

## Optimizing repeater schemes for the quantum internet

Kenneth Goodenough,<sup>1</sup> David Elkouss ,<sup>1</sup> and Stephanie Wehner<sup>1,2</sup>

<sup>1</sup>*QuTech, Delft University of Technology, Lorentzweg 1, 2628 CJ Delft, The Netherlands*

<sup>2</sup>*Kavli Institute of Nanoscience, Delft University of Technology, Lorentzweg 1, 2628 CJ Delft, The Netherlands*



(Received 23 June 2020; accepted 4 February 2021; published 18 March 2021)

The rate at which quantum communication tasks can be performed using direct transmission is fundamentally hindered by the channel loss. Quantum repeaters allow one, in principle, to overcome these limitations, but their introduction necessarily adds an additional layer of complexity to the distribution of entanglement. This additional complexity—along with the stochastic nature of processes such as entanglement generation, Bell swaps, and entanglement distillation—makes finding good quantum repeater schemes nontrivial. We develop an algorithm that can efficiently perform a heuristic optimization over a subset of quantum repeater schemes for general repeater platforms. We find a strong improvement in the generation rate in comparison to an optimization over a simpler class of repeater schemes based on BDCZ (Briegel, Dür, Cirac, Zoller) repeater schemes. We use the algorithm to study three different experimental quantum repeater implementations on their ability to distribute entanglement, which we dub *information processing* implementations, *multiplexed elementary pair generation* implementations, and combinations of the two. We perform this heuristic optimization of repeater schemes for each of these implementations for a wide range of parameters and different experimental settings. This allows us to make estimates on what are the most critical parameters to improve for entanglement generation, how many repeaters to use, and which implementations perform best in their ability to generate entanglement.

DOI: [10.1103/PhysRevA.103.032610](https://doi.org/10.1103/PhysRevA.103.032610)

### I. INTRODUCTION

The distribution of bipartite entanglement is critical for quantum communication tasks. Examples of such tasks include conference key agreement [1,2], clock synchronization [3–5], and secure multiparty quantum computation [6]. Photonic transfer of quantum states through optical fiber is one of the main candidates for long-distance entanglement generation. This is due to the potential of fast transmission speeds and the potential to be integrated with the hardware of classical networks. However, unlike classical bits, quantum states cannot be copied [7,8], which prevents us from amplifying the signal at intermediate points. In fact, the rate of entanglement generation over a fiber with transmissivity  $\eta \ll 1$  necessarily scales linearly in  $\eta = \exp(-\frac{L}{L_0})$  [9–12], where  $L$  is the total distance, and  $L_0$  is the attenuation length. Thus, for large enough distances, the losses are a limiting factor on the rate of entanglement generation.

Quantum repeaters aim to counteract the effects of loss [13–16]. Quantum repeater schemes are built on the concept of breaking the total length between two parties—Alice and Bob—up into several shorter (elementary) *links*. At the two end points of these elementary links there is a *repeater node*, which is a collection of quantum information processing (IP) devices. Depending on the scheme, the nodes have different requirements ranging from storage of quantum states to full-fledged quantum computation. By generating and storing entanglement over the elementary links and performing Bell state measurements on the locally held states, the distance over which entanglement is present can be

increased, until the two parties at the end are entangled [13–16].

However, the imperfect operations during this process lower the quality of the entanglement, potentially ruining the benefits of utilizing quantum repeater nodes. The effects of noise can be counteracted by using *entanglement distillation*, which can (in general probabilistically) turn multiple entangled pairs of lower fidelity into a smaller amount of pairs with higher fidelity [17–19].

An entanglement generation scheme between two spatially separated parties Alice and Bob consists of the generation of entanglement over elementary links, entanglement swaps, and distillation. Our goal is to find schemes that minimize the generation time of the entanglement between Alice and Bob for a given fidelity to the maximally entangled state in a suitable experimental model. However, finding optimal schemes is nontrivial for two reasons. First, the amount of schemes that can be performed grows superexponentially in the number of elementary links or nodes, making a full systematic optimization infeasible (see Ref. [20] and Appendix A). Second, entanglement generation, Bell state measurements, and distillation are all processes that are in general probabilistic. Finding the corresponding probability distributions is believed to be computationally intensive [21–24].

For the reasons mentioned above, it seems necessary to either approximate or simplify the problem. Notably, in [20], an algorithm based on dynamical programming was proposed capable of efficiently optimizing repeater schemes over the full parameter space. Under the heuristic approximation that all processes finish at the average time and there is no

decoherence over time in the quantum memories, the algorithm constructs the scheme for a large chain combining the optimal solutions over smaller (multihop) links.

We take a different route. Instead of approximating the behavior of the schemes by the mean, we simplify the problem by considering a relevant subset of schemes. In particular, we consider schemes that succeed at all levels near-deterministically. Such schemes have the benefit of having a small variance of the fidelity and generation time. We note that the requirement of being near-deterministic does not imply that our algorithm cannot handle nondeterministic processes. High success probabilities can be enforced even when certain processes are not deterministic—in that case, the probability of a single success can be increased by repeating the process a number of times, ensuring that the whole process can be made near-deterministic (see Sec. II for further details). Furthermore, this allows us to calculate the success probability of a scheme exactly, even when more complicated protocols such as distillation and probabilistic swapping are performed. Finally, this approach also allows us to calculate the average noise experienced during storage, in contrast to [20] (see Appendix C).

In this paper, we detail an algorithm (publicly available as a PYTHON script at [25]) that performs a heuristic optimization over the set of near-deterministic schemes when there are  $n$  elementary links in  $O[n^2 \log_2(n)]$  time, and  $O[n \log_2(n)]$  time if all the nodes have the same parameters and are equidistant. Concretely, the input to our algorithm is given by the experimental parameters of the nodes and connecting fibers, the distances between adjacent nodes, the possible protocols for elementary pair generation (EPG), swapping and distillation, and a set of algorithm-specific parameters (see Sec. II D 3). The algorithm returns a collection of optimized schemes for generating entanglement between Alice and Bob.

We exploit the fact that our algorithm is not specific to any particular experimental setup, which allows for the optimization over repeater schemes for several types of platforms.

The experimental platforms that we consider can be split up into three types.

(1) *Information processing platforms*—IP implementations have the ability to store quantum states and perform operations on them, such that it is possible to perform distillation. However, the number of quantum states that can be processed at the same time is presently limited to a small number. Examples of information processing implementations include systems such as trapped ions [26–28], nitrogen-vacancy (NV) centers in diamond [29,30], neutral atoms [31–33], and quantum dots [34,35].

(2) *Multiplexed elementary pair generation platforms*—Multiplexed elementary pair generation (MP) implementations lack the ability to properly perform operations on the stored states, prohibiting distillation. However, a large number ( $10^4$ – $10^7$ ) of states can potentially be generated, transmitted, and stored simultaneously with such implementations, effectively increasing the success probability for the elementary pair generation. Examples of such implementations include the different types of atomic ensembles [36–38].

(3) *A combination of IP and MP platforms*—Multiplexed elementary pair generation platforms can overcome the effects of losses over the elementary links more easily than informa-

tion processing platforms, but suffer from the lack of control and long coherence times available to information processing platforms. This motivates a combination of the two. That is, the elementary pair generation is performed with an MP implementation, after which the quantum state is transferred into an information processing system. Such a combined setup benefits from the high success probability of the generation of the elementary pairs, together with the ability to perform entanglement distillation and longer coherence times.

We find that the optimization returns schemes that outperform a simplified optimization over more structured schemes, similar to those in [13–16]. This highlights the complexity of repeater protocols for realistic repeater chains and the nontrivial nature of the optimization problem. With such optimized schemes in hand we use our algorithm to study a range of questions, such as which setups hold promise for near-term quantum networks, how many nodes should be implemented, and which experimental parameters are the most important to improve upon.

