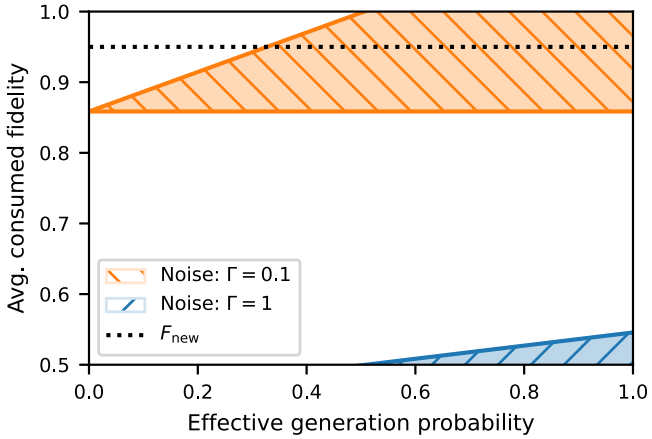

Fig. 4 | The upper bound on the average consumed fidelity marks unachievable values for any purification policy. Upper and lower bounds on the average consumed fidelity $\overline{F}$ from Eq. (16), versus the effective generation probability $p^*_{\text{gen}} = 1-(1-p_{\text{gen}})^n$. $\overline{F}$ can only take values within the shaded region. In this example, we use $p_{\text{con}} = 0.7$.

As previously explained, the monotonicity of $\overline{F}$ in q is not a trivial result. In fact, this behaviour is only certain for purification policies composed of protocols that can increase the fidelity of a newly generated link. That is, when k new links are generated, the protocol applied satisfies $J_k(F_{\text{new}}) \geq F_{\text{new}}$. This is a reasonable condition: otherwise, we would be applying purification protocols that decrease the fidelity of new links.

Proposition 2 also allows us to derive useful upper and lower bounds for $\overline{F}$ that apply to 1GnB systems using any purification policy.

Corollary 2. The average consumed fidelity is bounded as

$$\frac{\gamma + 4F_{\text{new}}p_{\text{con}}}{4\gamma + 4p_{\text{con}}} \leq \overline{F} \leq \frac{\gamma + 4F_{\text{new}}p_{\text{con}} + 3(1-p_{\text{con}})p^*_{\text{gen}}}{4\gamma + 4p_{\text{con}}}, \tag{16}$$

with $\gamma := e^{\Gamma} - 1$. Moreover, the lower bound is tight, and for any purification policy is achieved when $q = 0$.

Proof. See Supplementary Note F.

We see that the tight lower bound from Eq. (16) does not depend on the number of memories n, the probability of successful entanglement generation p_{gen}, or the purification policy. This is because this bound corresponds to $q = 0$. In such a case, no purification is applied, and the consumed fidelity only depends on the initial fidelity (F_{new}) and the amount of decoherence experienced until consumption (given by Γ and p_{con}).

The bounds on $\overline{F}$ can be used to determine if the parameters of the system need improvement to meet specific quality-of-service requirements. For example, let us consider Fig. 4, which shows the bounds for $p_{\text{con}} = 0.7$ and two different values of Γ. If noise is strong ($\Gamma = 1$ in this example), we observe that values of p^*_{gen} below 0.5 yield $\overline{F} < 1/2$, which means that, on average, the consumed link will not be entangled[14]. Hence, if the consumption request rate is $p_{\text{con}} = 0.7$, we need to increase p^*_{gen} beyond 0.5 (by increasing the number of B memories, n, or the probability of successful entanglement generation, p_{gen}) or to decrease the noise experienced in memory G in order to provide a useful average state. When the noise level is $\Gamma = 0.1$, Fig. 4 shows that $\overline{F} > 0.85$. Moreover, for $p^*_{\text{gen}} > 0.3$, the upper bound is above F_{new}, which means that a smart choice of purification policy may allow us to buffer entanglement with $\overline{F} > F_{\text{new}}$. Ultimately, this means that, in this regime, an entanglement buffer with faulty memories may be able to keep entanglement at higher fidelities than a perfect memory.

Choosing a purification policy

In previous studies of entanglement buffering, the choice of purification policy was restricted by the properties of the system. For example, in ref. 14, the 1G1B system was studied, where only 2-to-1 purification protocols can be implemented, and the jump function was assumed to be linear in the fidelity of the buffered link. Other works include simplifying assumptions (e.g., in ref. 15, a buffer is studied that employs the purification protocol proposed in ref. 46). The 1GnB buffering system offers more freedom in the choice of purification protocols. In a 1GnB buffer, each entanglement generation attempt is multiplexed and can generate up to n new links at a time. When $k \leq n$, new links are produced; any $(k+1)$-to-1 purification protocol can, in principle, be implemented. This provides an extra knob that can be used to tune the performance of the system to the desired values. Next, we investigate the impact that specific purification policies have on the system, and we provide guidelines on how to choose a suitable purification policy. Note that an exhaustive optimisation problem would be extremely computationally expensive to solve due to the large space of purification policies – optimising over a_k, b_k, c_k, d_k is not easy, since it is not certain that every combination of those parameters corresponds to an implementable purification circuit.