In Sec. II we detail the basics of our algorithm, which takes as input an arbitrary repeater chain configuration, and returns a collection of heuristically optimized schemes which generate entanglement between two specified nodes, i.e., the schemes have an optimal tradeoff between the fidelity and generation time (over the set of considered schemes). This section also contains the heuristics we use to reduce the search space or complexity of the algorithm in Sec. II D (with further details in Appendices A and B regarding the complexity and runtime) and closes with the pseudocode of our algorithm in Sec. II D 3. Section III contains an overview of how we model the three experimental platforms considered in this paper, namely, information processing (Sec. III A) implementations, multiplexed (Sec. III B) implementations, and a combination of the two (Sec. III C). We then use the algorithm to heuristically optimize over repeater schemes for each of the implementations for several different scenarios in Sec. IV. We close with a discussion of the results and the algorithm in Sec. V.

## II. ALGORITHM DESCRIPTION

In this section we first explain the general structure of quantum repeater schemes (Sec. II A). We then focus on the construction of so-called near-deterministic schemes (Sec. II B). Afterwards, we first detail a non-scalable brute-force algorithm for optimizing over such near-deterministic schemes (Sec. II C), after which we provide a feasible algorithm by implementing certain heuristics into the brute-force algorithm (Sec. II D). Appendices A and B contain a more explicit discussion regarding the complexity and runtime with and without the heuristics implemented.

### A. Structure of quantum repeater schemes

The goal of a quantum repeater scheme is to distribute an entangled state between two remote parties Alice and Bob. Quantum repeater schemes are built up from smaller schemes. Schemes are constructed by performing *connection* and *distillation* protocols on pairs of smaller schemes.

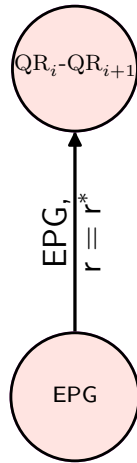


FIG. 1. Elementary pair generation (EPG) between adjacent nodes  $QR_i$  and  $QR_{i+1}$ . The schemes take a number of rounds  $r = r^*$ , even if entanglement is generated at an earlier round. See main text for further details.

Connection protocols extend the range over which entanglement exists. This can be done by elementary pair generation and entanglement swapping. EPG creates entanglement over elementary links (see Fig. 1). Entanglement swapping transforms two entangled states over two shorter (multihop) links to an entangled state over a longer multihop link using a Bell state measurement, (see Fig. 2).

Distillation protocols allow one to (possibly probabilistically) convert two entangled states to a single, more entangled state using only local operations and classical communication [18,39]. There exist more complicated protocols, where an arbitrary number of entangled states are converted to a smaller number of entangled states [40,41]. Here, we only consider distillation protocols taking two states to a single one [42]. See Fig. 3 for an illustration of a distillation protocol.

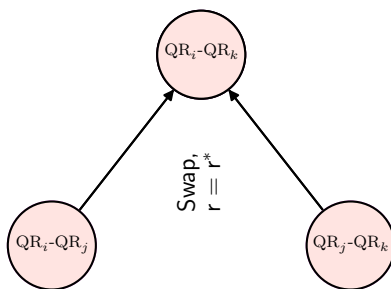


FIG. 2. Entanglement swapping between two entangled pairs between (multihop) links  $(QR_i, QR_j)$  and  $(QR_j, QR_k)$ , indicated by a circle node. By performing a Bell state measurement on the two local states at  $QR_j$ , the two entangled states turn into one entangled state between  $(QR_i, QR_k)$ . The schemes take a number of rounds  $r = r^*$  even if the scheme succeeds at an earlier round. See main text for further details. Note that the distances over which the entanglement has been generated for the (multihop) links  $(QR_i, QR_j)$  and  $(QR_j, QR_k)$  need not be the same.

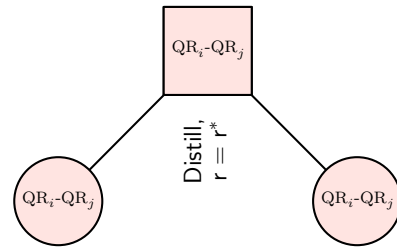


FIG. 3. Example of a generic entanglement distillation protocol, transforming (possibly probabilistically) two entangled states to a single, more entangled state between nodes  $QR_i$  and  $QR_j$ , using only local operations and classical communication. Distillation is indicated by a square node. The schemes take a number of rounds  $r = r^*$  even if distillation succeeds at an earlier round. See main text for further details. Note that  $QR_i$  and  $QR_j$  do not have to be directly connected by a fiber.

### B. Near-deterministic schemes

Entanglement generation schemes should preferably minimize the average generation time for a given fidelity  $F$ . However, the generation and distribution of entanglement is a *stochastic* process, greatly complicating the optimization over such schemes. Here, we simplify the problem by demanding that every step of the entanglement generation scheme is *near-deterministic*. This requirement can be enforced even when some of the processes are not deterministic, such as elementary pair generation or Bell swaps. The probability of having at least a single success can be increased by repeating the whole scheme up until that point for multiple attempts [43]. Near-deterministic schemes deliver a state with high probability at a specific time  $T$ , and it is this generation time  $T$  that we use as our metric in this paper [44].

Let us exemplify this idea through a process for EPG. This process might have a very small probability  $p$  to succeed in a single attempt, which takes a time  $T_{\text{attempt}}$  to perform. The probability of having at least a single success after  $r$  attempts is

$$p_{\text{single success}} = 1 - (1 - p)^r. \quad (1)$$

Thus, the probability of having at least one success can be increased to no less than  $p_{\text{min}}$  by trying for  $r = \lceil \frac{\log_2(1-p_{\text{min}})}{\log_2(1-p)} \rceil$  attempts. We now consider protocols where the state is stored until a total time  $rT_{\text{attempt}}$  has passed, even if a success occurs before  $r$  attempts have passed. This ensures that a state can be delivered near-deterministically (i.e., with probability at least  $p_{\text{min}}$ ) at a prespecified time  $T = rT_{\text{attempt}}$ . However, it comes at the cost of increased decoherence, since the state might have to be stored for a longer time (see [23] for a related concept).

Consider now the success probability of distillation protocols and (optical) Bell state measurements. Both protocols require the two states to be present, which holds with probability equal to the product of the probabilities of the two individual schemes having succeeded. Furthermore, distilling and swapping typically have a nonzero failure probability, potentially decreasing the success probability even further. However, we can use the same strategy used previously to increase the total success probability. That is, by repeating

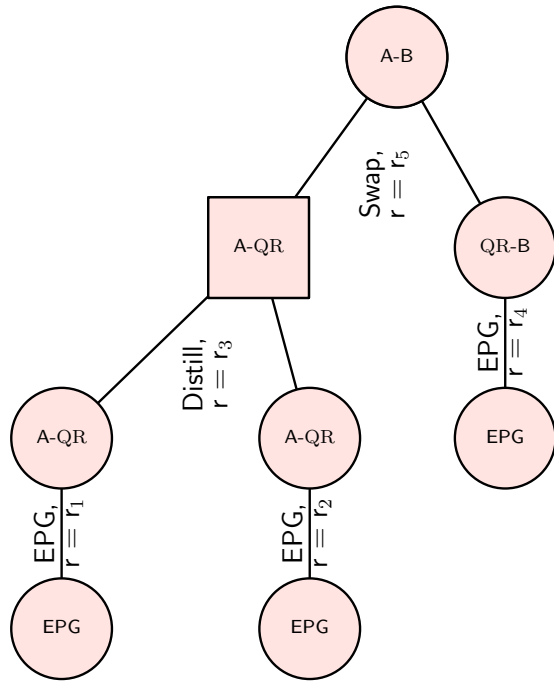


FIG. 4. Schematic description of how near-deterministic schemes are constructed from the protocols shown in Figs. 1–3. Here entanglement is generated between the nodes A and B, using an intermediate node labeled by QR. The overall structure is that of a binary tree (modulo the leaves indicating elementary pair generation, indicated by EPG), since swapping and distillation are always performed between exactly two schemes. Each subtree is required to succeed with probability at least  $p_{\min}$ , which can be enforced by repeating the whole subtree for a number of attempts  $r$ . Here, the specific number of attempts is indicated by  $r_b$ ,  $b \in \{1, 2, 3, 4, 5\}$ . The circular nodes indicate either elementary pair generation or swapping, and the square nodes indicate distillation.

the whole scheme up to that point, it is possible to increase the success probability to at least the threshold  $p_{\min}$ . Let us consider this concept for the example of a swap operation between two elementary pairs. The total success probability can now be increased by repeating the whole process of generating both elementary pairs and performing the swap operation.

This concept can be extended to more complex repeater schemes, ensuring that each step in the repeater scheme succeeds with high probability. A repeater scheme can thus be constructed by combining protocols from the ground up, where the average state, generation time  $T$ , and success probability  $p$  of each scheme are only a function of the number of attempted rounds  $r$ , the protocol used, the parameters of the repeater chain, and the used schemes. We show an example of how such schemes can be constructed in Fig. 4.

We note here that such near-deterministic schemes require us to keep states stored for some time, even if the underlying process has already succeeded, similar to the approaches in [23,45]. This evidently comes at the cost of increased storage times, and thus a greater amount of average decoherence. Near-deterministic schemes also have benefits, however. First, with near-deterministic schemes it is possible to make the variance of the resultant probability distributions arbitrarily small by increasing  $p_{\min}$ . Thus, near-deterministic protocols

are able to deliver entanglement at a prespecified time with high probability, which may be important for quantum information protocols consisting of multiple steps [45], such as entanglement routing [46,47]. Second, it is possible to calculate exactly the generation times and fidelities of near-deterministic schemes with relative ease, allowing for the optimization over such schemes.

Let us now compare near-deterministic schemes with two similar frameworks considered in [20,23]. Both near-deterministic schemes and the schemes considered in [20] take as building blocks a similar set of probabilistic protocols. In [20], however, the protocols are freely combined, which makes it challenging to estimate the average time they take to generate entanglement. This problem is sidestepped in [20] by heuristically assuming that all protocols take average time. In contrast, in our framework, we combine protocols in blocks that have high success probability and take a fixed amount of time. This reduces the class of schemes but allows us to estimate exactly the generation time and the fidelity of the state generated.

The protocols from [23] are constructed in a similar fashion as we do in this paper without the near-deterministic requirement. More concretely, the steps in the protocols are repeated a maximum number of times until success and the state is only read after the time equivalent to the maximum number of attempts has elapsed. The protocols in [23] are, however, more structured as they consist of combinations of elementary pair generation and swapping over repeater chains with a power of two elementary links. This structure enabled the analytical optimization of the number of attempts under dephasing noise.

### C. Brute-force algorithm

We now introduce a brute-force algorithm to optimize entanglement distribution over the set of near-deterministic schemes between two distant nodes Alice and Bob. The algorithm takes as input the experimental parameters of the nodes and connecting fibers, the distances between adjacent nodes, a set of protocols for elementary pair generation, swapping and distillation, a minimum success probability, and a limit on the maximum number of attempts and the maximum number of distillation rounds. The output consists of a data structure containing the schemes that minimize generation time parametrized by success probability and fidelity.

The brute-force algorithm generates and stores every possible scheme that can be created from the input conditions. Then for each achieved fidelity, it walks over the stored schemes to find the scheme minimizing the generation time achieving at least that fidelity. In the following we sketch only the first part, as this is enough to argue that such an approach is non-scalable.

First, the algorithm takes the set  $\mathcal{E}$  of protocols for elementary pair generation, together with the different number of attempts considered (of which there are at most  $r_{\text{discr}}$ ), and explores all possible combinations of elementary pair generation protocols and number of attempts for each elementary link. Each of these combinations is stored if the success probability is larger than a specified  $p_{\min}$ .

Next, the algorithm takes the set of distillation protocols  $\mathcal{D}$  and a maximum number of distillation rounds  $m$ . For

each elementary link, the algorithm loops over the number of distillation rounds:  $1, \dots, m$ . For each number of rounds, the algorithm explores all combinations of pairs of schemes, number of attempts, and distillation protocols and stores the resulting scheme if the success probability is larger than  $p_{\min}$ .

The algorithm then proceeds iteratively over multihop links of length  $i \in \{2, 3, \dots, n\}$ , where  $n$  is the total number of elementary links between the target nodes. Each iteration is divided into a swapping and distillation step.

In the swapping step the algorithm considers all adjacent (multihop) links of lengths  $i_1, i_2$  such that  $i_1 + i_2 = i$ . For each valid pair of adjacent links and for each pair of schemes stored over the adjacent links, the algorithm explores all combinations of number of attempts and protocols in the set of swapping protocols  $\mathcal{S}$ . It stores a resulting scheme if the success probability is larger than  $p_{\min}$ .

In the distillation step, the algorithm proceeds analogously to the description above for distillation over elementary links. The output of the brute-force algorithm is then a collection of schemes. Each of these schemes is built up from smaller schemes, similar to the scheme shown in Fig. 4.

While the approach just described might work for a very small chain, the number of schemes grows too quickly. In particular, the number of near-deterministic schemes to consider in the brute-force approach is lower bounded by

$$O((r_{\text{discr}})^2 |\mathcal{E}| |\mathcal{S}|^n) \quad (2)$$

when distillation protocols are not considered and by

$$O(r_{\text{discr}} |\mathcal{E}| |\mathcal{S}| |\mathcal{D}|^{2^{m-n}}) \quad (3)$$

when distillation is considered (see Appendix A). Here  $n$  is the number of elementary links,  $|\mathcal{E}|$  is the number of ways elementary pairs can be generated (due to for example varying a parameter over some set of values),  $|\mathcal{S}|$  is the number of swapping protocols,  $|\mathcal{D}|$  is the number of distillation protocols,  $r_{\text{discr}}$  is the different number of attempts considered, and  $m$  is the number of distillation rounds.

#### D. Heuristic algorithm

Now we introduce an efficient heuristic optimization algorithm. The heuristic algorithm takes as starting point the brute-force algorithm presented before and incorporates a number of modifications that reduce the search space, thus overcoming the fast-growing complexity of the brute-force algorithm. We divide the modifications into heuristics for the pruning of schemes and heuristics for good schemes and detail them in the following. In the following we first discuss the modifications to the brute-force algorithm before presenting the pseudocode of the algorithm and analyzing its complexity.

##### 1. Heuristics for the pruning of schemes

The brute-force algorithm explores a grid of parameters at each step and stores all schemes with success probability above  $p_{\min}$  independently of their quality. Instead, we can identify schemes that either are unlikely to combine into good

schemes at subsequent steps or are very similar to existing schemes and not store them.

A first strategy is to only store schemes that deliver a state with fidelity above the threshold  $F_{\text{threshold}} \geq \frac{1}{2}$ .

A second strategy is to *coarse-grain* the fidelity and success probabilities. For this, the algorithm rounds the fidelity  $F$  and success probability  $p$  of each scheme to  $\tilde{F}$  and  $\tilde{p}$ , the closest values in the sets  $[F_{\text{threshold}}, F_{\text{threshold}} + \varepsilon_F, F_{\text{threshold}} + 2\varepsilon_F, \dots, 1]$  and  $[p_{\min}, p_{\min} + \varepsilon_p, p_{\min} + 2\varepsilon_p, \dots, p_{\max}]$  (see Appendix B).