There are two trivial deterministic policies ($p_k = 1, \forall\, k$) that we will use as a baseline:

- In the *identity policy*, the system does not perform any operation on the buffered link, which yields an output fidelity $J_k(F) = F, \forall\, k > 0$. This is equivalent to setting $q = 0$. As discussed in "Monotonic performance", the identity policy therefore maximises the availability and minimises the average consumed fidelity.
- In the *replacement policy*, the system replaces the buffered entangled link by a new link, yielding an output fidelity $J_k(F) = F_{\text{new}}, \forall\, k > 0$. This corresponds to $a_k = 0$, $b_k = F_{\text{new}} - 1/4$, $c_k = 0$, and $d_k = 1$. Since this policy is deterministic, from the discussion in the section titled Monotonic performance, we find that the replacement policy also provides maximum availability for any value of q. Since $\overline{F}$ is maximized for $q = 1$ (Proposition 2), we will only consider a replacement policy that always chooses to replace the link in memory when a new link is generated. That is, the replacement policy implicitly assumes $q = 1$.

Another simple strategy is the *DEJMPS policy*. This policy consists in applying the well-known 2-to-1 DEJMPS purification protocol[17] using the buffered link and a newly generated link as inputs. If more than one link is successfully generated, we use only one of them and discard the rest. We provide the purification coefficients a_k, b_k, c_k, and d_k for this policy in Supplementary Note C.1. One of the main drawbacks of the DEJMPS policy is that it does not take full advantage of the multiplexed entanglement generation, as it only uses one of the newly generated links and discards the rest. A technique that could improve the performance of the policy is

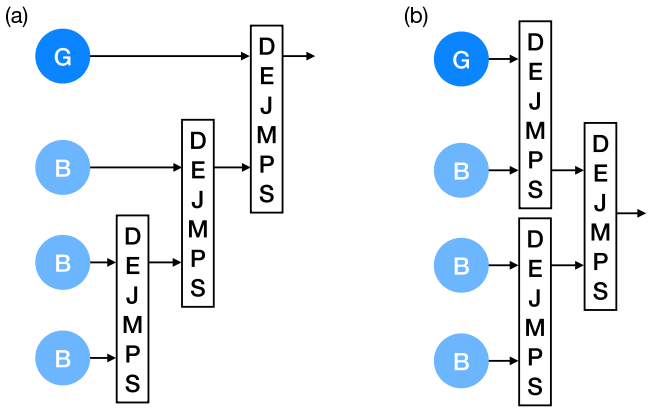

Fig. 5 | The ordering in a concatenated policy matters. Example of two different orderings when the buffered link (G) and three newly generated links (B) are used. We call ordering **a** "concatenated DEJMPS". Ordering **b** is often called "nested"[53].

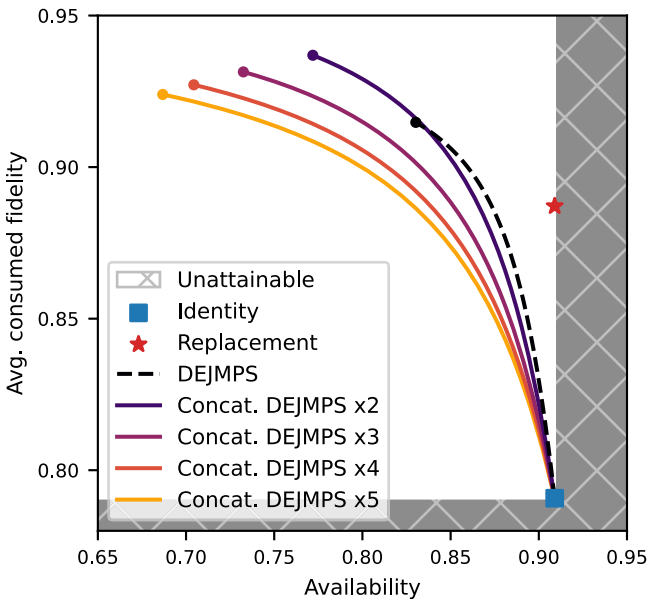

Fig. 6 | Concatenating simple purification policies decreases A but may increase $\overline{F}$. Performance of 1GnB systems with different purification policies, in terms of availability A and average consumed fidelity $\overline{F}$. The shaded area corresponds to unattainable values of A and $\overline{F}$ (see Eqs. (13) and (16)). Lines and markers show the combinations of A and $\overline{F}$ achievable by different purification policies: identity (square marker), replacement (star marker), DEJMPS (dashed line), and concatenated DEJMPS (solid lines). Concatenation can boost $\overline{F}$ (e.g., the maximum $\overline{F}$ of twice-concatenated DEJMPS is larger than DEJMPS), but excessive concatenation may eventually lead to a drop in $\overline{F}$. Parameter values used in this example: $n = 10$, $p_{\text{gen}} = 0.5$, ρ_{new} is a Werner state with $F_{\text{new}} = 0.9$, $p_{\text{con}} = 0.1$, and $\Gamma = 0.02$.

concatenation, which consists in applying DEJMPS to all links (the buffered one and the newly generated ones) consecutively until only one link remains, which will be stored in memory G. Note that the concatenation of DEJMPS subroutines can be applied using different orders of the links (see Fig. 5). The order determines the output fidelity and probability of success[47], which affects the performance of the buffering system. In what follows, we consider the *concatenated DEJMPS* policy, where DEJMPS is applied sequentially to all the newly generated links and the buffered link is used in the last application of DEJMPS, as in Fig. 5a. In our analysis, we found that different orderings provided qualitatively similar behaviour of our two performance metrics (see Supplementary Note G.1 for further details).

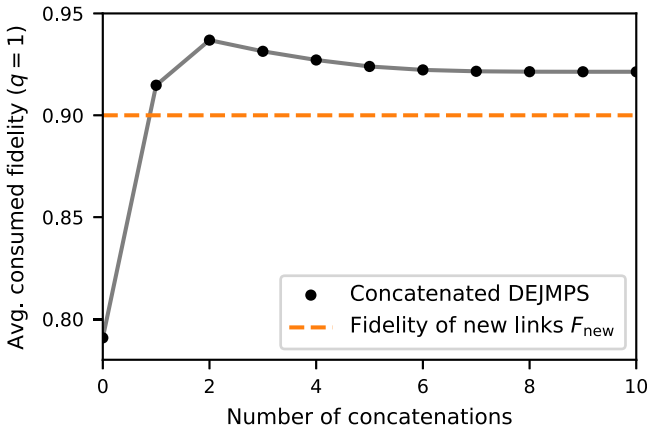

Fig. 7 | Excessive concatenation worsens the performance. The maximum average consumed fidelity $\overline{F}$ is achieved by a purification policy that concatenates DEJMPS a limited number of times. Zero concatenations corresponds to an identity policy (no purification is performed). One concatenation corresponds to the DEJMPS policy. Excessive concatenation may decrease $\overline{F}$. Parameter values used in this example: $n = 10$, $p_{\text{gen}} = 0.5$, ρ_{new} is a Werner state with $F_{\text{new}} = 0.9$, $p_{\text{con}} = 0.1$, and $\Gamma = 0.02$.

Figure 6 shows the performance of several policies: identity, replacement, DEJMPS, and concatenated DEJMPS × N. The latter is a policy that applies DEJMPS sequentially up to N times and discards any extra links: if $k \leq N$ links are generated, then k concatenations are performed, and if $k > N$ links are generated, N concatenations are performed. We note that concatenated DEJMPS × 1 is just the same as the DEJMPS policy. DEJMPS and concatenated DEJMPS are plotted for $q \in [0, 1]$. The maximum average consumed fidelity is indicated with a dot, and it is achieved when $q = 1$. The first observation from this figure is that a higher level of concatenation decreases the availability. This is because it requires multiple DEJMPS subroutines to succeed, which decreases the overall probability of successful purification. However, a higher level of concatenation can significantly increase the average consumed fidelity $\overline{F}$. For example, the maximum $\overline{F}$ that DEJMPS can achieve is 0.915, while concatenated DEJMPS × 2 leads to $\overline{F} = 0.937$ (for $q = 1$). Nevertheless, for the parameter values explored, we also find that increasing the number of concatenations beyond two often reduces both A and $\overline{F}$. This behaviour is shown more explicitly in Fig. 7, where we plot the maximum $\overline{F}$ versus the maximum number of concatenations N. In this example, the number of B memories is $n = 10$, and therefore it is only possible to perform up to 10 concatenated applications of DEJMPS. We observe that $\overline{F}$ is maximized for two concatenations. The same was observed for different parameter values – in some edge cases, $\overline{F}$ increases with more concatenations, although the increase is marginal (see Supplementary Note G.2 for further details). In conclusion, this result shows that even if many new links are successfully generated in parallel, it can sometimes be beneficial to use only one or two of them for purification while discarding the rest. A follow-up question arises: *what if we employ more sophisticated (k+1)-to-1 protocols instead of simply concatenating 2-to-1 protocols? Can we then improve the performance of the buffer?* This is the question that we explore next.

Much recent work has focused on the search for optimal purification protocols[36,48,49], where optimal protocols are typically defined as those which maximise the output fidelity, or in some cases the success probability. Here, we evaluate the performance of a 1GnB system with some of these protocols, and we find a surprising result: simple protocols like DEJMPS can vastly outperform these more complex protocols in terms of buffering performance. In particular, we consider the bilocal Clifford protocols that maximise the output fidelity, given in ref. 36. We refer to this policy as the *optimal bilocal Clifford (optimal-bC) policy*. In Supplementary Note C.2, we discuss the details of this policy and provide its purification coefficients a_k, b_k, c_k, and d_k.