If no scheme with the same  $\tilde{F}$  and  $\tilde{p}$  exists, the scheme is stored. Otherwise, we compare the two generation times of the two schemes. If the old scheme has a lower generation time, the new scheme is not stored. Otherwise, the new scheme replaces the old one. We note here that the actual values of  $F$  and  $p$  are stored, and not the values  $\tilde{F}$  and  $\tilde{p}$ .

The third strategy consists in pruning suboptimal protocols after having considered all protocols over a given (multihop) link. A scheme is suboptimal if there exists another scheme over that (multihop) link with the same  $\tilde{p}$  and has a lower generation time but equal or higher fidelity. We detail the implementation of the above pruning heuristics in Algorithm 2.

##### 2. Heuristics for good schemes

Pruning reduces the amount of suboptimal schemes that are kept stored. This prevents those schemes from being combined with other schemes, reducing the algorithm runtime. However, it would be preferable if those schemes would not even be considered in the first place. For this reason, we use *heuristics* on what schemes to consider. The heuristics that we use are *banded distillation*, *banded swapping*, and the *bisection heuristic*, which we will detail in what follows.

Many distillation protocols acting on two states yield states of fidelity larger than the input states only when the input states have fidelities that are relatively close to each other [48]. This motivates restricting distillation to states that have fidelities  $F_1$  and  $F_2$  separated at most by some threshold  $\varepsilon_{\text{distill}}$ :

$$|F_1 - F_2| \leq \varepsilon_{\text{distill}}. \quad (4)$$

This heuristic, first considered in [49], is called *banded distillation*.

Inspired by banded distillation we introduce a similar heuristic for entanglement swapping that we dub *banded swapping*. A naive extension of banded distillation to swapping would be to require that the absolute difference of the fidelities of the two swapped states be small. However, by investigating the heuristically optimized schemes, our numerical exploration (see Appendix B) suggests that the number of nodes over which the entanglement is generated also plays a role. In particular, we find that it is sufficient to restrict swapping to states that satisfy

$$|i_1 - i_2| \leq 2 \log_2(i_1 + i_2 - 1) \quad (5)$$

and

$$\left| \frac{\log_2(F_1)}{i_1} - \frac{\log_2(F_2)}{i_2} \right| \leq \varepsilon_{\text{swap}} \quad (6)$$

where  $\varepsilon_{\text{swap}}$  controls the granularity of the heuristic,  $F_1$  and  $F_2$  are the fidelities of the two states, and  $i_1$  and  $i_2$  are the numbers of elementary links over which the entanglement was generated, e.g., the number of elementary links between  $\text{QR}_i\text{-QR}_j$  and  $\text{QR}_j\text{-QR}_k$  in Fig. 2, respectively. We note that the first condition was already present in [20]. We found that the above heuristics work well even in the case of asymmetric repeater chains.

The third heuristic—which we call the *bisection heuristic*—is inspired by the BDCZ scheme [13]. Similarly to the BDCZ scheme, it applies to *symmetric repeater chains*. That is, repeater chains where all nodes have the same parameters and are connected by identical elementary links. However, unlike the BDCZ scheme which is only applicable if the number of elementary links is equal to a power of 2, the bisection heuristic is applicable independent of the number of elementary links.

The heuristic works as follows. Factorization allows us to write the total number of elementary links as  $n = 2^j h$ , where  $j$  is the number of times  $n$  is divisible by 2, and  $h$  is the odd part of  $n$ . First, an optimization is performed over a link of length  $h$ . From then on, similar to the BDCZ scheme, swapping only occurs between entanglement that has been generated over a total number of elementary links equal to a multiple of  $h$ . This heuristic has the possibility of dramatically reducing the algorithm runtime for certain values of  $n$ .

### 3. Pseudocode of the heuristic algorithm

We now present the pseudocode of the heuristic algorithm. The general algorithm is described in Algorithm 3, while the subroutines for storing the schemes and for the pruning heuristic are given in Algorithm 1 and Algorithm 2.

**Algorithm 1.** STORESCHEME, subroutine for storage of the schemes. Here, ‘link’ refers to either an elementary or multi-hop link.

---

**Input:** scheme, store,  $p_{\min}$ ,  $F_{\text{threshold}}$ , link,  $\varepsilon_F$ ,  $\varepsilon_p$   
**Output:** store with scheme possibly added

---

```

 $F \leftarrow$  fidelity stored in scheme ;
 $p \leftarrow$  probability stored in scheme ;
 $n_{\varepsilon_p} \leftarrow \text{argmin}_{n \in \mathbb{N}} \text{ s.t. } p < p_{\min} + n \cdot \varepsilon_p$ ;
 $n_{\varepsilon_F} \leftarrow \text{argmin}_{n \in \mathbb{N}} \text{ s.t. } F < F_{\text{threshold}} + n \cdot \varepsilon_F$ ;
if  $F \geq F_{\text{threshold}}$  then
  if store[link][ $n_{\varepsilon_p}$ ][ $n_{\varepsilon_F}$ ] already exists then
     $T' \leftarrow$  generation time of store[link][ $n_{\varepsilon_p}$ ][ $n_{\varepsilon_F}$ ];
    if  $T < T'$  then
      | store[link][ $n_{\varepsilon_p}$ ][ $n_{\varepsilon_F}$ ]  $\leftarrow$  scheme
    end
  else
    | store[link][ $n_{\varepsilon_p}$ ][ $n_{\varepsilon_F}$ ]  $\leftarrow$  scheme
  end
end
return store

```

---

**Algorithm 2.** PRUNE, prunes the sub-optimal schemes stored for a given link. Here, ‘link’ refers to either an elementary or multi-hop link.

---

**Input:** store,  $p_{\min}$ , link,  $\varepsilon_p$   
**Output:** store with sub-optimal schemes over link pruned

---

```

for  $n \geq 0$  s.t.  $p_{\min} + n \cdot \varepsilon_p \leq 1$  do
  orderedSchemes  $\leftarrow$  store[link][ $n$ ], ordered by
  fidelity from high to low;
   $N \leftarrow$  size of orderedSchemes;
  maxTime  $\leftarrow$  generation time of orderedSchemes[0];
  for  $i \leftarrow 1, \dots, N$  do
    if maxTime  $\leq$  generation time of
    orderedSchemes[ $i$ ] then
      | Remove orderedSchemes[ $i$ ] from
      | store[link][ $n$ ]
    else
      | maxTime  $\leftarrow$  generation time of
      | orderedSchemes[ $i$ ]
    end
  end
end
return store

```

---

The algorithm takes as input an additional number of parameters on top of the parameters already discussed for the brute-force algorithm. These parameters regard the heuristics and were described in the previous section. These parameters are  $\varepsilon_F$  and  $\varepsilon_p$  (the discretization used for the pruning of schemes for the fidelity and success probability, respectively) and  $F_{\text{threshold}}$  and  $p_{\min}$  (the minimum values required to consider a scheme for the fidelity and success probability, respectively). A software implementation requires also a number of experimental parameters for characterizing the hardware and estimating the output of each scheme, however we leave the explicit description of the hardware parameters out of the pseudocode. For details of the actual implementation, please refer to the repository [25].