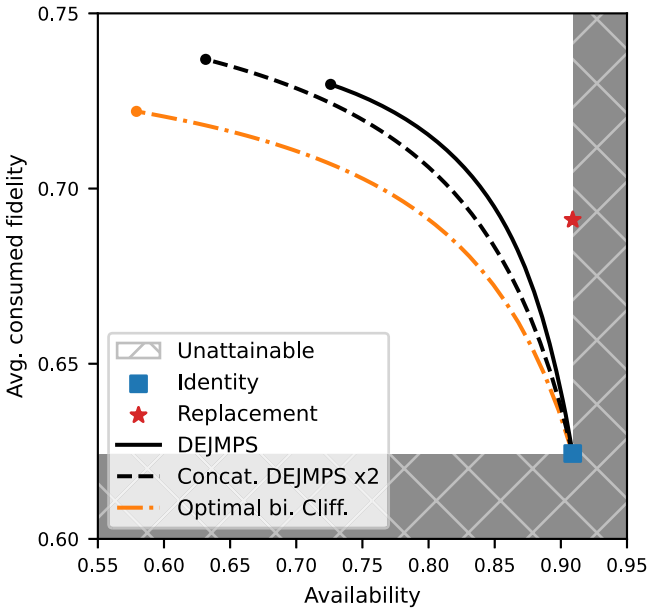

Fig. 8 | Simple policies perform better despite discarding freshly generated entanglement. Performance of 1G*n*B systems with different purification policies, in terms of availability A and average consumed fidelity $\overline{F}$. The shaded area corresponds to unattainable values of A and $\overline{F}$ (see Eqs. (13) and (16)). Lines and markers show the combinations of A and $\overline{F}$ achievable by different purification policies: identity (square marker), replacement (star marker), DEJMPS (solid line), twice-concatenated DEJMPS (dashed line), and optimal-bC (dotted line). Parameter values used in this example: $n = 5$, $p_{\text{gen}} = 0.8$, ρ_{new} is a Werner state with $F_{\text{new}} = 0.7$, $p_{\text{con}} = 0.1$, and $\Gamma = 0.02$.

Figure 8 shows the performance of the optimal-bC policy in comparison to DEJMPS and twice-concatenated DEJMPS. The optimal-bC policy provides a significantly lower availability, A, without providing any advantage in average consumed fidelity, $\overline{F}$. In other words, for any desired A, using DEJMPS or twice-concatenated DEJMPS always provides a larger $\overline{F}$ than the optimal-bC policy. If we want to increase A as much as possible, the replacement policy is better than any other, as discussed earlier. We say that the performance of DEJMPS, twice-concatenated DEJMPS, and replacement forms the *Pareto frontier*[50], which informally is the set of best achievable values for A and $\overline{F}$ for this collection of protocols. We tested different parameter combinations and found that the Pareto frontier was often made of DEJMPS, concatenated DEJMPS, and replacement. The reason for these simple policies to outperform the optimal-bC policy is that the optimal bilocal Clifford protocols maximise the output fidelity at the expense of a reduced probability of success. At some point, the sacrifice in the probability of success can outweigh the benefit of a larger output fidelity, thereby reducing the overall performance of the buffer in terms of both A and $\overline{F}$.

Our comparison between simple and optimal purification protocols is by no means an exhaustive study. However, it shows that purification protocols that maximise only the output fidelity (or probability of success) must not be blindly used in more complex systems involving many impacting factors, such as decoherence and consumption, such as entanglement buffers. In fact, we find that discarding some of the newly generated links and applying a 2-to-1 protocol can provide larger A and $\overline{F}$ than using all of the links in a more sophisticated purification subroutine. Note that this does not mean that multiplexed entanglement generation is not useful: even if we only employ 2-to-1 protocols, multiplexing boosts the effective entanglement generation rate, which allows for a more frequent purification of the buffered link.

Additionally, we also tested other complex policies that use (sub-optimal) k-to-1 protocols, such as the *513 EC policy*, which uses a 5-to-1 protocol based on a [[5, 1, 3]] quantum error correcting code. In Supplementary Note D, we explain this policy in detail and show that it can outperform DEJMPS and twice-concatenated DEJMPS in some parameter regions.

As discussed above, concatenating protocols multiple times does not necessarily improve the performance of the buffer (neither in terms of A nor $\overline{F}$). The reason is that, when concatenating, a single failure in one of the purification subroutines (in our examples, DEJMPS) leads to failure of the whole concatenated protocol. This can be easily solved: instead of considering the concatenated protocol as a black box that only succeeds when all subroutines succeed, *what if we condition the execution of each subroutine on the success/failure of previous subroutines?*

Consider for example, the concatenated protocol from Fig. 5a. If any of the DEJMPS subroutines fail, the whole protocol fails, and the buffered link has to be discarded. However, we can fix this by raising a failure flag whenever any of the first two subroutines fails. If this flag is raised, the third subroutine is not executed, and we leave the buffered link untouched. The flagged version of a concatenated protocol has a larger probability of success, but can also have a lower output fidelity. This means that it is not clear a priori what is the impact of flags is on the buffer performance. We now analyse a simple case in which we conclude that flags can be either beneficial or detrimental depending on the values of system parameters such as the level of noise Γ, and not only on the purification policy itself.

Let us consider a policy that operates as follows. For simplicity, we assume that newly generated states ρ_{new} are Werner states with fidelity F_{new}. When k new links are generated and there is already a link stored in memory G:

1. If $k = 1$, we apply the replacement protocol, which has coefficients $a_1 = 0$, $b_1 = F_{\text{new}} - 1/4$, $c_1 = 0$, and $d_1 = 1$.
2. If $k \geq 2$, we apply the DEJMPS protocol to two of the fresh links and discard the rest. Then, we replace the link in memory with the output from the DEJMPS subroutine, without checking whether it was successful or not. This means that the output fidelity of the protocol is the same as the output fidelity from the DEJMPS subroutine. Since replacement is deterministic, the success probability of this protocol is also the same as the success probability of the DEJMPS subroutine. The purification coefficients for $k \geq 2$ are therefore given by $a_k = 0$, $b_k = a(\rho_{\text{new}})\left(F_{\text{new}} - 1/4\right) + b(\rho_{\text{new}})$, $c_k = 0$, and $d_k = c(\rho_{\text{new}})\left(F_{\text{new}} - 1/4\right) + d(\rho_{\text{new}})$, where a, b, c, and d are the coefficients of the DEJMPS protocol (given in Supplementary Note C.1).

Now, let us consider a flagged variant of the previous policy, with coefficients a'_k, b'_k, c'_k, and d'_k. It works as follows:

1. When $k = 1$, we apply the replacement protocol.
2. When $k \geq 2$ links are generated, the DEJMPS protocol is applied to two of the fresh links, and the rest are discarded. Then, the link in memory is replaced with the output from the DEJMPS subroutine, but only if the subroutine succeeds (otherwise, the buffered link is left untouched). This protocol is now fully deterministic, since the buffered link is never removed from memory. Consequently, $c'_k = 0$, and $d'_k = 1$. The output fidelity of this protocol can be computed as the weighted average of the original fidelity of the link in memory and the output fidelity of the DEJMPS subroutine – the first term must be weighted by the probability of failure of the subroutine, and the second term by the probability of success. Then, the remaining purification coefficients can be computed as $a'_k = 1 - c(\rho_{\text{new}})\left(F_{\text{new}} - 1/4\right) - d(\rho_{\text{new}})$ and $b'_k = a(\rho_{\text{new}})(F_{\text{new}} - 1/4) + b(\rho_{\text{new}})$, where a, b, c, and d are the coefficients of the DEJMPS protocol (given in Supplementary Note C.1).

By introducing the flags, we have created a protocol with probability of success $p'_k = 1 \geq p_k$, where p_k is the probability of success of the original protocol. However, it can be shown that the output fidelity of the flagged protocol is $J'_k(F) \leq J_k(F)$, where J_k is the jump function of the original protocol. This holds when DEJMPS can improve the fidelity of the newly generated links, i.e., when $J(F_{\text{new}}) \geq F_{\text{new}}$, where J is the jump function of

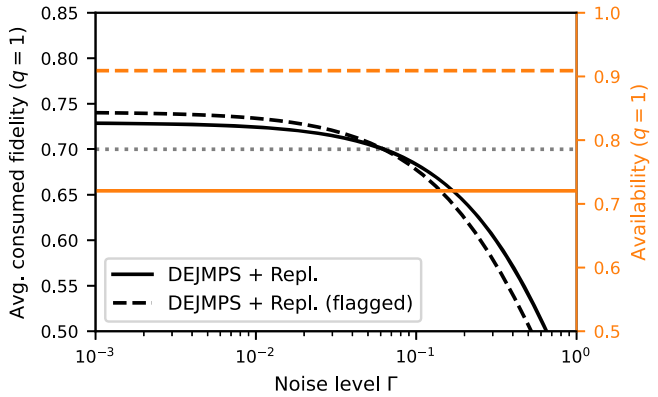

Fig. 9 | Flagged protocols boost the availability but may decrease the average consumed fidelity. Availability A and average consumed fidelity $\overline{F}$ versus the noise level Γ, for a 'DEJMPS + Replacement' policy and its flagged version. In the first policy, the buffered link is lost when a DEJMPS subroutine fails. The second policy incorporates a flag that prevents this from happening -- it succeeds deterministically at the expense of a lower output fidelity. The flagged policy yields larger A, but may decrease $\overline{F}$ in some parameter regimes (e.g., when Γ is large). Parameter values used in this example: $n = 2$, $p_{\text{gen}} = 1$, ρ_{new} is a Werner state[54] with $F_{\text{new}} = 0.7$, and $p_{\text{con}} = 0.1$.

DEJMPS. The opposite regime is not interesting, since DEJMPS is decreasing the fidelity of the link, and we would be better off not purifying.