### 4. Complexity and runtime of the heuristic algorithm

As we show in Appendix A, the heuristics allow us to go from a number of considered schemes that grows superexponentially in the number of elementary links, to a number of schemes that is upper bounded by

$$O\left[2^{r_{\text{discr}}}\left(\frac{(1 - F_{\text{threshold}})(1 - p_{\min})}{\varepsilon_F \varepsilon_p}\right)^2 n^2 \log_2(n)\right], \quad (7)$$

implying that the number of considered schemes is now only on the order of  $n^2 \log_2(n)$ , as opposed to superexponential in  $n$ . Here  $r_{\text{discr}}$  is the maximum number of values allowed for the number of attempts  $r$ ,  $F_{\text{threshold}}$  is the minimum fidelity we allow a scheme to have,  $p_{\min}$  is the minimum accepted success probability,  $\varepsilon_F$  and  $\varepsilon_p$  are the discretizations used for the coarse-graining, and  $n$  is the number of elementary links. Furthermore, in the case of a symmetric repeater chain (i.e., every node has the same parameters and the nodes are equidistant), the optimization can be further simplified. As

**Algorithm 3.** Heuristic optimisation over near-deterministic schemes for a repeater chain of  $n$  elementary links. Here, ‘link’ refers to either an elementary or multi-hop link.

---

**Input:**  $n$  : number of elementary links  $n$  in repeater chain  
 $\varepsilon_F, \varepsilon_p$  : coarse-graining parameters for the fidelity and probability  
 $\varepsilon_{\text{distill}}, \varepsilon_{\text{swap}}$  : parameters for the heuristics for distillation and swapping  
 $m$  : maximum number of distillation rounds  
 $p_{\text{min}}, p_{\text{max}}$  : minimum and maximum scheme success probabilities  
 $r_{\text{discr}}$  : number of different attempt values  
 $F_{\text{threshold}}$  : minimum fidelity for schemes to be stored  
 $\mathcal{E}, \mathcal{S}, \mathcal{D}$ : sets of protocols for elementary pair generation, swapping and distillation  
 $\mathcal{L}_i, i \in [1, n]$ : set of links of length  $i$

**Output:** store: a data structure containing entanglement generation schemes with the minimum generation time parametrised by the link, coarse grained success probability and coarse grained fidelity.

---

Initialise store

```

for  $i \leftarrow 1$  to  $n$  do
  for link in  $\mathcal{L}_i$  do
    if  $i = 1$  then
      // Loop over elementary pair generation protocols
      for EPGProtocol in  $\mathcal{E}$  do
        for  $r$  such that (1) or (F1) (IP/MP platforms, resp.) is between  $p_{\text{min}}$  and  $p_{\text{max}}$  in  $r_{\text{discr}}$  steps do
          scheme  $\leftarrow$  EPGProtocol( $r$ , link,  $n$ )
          store  $\leftarrow$  STORESCHEME(scheme, store,  $p_{\text{min}}$ ,  $F_{\text{threshold}}$ , link,  $\varepsilon_F$ ,  $\varepsilon_p$ )
        end
      end
    else
      // Loop over all schemes satisfying the swapping heuristic and over all swapping protocols
      for every link1 and link2 such that entanglement can be created over link by swapping between those links do
        for every pair  $(s_1, s_2)$  of stored schemes in store[link1] and store[link2] satisfying (5) and (6) do
          for swapProtocol in  $\mathcal{S}$  do
            for  $r$  such that (1) or (F1) (IP/MP platforms, resp.) is between  $p_{\text{min}}$  and  $p_{\text{max}}$  in  $r_{\text{discr}}$  steps do
              scheme  $\leftarrow$  swapProtocol( $s_1, s_2, r$ , link,  $n$ )
              store  $\leftarrow$  STORESCHEME(scheme, store,  $p_{\text{min}}$ ,  $F_{\text{threshold}}$ , link,  $\varepsilon_F$ ,  $\varepsilon_p$ )
            end
          end
        end
      end
    end
  end
  for  $j \leftarrow 1$  to  $m$  do
    // Loop over all schemes satisfying the distillation heuristic and over all distillation protocols
    for every pair  $(s_1, s_2)$  of stored schemes in store[link] and satisfying (4) do
      for distillationProtocol in  $\mathcal{D}$  do
        for  $r$  such that (1) or (F1) (IP/MP platforms, resp.) is between  $p_{\text{min}}$  and  $p_{\text{max}}$  in  $r_{\text{discr}}$  steps do
          scheme  $\leftarrow$  distillationProtocol( $s_1, s_2, r$ , link,  $n$ )
          store  $\leftarrow$  STORESCHEME(scheme, store,  $p_{\text{min}}$ ,  $F_{\text{threshold}}$ , link,  $\varepsilon_F$ ,  $\varepsilon_p$ )
        end
      end
    end
  end
  store  $\leftarrow$  PRUNE(store,  $p_{\text{min}}$ ,  $F_{\text{threshold}}$ , link,  $\varepsilon_p$ )
end
end
return store

```

---

we show in Appendix A, the number of schemes to consider in the symmetric case is upper bounded by

$$O\left[r_{\text{discr}}\left(\frac{(1 - F_{\text{threshold}})(1 - p_{\text{min}})}{\varepsilon_F \varepsilon_p}\right)^2 n \log_2(n)\right]. \quad (8)$$

In practice, we find that our algorithm runtime ranges from approximately 100 s to approximately 100 min, when considering 1 and 35 intermediate nodes for a symmetric repeater chain, respectively. We investigate the effects of the heuristics on the algorithm runtime in more detail in Appendix B, where we perform an experimental analysis of the algorithm runtime and its “accuracy” when varying  $\varepsilon_F$ ,  $\varepsilon_p$ ,  $\varepsilon_{\text{swap}}$ , and  $\varepsilon_{\text{distill}}$ . We use these results to settle on the values for  $\varepsilon_F$ ,  $\varepsilon_p$ ,  $\varepsilon_{\text{swap}}$ , and  $\varepsilon_{\text{distill}}$ . We only investigate the bisection heuristic when going to a larger number of nodes in Sec. IV C.

### III. PLATFORM MODELS

The algorithm discussed is independent of the underlying physical implementation, and can thus be applied to several experimental platforms. We use our algorithm to study three different types of platforms encapsulating a large range of technologies. The three platforms share the capability to store quantum information but differ in their quantum information processing capabilities. We call these platforms: information processing platforms, multiplexed elementary pair generation platforms, and combined platforms. Information processing platforms have the ability to perform operations on the stored qubits, but are currently limited to a small number of qubits. Multiplexed elementary pair generation platforms, on the other hand, lack the ability to perform operations on stored states, but can generate and store a potentially very large number of different states simultaneously. Obviously, these platforms differ greatly, but both approaches have complementary qualities for long-distance entanglement generation. This motivates us to also compare a combination of the two, that is, a setup where the elementary pairs are generated with a multiplexed elementary pair generation platform, but swapping and distillation are performed by an information processing platform.

In the rest of the section, we discuss the basics of each of the implementations and the modeling of the underlying processes.

#### A. Quantum repeaters based on information processing platforms

We call IP platforms those that have the capability to perform gates on the stored states, thus enabling entanglement distillation. The number of quantum states that can be stored and processed is presently limited. Experimental information processing platforms that have demonstrated excellent control over storage qubits include NV centers in diamond [19,30,45,50–53], neutral atoms [31,32], non-NV color centers in diamond [54,55], quantum dots [56–58], and trapped ions [26–28].

In this paper we consider two protocols for the generation of elementary pairs for information processing platforms. These protocols are the single-click [59–61] and double-click protocol [62]. We give an example based on nitrogen-vacancy

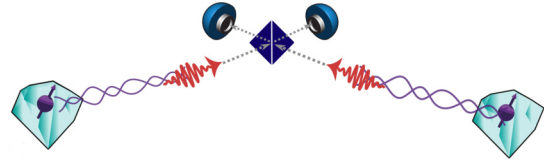


FIG. 5. An example of an elementary link implemented with an information processing platform. The two nodes are connected by a fiber with a beamsplitter in the middle and two detectors. For the case considered in this figure, the two nodes are nitrogen-vacancy centers in diamond. For both protocols, the two nodes both send one-half of an entangled state to the middle, which after interference and successful detection leads to a shared state between the two nodes. Figure taken with permission from [61].

centers in diamond in Fig. 5. We stress that this is just one example of an information processing platform, and that our algorithm can be applied to other platforms.