As shown in the previous example, internal flags increase the probability of success of purification protocols, which should boost the availability of the buffer. However, flags may have the side effect of reducing the output fidelity, and therefore, it is not clear what their impact is on the average consumed fidelity. In Fig. 9, we show the performance of a 1GnB system using the policy described above, versus the level of noise in memory G. We show A (orange lines) and $\overline{F}$ (black lines) for the original policy (solid lines) and the flagged policy (dashed lines). As expected, the availability is larger for the flagged policy. The behaviour of $\overline{F}$ is more interesting. When the level of noise is low, the flagged policy provides better performance, since it prevents high-quality entanglement from being lost to a failed purification. However, when noise is strong, flagging becomes detrimental in terms of $\overline{F}$: the buffer is likely to store low-quality entanglement due to the strong noise, and flags prevent the buffered link from being discarded earlier due to failed purification and being replaced by a fresh link. Note that other strategies, such as using the output state regardless of the success or failure flag[51] and using hyperentangled states[52], can also be employed for designing deterministic purification protocols.

In conclusion, internal flags are a solid tool to improve the availability of entanglement buffers based on concatenated purification protocols. However, they can decrease the average consumed fidelity in some parameter regimes. Hence, flagged purification policies should not be assumed to be better than their non-flagged counterparts, and their performance should be carefully evaluated before being adopted.

Discussion

In this paper, we have studied the behaviour of entanglement buffers with one long-lived memory and n short-lived memories (1GnB system). In particular, we have provided analytical expressions for the two main performance measures: the availability and the average consumed fidelity. These expressions provide valuable insights, such as the fundamental limits to the performance of 1GnB systems discussed earlier.

Since our analytical solutions are not computationally expensive to evaluate, we expect our buffering setup to be easy to incorporate in the study of more complex network architectures, such as quantum repeater chains or even large-scale quantum networks. Additionally, larger buffering systems with multiple long-lived memories, e.g., an mGnB setup, can be implemented with multiple 1GnB systems in parallel.

Due to the vast freedom in the choice of purification policy, there are multiple ways in which our analysis of purification strategies for entanglement buffers can be extended. Notably, determining the optimal ordering in which simple protocols should be applied to newly generated links (e.g., concatenated, nested[53], or banded[47]) is left as future work. Additionally, finding policies that optimize availability or average consumed fidelity remains an important open question.

Methods

Analytical framework

We study a two-node entanglement buffer, denoted as 1GnB, in which each node possesses one long-lived ("good") quantum memory and n short-lived ("bad") memories. The system operates in discrete time steps. In each time slot, entanglement generation is attempted simultaneously in all n bad memories, each succeeding with probability p_{gen}. The good memory stores a bipartite Werner state with fidelity F that decoheres exponentially at rate Γ. When new links are generated, they may be used to purify the stored state with probability q; otherwise, they are discarded. Purification protocols can be modelled analytically only with four coefficients, as in Eqs. (3) and (4)– further details on how to find these coefficients are given in Supplementary Note C. Lastly, a consumption request arrives with probability p_{con} per time step and, if entanglement is available, the stored link is consumed and removed from memory.

Performance metrics

We employ tools from renewal theory to study the evolution of F over time and to characterize the performance of the buffer in terms of the availability A and the average consumed fidelity $\overline{F}$ (see Definitions 2 and 3; alternative but equivalent definitions are discussed in Supplementary Note A). Closed-form expressions for A and $\overline{F}$ are given in Theorems 1 and 2, which are derived in Supplementary Note B.

Data availability

No data is needed to reproduce our results, since all results in this manuscript are analytical. The code used to perform the analysis and generate all the plots shown in this paper can be found in the following GitHub repository: https://github.com/AlvaroGI/buffering-1GnB. This repository also includes a discrete-event simulator of a 1GnB system that we used to validate our analytical results.

Code availability

The code used to perform the analysis and generate all the plots shown in this paper can be found in the following GitHub repository: https://github.com/AlvaroGI/buffering-1GnB. This repository also includes a discrete-event simulator of a 1GnB system that we used to validate our analytical results.

Received: 22 April 2025; Accepted: 4 December 2025;
Published online: 20 February 2026

References

1. Ekert, A. K. Quantum cryptography based on Bell's theorem. *Phys. Rev. Lett.* **67**, 661 (1991).
2. Bennett, C. H., Brassard, G. & Mermin, N. D. Quantum cryptography without Bell's theorem. *Phys. Rev. Lett.* **68**, 557 (1992).
3. Lloyd, S. Enhanced sensitivity of photodetection via quantum illumination. *Science* **321**, 1463–1465 (2008).
4. Qian, K. et al. Heisenberg-scaling measurement protocol for analytic functions with quantum sensor networks. *Phys. Rev. A* **100**, 042304 (2019).
5. England, D. G., Balaji, B. & Sussman, B. J. Quantum-enhanced standoff detection using correlated photon pairs. *Phys. Rev. A* **99**, 023828 (2019).

6. Wu, B.-H., Guha, S. & Zhuang, Q. Entanglement-assisted multi-aperture pulse-compression radar for angle resolving detection. *Quantum Sci. Tech.* **8**, 035016 (2023).
7. Brassard, G., Broadbent, A. & Tapp, A. Quantum pseudo-telepathy. *Found. Phys.* **35**, 1877–1907 (2005).
8. Broadbent, A. & Tapp, A. Can quantum mechanics help distributed computing?. *ACM SIGACT News* **39**, 67–76 (2008).
9. Chakraborty, K., Rozpedek, F., Dahlberg, A. & Wehner, S. Distributed routing in a quantum internet. *arXiv preprint arXiv:1907.11630* (2019).
10. Ghaderibaneh, M., Gupta, H., Ramakrishnan, C. & Luo, E. Pre-distribution of entanglements in quantum networks. In *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 426–436 (IEEE, 2022).
11. Pouryousef, S., Panigrahy, N. K. & Towsley, D. A quantum overlay network for efficient entanglement distribution. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*, 1–10 (IEEE, 2023).
12. Iñesta, ÁG. & Wehner, S. Performance metrics for the continuous distribution of entanglement in multiuser quantum networks. *Phys. Rev. A* **108**, 052615 (2023).
13. Askarani, M. F., Chakraborty, K. & Do Amaral, G. C. Entanglement distribution in multi-platform buffered-router-assisted frequency-multiplexed automated repeater chains. *New J. Phys.* **23**, 063078 (2021).
14. Davies, B., Iñesta, ÁG. & Wehner, S. Entanglement buffering with two quantum memories. *Quantum* **8**, 1458 (2024).
15. Elsayed, K. S., KhudaBukhsh, W. R. & Rizk, A. On the fidelity distribution of purified link-level entanglements. In *ICC 2024-IEEE International Conference on Communications*, 485–490 (IEEE, 2024).
16. Bennett, C. H. et al. Purification of noisy entanglement and faithful teleportation via noisy channels. *Phys. Rev. Lett.* **76**, 722 (1996).
17. Deutsch, D. et al. Quantum privacy amplification and the security of quantum cryptography over noisy channels. *Phys. Rev. Lett.* **77**, 2818 (1996).
18. Dür, W., Briegel, H.-J., Cirac, J. I. & Zoller, P. Quantum repeaters based on entanglement purification. *Phys. Rev. A* **59**, 169 (1999).
19. Yan, P.-S., Zhou, L., Zhong, W. & Sheng, Y.-B. Advances in quantum entanglement purification. *Sci. China Phys. Mech. Astron.* **66**, 250301 (2023).
20. Haldar, S. et al. Reducing classical communication costs in multiplexed quantum repeaters using hardware-aware quasi-local policies. *Commun. Phys.* **8**, 132 (2025).
21. Victora, M. et al. Entanglement purification on quantum networks. *Phys. Rev. Res.* **5**, 033171 (2023).
22. Bradley, C. E. et al. A ten-qubit solid-state spin register with quantum memory up to one minute. *Phys. Rev. X* **9**, 031045 (2019).
23. Abobeih, M. H. et al. One-second coherence for a single electron spin coupled to a multi-qubit nuclear-spin environment. *Nat. Commun.* **9**, 2552 (2018).
24. Pompili, M. et al. Realization of a multinode quantum network of remote solid-state qubits. *Science* **372**, 259–264 (2021).
25. Wengerowsky, S., Joshi, S. K., Steinlechner, F., Hübel, H. & Ursin, R. An entanglement-based wavelength-multiplexed quantum communication network. *Nature* **564**, 225–228 (2018).
26. Chen, K. C. et al. Zero-added-loss entangled-photon multiplexing for ground-and space-based quantum networks. *Phys. Rev. Appl.* **19**, 054029 (2023).
27. Mower, J. & Englund, D. Efficient generation of single and entangled photons on a silicon photonic integrated chip. *Phys. Rev. A* **84**, 052326 (2011).
28. Krutyanskiy, V., Canteri, M., Meraner, M., Krcmarsky, V. & Lanyon, B. Multimode ion-photon entanglement over 101 kilometers. *PRX Quantum* **5**, 020308 (2024).
29. Collins, O., Jenkins, S., Kuzmich, A. & Kennedy, T. Multiplexed memory-insensitive quantum repeaters. *Phys. Rev. Lett.* **98**, 060502 (2007).
30. Munro, W., Harrison, K., Stephens, A., Devitt, S. & Nemoto, K. From quantum multiplexing to high-performance quantum networking. *Nat. Photonics* **4**, 792–796 (2010).
31. van Dam, S. B., Humphreys, P. C., Rozpedek, F., Wehner, S. & Hanson, R. Multiplexed entanglement generation over quantum networks using multi-qubit nodes. *Quantum Sci. Tech.* **2**, 034002 (2017).
32. Barrett, S. D. & Kok, P. Efficient high-fidelity quantum computation using matter qubits and linear optics. *Phys. Rev. A* **71**, 060310 (2005).
33. Togan, E. et al. Quantum entanglement between an optical photon and a solid-state spin qubit. *Nature* **466**, 730–734 (2010).
34. Bernien, H. et al. Heralded entanglement between solid-state qubits separated by three metres. *Nature* **497**, 86–90 (2013).
35. Zhou, Y. et al. Long-lived quantum memory enabling atom-photon entanglement over 101 km of telecom fiber. *PRX Quantum* **5**, 020307 (2024).
36. Jansen, S., Goodenough, K., de Bone, S., Gijswijt, D. & Elkouss, D. Enumerating all bilocal Clifford distillation protocols through symmetry reduction. *Quantum* **6**, 715 (2022).
37. Bennett, C. H., DiVincenzo, D. P., Smolin, J. A. & Wootters, W. K. Mixed-state entanglement and quantum error correction. *Phys. Rev. A* **54**, 3824 (1996).
38. Horodecki, M., Horodecki, P. & Horodecki, R. General teleportation channel, singlet fraction, and quasidistillation. *Phys. Rev. A* **60**, 1888 (1999).
39. Benjamin, S. C., Browne, D. E., Fitzsimons, J. & Morton, J. J. Brokered graph-state quantum computation. *New J. Phys.* **8**, 141 (2006).
40. Rozpedek, F. et al. Near-term quantum-repeater experiments with nitrogen-vacancy centers: Overcoming the limitations of direct transmission. *Phys. Rev. A* **99**, 052330 (2019).
41. Lee, Y., Bersin, E., Dahlberg, A., Wehner, S. & Englund, D. A quantum router architecture for high-fidelity entanglement flows in quantum networks. *npj Quantum Inf.* **8**, 75 (2022).
42. Campbell, E. T. & Benjamin, S. C. Measurement-based entanglement under conditions of extreme photon loss. *Phys. Rev. Lett.* **101**, 130502 (2008).
43. Jones, C., Kim, D., Rakher, M. T., Kwiat, P. G. & Ladd, T. D. Design and analysis of communication protocols for quantum repeater networks. *New J. Phys.* **18**, 083015 (2016).
44. Dehaene, J., Van den Nest, M., De Moor, B. & Verstraete, F. Local permutations of products of bell states and entanglement distillation. *Phys. Rev. A* **67**, 022310 (2003).
45. Dür, W. & Briegel, H. J. Entanglement purification and quantum error correction. *Rep. Prog. Phys.* **70**, 1381 (2007).
46. Ruan, L., Kirby, B. T., Brodsky, M. & Win, M. Z. Efficient entanglement distillation for quantum channels with polarization mode dispersion. *Phys. Rev. A* **103**, 032425 (2021).
47. Van Meter, R., Ladd, T. D., Munro, W. J. & Nemoto, K. System design for a long-line quantum repeater. *IEEE/ACM Trans. Netw.* **17**, 1002–1013 (2008).
48. Rozpedek, F. et al. Optimizing practical entanglement distillation. *Phys. Rev. A* **97**, 062333 (2018).
49. Krastanov, S., Albert, V. V. & Jiang, L. Optimized entanglement purification. *Quantum* **3**, 123 (2019).
50. Marler, R. T. & Arora, J. S. Survey of multi-objective optimization methods for engineering. *Struct. Multidiscip. Optim.* **26**, 369–395 (2004).
51. Zhou, L., Zhong, W. & Sheng, Y.-B. Purification of the residual entanglement. *Optics Express* **28**, 2291–2301 (2020).
52. Sheng, Y.-B. & Deng, F.-G. Deterministic entanglement purification and complete nonlocal Bell-state analysis with hyperentanglement. *Phys. Rev. A* **81**, 032307 (2010).