The setup for both the single-click and the double-click protocols consists of two nodes with at least one memory qubit. The two nodes are connected via an optical channel to an intermediate beamsplitter station with a detector at each of the output ports (see Fig. 5).

Let us now detail first the single-click protocol. The qubits at the nodes are prepared in a superposition of the ground state ( $|\downarrow\rangle$ ) and the first excited state ( $|\uparrow\rangle$ ):  $\sin(\theta)|\downarrow\rangle + \cos(\theta)|\uparrow\rangle$ . Upon receiving an appropriate excitation signal, the memory emits a photon ( $|1\rangle$ ) if it is in the excited state, and no photon ( $|0\rangle$ ) otherwise. Since the memory qubit is in a superposition, this results in a memory-photon entangled state  $\sin(\theta)|\downarrow\rangle|0\rangle + \cos(\theta)|\uparrow\rangle|1\rangle$ . The two photons are then directed to and interfered on the intermediate beamsplitter. One experimental complication here is that the phase picked up by the photons as they travel through the fiber is unknown unless the fibers are stabilized. However, if this is the case, upon the detection of a single photon (single click) at the beamsplitter station, the creation of an entangled pair can be heralded to the two nodes.

The double-click protocol on the other hand does not rely on phase stabilization. For the double-click protocol, each node prepares a qubit in a uniform superposition of the ground and first excited state [62]. By applying specific pulses to the qubits, a photon will be coherently emitted in the early or late time bin, depending on the state of the qubit at the node. The photons are then interfered at the beamsplitter station. The entanglement between the two qubits is heralded to the two nodes upon the detection of two consecutive clicks at the beamsplitter station. While the double-click protocol does not require phase stabilization, it has a lower success rate in comparison to the single-click protocol.

The parameter  $\theta$  is tuneable, which allows for a tradeoff between the success probability and the fidelity of the heralded state for the single-click protocol [19,61,63]. For the double-click protocol there is no such tradeoff, however.

For the single-click protocol we use the error model from [61]. For the double-click protocol we use the error model from [62].

Entanglement distillation across two separated matter qubits has been achieved with an NV-center setup [19], where



a specific entanglement distillation protocol [63] was implemented. This distillation protocol is optimal when the involved states are correlated in a particular manner [64]. In general, however, the states that we consider are not of this form. For this reason, we will consider here only the DEJMPS (Deutsch, Ekert, Jozsa, Macchiavello, Popescu, Sanpera) protocol [39], which was originally designed to work well for maximally entangled states with depolarizing noise. In this protocol, we first apply a local rotation on each of the qubits, then two local controlled-NOT (CNOT) operations, and measure the targets of the CNOT operations in the computational basis. We deem the distillation to be a success when the measurement outcomes are equal.

We now sketch the underlying abstract error models and the various experimental parameters.

State preparation for the generation of elementary pairs takes some time  $t_{\text{prep}}$ , performing the gates for distillation takes time  $t_{\text{distill}}$ , and performing a Bell state measurement takes time  $t_{\text{swap}}$ . State preparation is also imperfect, which we model as dephasing with parameter  $F_{\text{prep}}$ . States stored in the memories for a time  $t$  are subject to decoherence. We model this decoherence as joint depolarizing and dephasing noise (see Appendix C for details on the decoherence model).

The fiber has a refractive index of  $n_{\text{RI}}$  and an attenuation length  $L_0$ . The attenuation length is defined such that  $\eta = e^{-L/L_0}$ , where  $\eta$  is the transmissivity and  $L$  is the length of the fiber. There are three other sources of photon loss that we model [61,65]—the probability of successfully emitting a photon  $p_{\text{em}}$ , the probability of emitting a photon with the correct frequency and it not being filtered out (conditioned on having emitted the photon)  $p_{\text{pps}}$ , and the probability of the detector successfully clicking when a photon is incident  $p_{\text{det}}$ .

Applying gates induces noise on the states. Performing a Bell state measurement induces depolarizing and dephasing with parameters  $\lambda_{\text{BSM, depol}}$  and  $\lambda_{\text{BSM, deph}}$ , respectively. Performing the CNOT operations for distillation also leads to depolarizing and dephasing with parameters  $\lambda_{\text{CNOT, depol}}$  and  $\lambda_{\text{CNOT, deph}}$ , respectively. Furthermore, we model measurement errors by applying depolarizing noise with parameter  $\lambda_{\text{meas, depol}}$  before measuring a state. Finally, the uncertainty in the phase stabilization  $\Delta\phi$  induces dephasing in the state preparation for the single-click protocol (see [61]).

### B. Quantum repeaters based on multiplexed elementary pair generation platforms

MP platforms based on atomic ensembles are a promising candidate for quantum repeater implementations [36–38,66]. Such implementations generate elementary pairs with a potentially large number of modes at the same time. While multiplexed elementary pair generation platforms lack the ability to perform gates on the states stored in the memories, they have the potential to process a large number of states simultaneously, which can dramatically increase the probability at which elementary pairs can be generated. Here we discuss the basics of a model for the quantum repeater scheme proposed in [38] (see Appendix D). This repeater scheme uses photon-number and spectrally resolving detectors, frequency-multiplexed multimode memories, and parametric down conversion (PDC) sources.

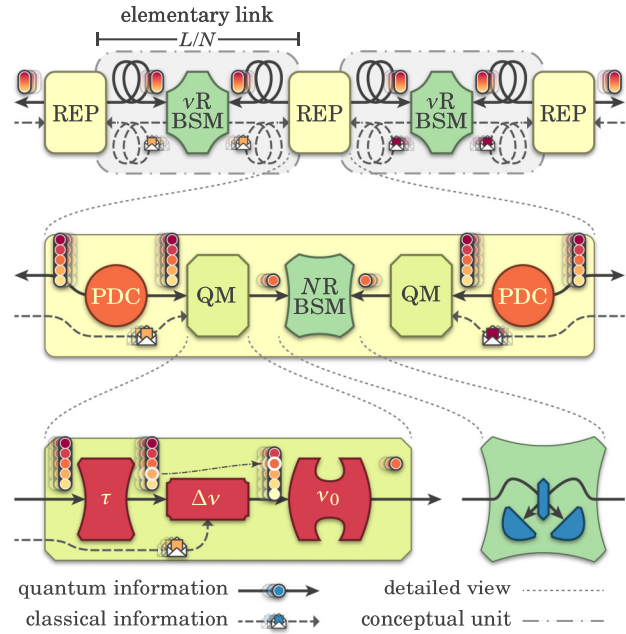


FIG. 6. Schematic description of an MP implementation. Top: The total distance  $L$  is split into  $N$  elementary links, each with a spectrally resolving BSM (indicated by  $\nu\text{RBSM}$ ) in the middle, and with two nodes (each indicated by  $\text{REP}$ ) at the end point of the elementary links. Middle: Zoom in of a node. Each node contains two PDC sources of multiplexed bipartite entanglement, two quantum memories (indicated by  $\text{QM}$ ) and a number-resolving Bell state measurement station (indicated by  $\text{NRBSM}$ ). Bottom: Detailed view of  $\text{QM}$  and  $\text{NRBSM}$ . Each quantum memory not only stores (in the unit indicated by  $\tau$ ), but can also perform a frequency shift (in a unit indicated by  $\Delta\nu$ ) and a frequency filter (indicated by the unit  $\nu_0$ ), while each  $\text{NRBSM}$  contains a beamsplitter and two single-photon detectors, which performs a Bell state measurement on the frequency-shifted photons. Illustration taken with permission from [38].