53. Briegel, H.-J., Dür, W., Cirac, J. I. & Zoller, P. Quantum repeaters: the role of imperfect local operations in quantum communication. *Phys. Rev. Lett.* **81**, 5932 (1998).
54. Werner, R. F. Quantum states with Einstein-Podolsky-Rosen correlations admitting a hidden-variable model. *Phys. Rev. A* **40**, 4277 (1989).

Acknowledgements

We thank S. Jansen, C. Cicconetti, P. Kaku, and J. van Dam for discussions and feedback. Á.G.I. acknowledges financial support from the Netherlands Organisation for Scientific Research (NWO/OCW), as part of the Frontiers of Nanoscience programme. B.D. acknowledges financial support from a KNAW Ammodo Award (S.W.). S.W. acknowledges support from an NWO VICI grant.

Author contributions

B.D. and Á.G.I. conceived and defined the project. B.D. and S.K. proved Theorems 1 and 2. Á.G.I. and B.D. proved Propositions 1 and 2. Á.G.I. carried out the analysis from “Monotonic performance” and “Discussion”, and coded the discrete-event simulation (used to validate analytical results). Á.G.I. and B.D. wrote this manuscript. S.W. supervised the project.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41534-025-01161-3.

Correspondence and requests for materials should be addressed to Bethany Davies.