An elementary link consists of two PDC sources, each located at one of the two nodes. The PDC sources emit entangled states for a large set of frequencies. One half of each entangled state is sent towards a jointly collocated quantum memory, which can store a large number of modes simultaneously. The other half is sent to an intermediate station between the two nodes, where it interferes on a spectrally resolving beamsplitter with the corresponding state sent from an adjacent node. If at least one successful click pattern is detected at the output of the beamsplitter, the information of the corresponding mode is sent to the nodes. The information is used to filter out the other modes, after which frequency conversion is performed to a predetermined frequency at each of the nodes. The frequency conversion to a predetermined frequency ensures that at each node the successful modes from the two adjacent links can interfere at a local beamsplitter station. Photon-number resolving detectors are collocated at the output of the local beamsplitter to identify and discard multiphoton events. A schematic description can be found in Fig. 6.

Let us now investigate the parameters underlying the scheme we have just described. Consider a PDC source emitting entangled states with time-bin encoding. An ideal source would emit states of the form  $\frac{1}{\sqrt{2}}(|10, 01\rangle + |01, 10\rangle)$ , where the notation  $|nm, mn\rangle$  indicates  $n/m$  photons in the “early or late” bin in one-half of the state and  $m/n$  photons in the “early or late” bin in the other half. However, realistic PDC sources include additional terms. The resulting state can be approximated [38] by a state of the form

$$|\psi_{N_s}\rangle = \sqrt{p_0}|00, 00\rangle + \sqrt{\frac{p_1}{2}}(|10, 01\rangle + |01, 10\rangle) + \sqrt{\frac{p_2}{3}}(|20, 02\rangle - |11, 11\rangle + |02, 20\rangle), \quad (9)$$

with

$$p_0 = \frac{1}{(N_s + 1)^2}, \quad p_1 = \frac{2N_s}{(N_s + 1)^3}, \quad p_2 = 1 - p_0 - p_1. \quad (10)$$

Here  $N_s$  is the mean photon number present in the state and is a tuneable parameter. Increasing the mean photon number  $N_s$  increases the probability of detecting two clicks at the middle station (as can be seen from the decrease in the parameter  $p_0$ ), while at the same time lowering the fidelity of the state conditioned on detecting two clicks.

Note that (10) is a truncated version of the state derived in [67], i.e., all the higher-order terms are included in  $p_2$ . As described in [38], the multiphoton components limit the ability to generate entanglement without the use of photon-number resolving detectors.

The number of modes  $N_{\text{modes}}$  increases the success probability of elementary pair generation. If the success probability of the creation of a single elementary pair is given by  $p_{\text{el}}$ , the success probability of generating at least one elementary pair is given by  $1 - (1 - p_{\text{el}})^{N_{\text{modes}}}$ . Thus,  $N_{\text{modes}}$  should be on the order of  $\frac{1}{p_{\text{el}}}$ , since  $\lim_{p_{\text{el}} \rightarrow 0} 1 - (1 - p_{\text{el}})^{\frac{1}{p_{\text{el}}}} = 1 - e^{-1}$ . Finally, while a purely deterministic Bell state measurement is impossible using only linear optics [68,69], there are theoretical workarounds to increase the success probability [70–76]. We consider the approach introduced in [72], where the success probability of the Bell state measurement can be increased to  $1 - \frac{1}{2^{N+1}}$  by using  $2^{N+1} - 2$  ancillary photons.

We assume the states can be retrieved from the memories on demand. On-demand retrieval is necessary for our algorithm to work, since the storage times are not fixed. This is due to the uncertainty in which attempt entanglement will be generated. On-demand retrieval can be achieved with rare-earth ion ensembles by, for example, switching coherence from electronic levels to spin levels, as done in [77,78]. Besides allowing for on-demand recall, this also has the added benefit of increased memory lifetime [79].

We consider the same type of noise for operations as we did for information processing platforms. This means that measurements have an associated amount of depolarizing and dephasing. Finally, “decoherence” over time for the memory manifests as an exponential decay in the output efficiency of the memory, not in a reduction of the fidelity of the state [79,80]. Thus, the longer a state is stored, the smaller the probability it can be retrieved for measuring or further processing.

### C. Combining the two setups

An information processing implementation has the benefit of long coherence times and control over the memory qubits, which allows for distillation. On the other hand, multiplexed elementary pair generation platforms do not support distillation, but have the benefit of emitting and storing a large number of modes, increasing the success probability of the elementary pair generation significantly. Optimistically, one could imagine a futuristic setup which combines the strengths of the two setups. That is, elementary pair generation is performed by a multiplexed elementary pair generation platform, after which the successfully generated pairs are frequency converted into a frequency that can be stored in an information processing platform. The state is then stored in a memory, which can be done using, for example, a reflection-based heralded transfer [81,82]. For simplicity, we assume that the transfer and frequency conversion do not introduce any further noise or losses.

## IV. RESULTS

In this section, we study information processing platforms, multiplexed elementary pair generation platforms, and the combination thereof with the algorithm that we introduced in Sec. III. In order to compare different optimization results, we have chosen four sets of parameters for both platforms. With these sets, we first investigate the performance of information processing platforms for short ( $\approx 15$ – $50$  km), intermediate ( $50$ – $200$  km), and large (i.e.,  $\approx 200$ – $800$  km) distances. We then perform a similar investigation for multiplexed elementary pair generation platforms, after which we investigate the combination of the two. In order to get an understanding of the necessary parameters to generate remote entanglement with each platform or combination, the four sets of parameters for each platform are strictly ordered, with set 4 having the best parameters. We begin each three of the investigations with a specification of the input to our algorithm, which consists of the used elementary pair generation, swapping and distillation protocols, experimental parameters, and the parameters specific to the algorithm discussed previously.

In order to investigate longer repeater chains, we consider only *symmetric repeater chains* (see Sec. IID) in this section unless specified otherwise.

### A. Scheme optimization results for IP platforms

In the following we discuss the heuristic optimization results for information processing platforms. Let us first briefly discuss the protocols that we include in the optimization.

We consider two protocols for elementary pair generation: the single- and double-click protocol (see Sec. III A). The single-click protocol has an additional parameter  $\theta$ , which modulates the weight of the zero- and one-photon component [61]. We optimize over all single-click protocols with  $\theta$  taking values between  $\frac{1}{2}$  and  $\pi$ , equally spaced in 300 steps, thus  $|\mathcal{E}| = 301$ .

Both for swapping and distillation we consider a single protocol,  $|\mathcal{S}| = |\mathcal{D}| = 1$ . For swapping we perform a deterministic Bell state measurement on matter qubits while for distillation we implement the DEJMPS protocol. For swap-

TABLE I. Base parameters used for information processing platforms.

$t_{\text{prep}}$ (entanglement preparation time)	6 $\mu\text{s}$ [50]
$F_{\text{prep}}$ (dephasing for state preparation)	0.99 [50]
DcS (dark count rate)	10 Hz [50]
$L_0$ (attenuation length)	22 km [83]
$n_{\text{RI}}$ (refractive index of the fiber)	1.44 [83]
$\Delta\phi$ (optical phase uncertainty)	14.3° [45]
$F_{\text{gates, deph}}$ (dephasing for all gates)	1

ping and distillation, we optimize over all pairs of schemes that satisfy the banded swapping and distillation heuristics (see Sec. II D).

For all of the schemes,  $r$  ranges from  $r_{\text{min}}$  to  $r_{\text{max}}$  in (at most)  $r_{\text{discr}} = 200$  steps, where  $r_{\text{min}}$  and  $r_{\text{max}}$  are chosen such that the success probabilities are at least  $p_{\text{min}}$  and  $p_{\text{max}}$ , respectively.

We set  $\varepsilon_{\text{swap}} = \varepsilon_{\text{distill}} = 0.05$ ,  $\varepsilon_F = 0.01$ , and  $\varepsilon_p = 0.02$ . These parameters were settled on by investigating the tradeoff between the accuracy of the algorithm and its runtime (see Appendix B for a detailed analysis). We only consider  $m = 2$  distillation rounds. Finally, we set  $p_{\text{min}} = 0.9$ .

We now specify four sets of parameters for information processing platforms. We fix the parameters in Table I as a baseline common to all sets. We then choose sets of parameters for the efficiency coherence times, efficiencies, and gate fidelities, which can be found in Table II.

### 1. Entanglement generation for short distances with IP platforms

Small-scale experiments relevant for entanglement distribution with information processing platforms have already been performed [19,45,50,51,53,84,85], demonstrating the potential of such platforms for quantum networks. It is therefore of interest to understand what is within reach for information processing platforms, and what are the relevant parameters to improve. Thus, in this section we investigate how well we can perform entanglement generation with a small number of nodes and near-term parameters over short distances with information processing platforms. In particular, we are interested in when the introduction of a node becomes useful. To this end, we first consider entanglement generation over a distance of 50 km with parameter set 1. We show the results from our heuristic optimization in Fig. 7, where we consider the scenarios with no node, a single node, and two intermediate nodes. Furthermore, we plot the results where

TABLE II. Four different sets of example parameters considered for information processing platforms.

	Set 1	Set 2	Set 3	Set 4
$T_{\text{deph}}$ (dephasing with time)	3 s	10 s	50 s	100 s
$T_{\text{depol}}$ (depolarizing with time)	3 s	10 s	50 s	100 s
$p_{\text{em}}$ (probability of emission)	0.8	0.9	0.95	0.99
$p_{\text{ps}}$ (probability of postselection)	0.8	0.9	0.95	0.99
$F_{\text{gates}}$ (depolarization of all gates)	0.98	0.99	0.995	0.999

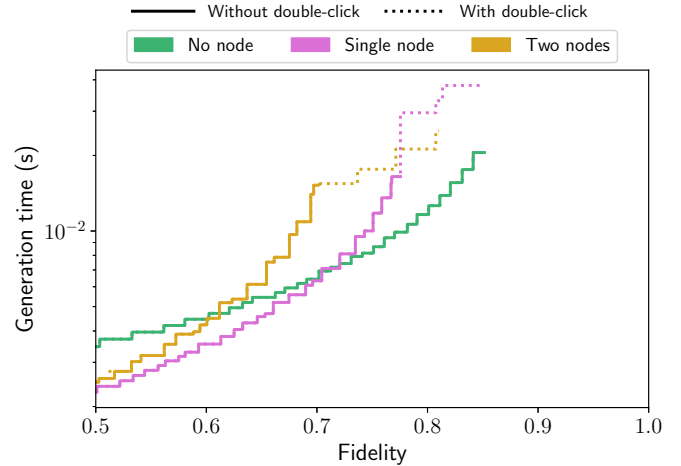


FIG. 7. Results of the achieved fidelity and generation time for a total distance of 50 km using parameter set 1 (see Table II) for information processing nodes, where we consider having 0 (green), 1 (purple), or 2 (yellow) of such intermediate nodes. The solid line corresponds to a heuristic optimization where we have excluded the double-click protocol, and the dotted line corresponds to a heuristic optimization with both the single- and double-click protocol. The double-click protocol does not provide a benefit for direct transmission, since the double-click protocol suffers more strongly from losses than the single-click protocol.

we include only the single-click protocol, and both the single- and double-click protocol.

First, the double-click protocol provides only a benefit for higher fidelities and for the scenarios with one and two intermediate nodes. This can be attributed to the fact that the double-click protocol is inherently less noisy if there are no losses, but is more sensitive to losses than the single-click protocol. However, this does not necessarily imply that all the elementary pairs have been generated with the double-click protocol. As we will see in later results, we will find schemes where elementary pairs are generated using both the single- and double-click protocol, indicating the importance of considering such complex schemes in our optimization.

Second, we observe that there is a crossover point for  $F \approx 0.7$  below which adding a node allows for a shorter generation time. Thus, implementing a quantum node over a modest distance of less than 50 km can in fact increase the generation rate by a moderate amount for low fidelities ( $\lesssim 0.7$ ). However, increasing the total distance does not shift this crossover point, since the maximum achieved fidelity with a single node also drops down if the parameters do not change.

Next, we explore the impact of a single parameter in the performance of implementations expected in the longer term. To this end, in Fig. 8 we investigate how the minimum generation time for several fixed target fidelities ( $F = 0.7, 0.8, 0.9$ ) scales, when varying the gate fidelities and coherence times and using parameter set 2. More specifically, we vary the gate fidelities from 0.98 to 1 and the coherence times  $T_{\text{deph}}$  and  $T_{\text{depol}}$  from 1 to 100 s. We perform a similar investigation in Fig. 10, where instead of varying the coherence times we vary the success probabilities of the detector successfully clicking ( $p_{\text{det}}$ ), successfully emitting a photon from a node ( $p_{\text{em}}$ ), and

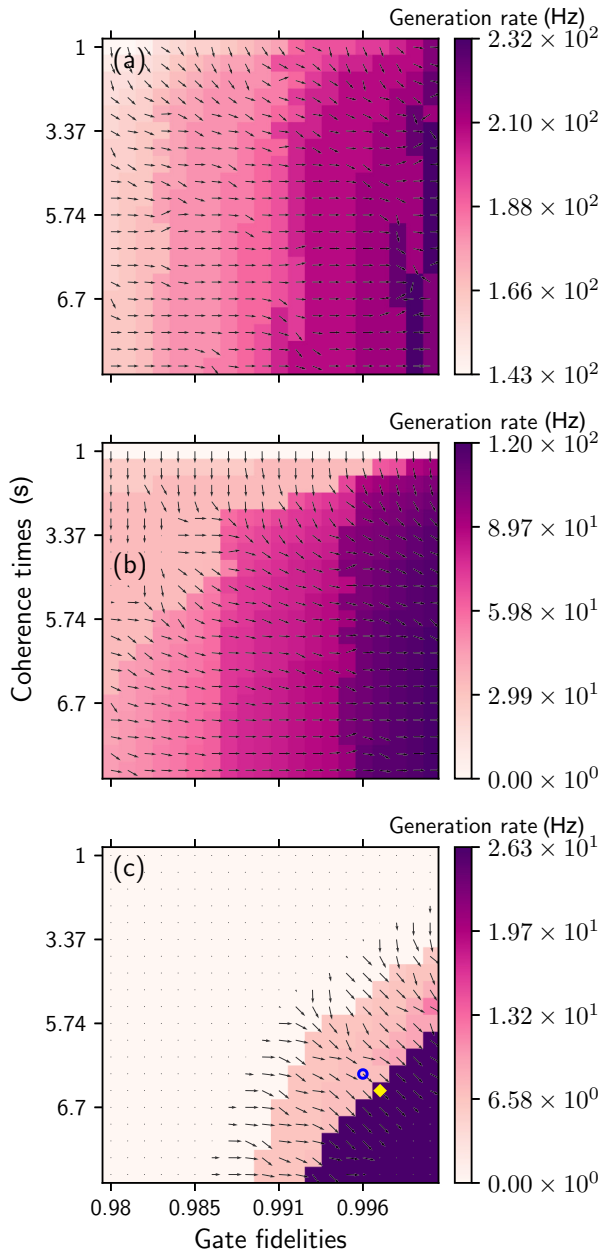


FIG. 8. Maximum generation rates for several different values of the coherence times (1–10 s) and gate fidelities (0.98 to 1) and for several different target fidelities, for a distance of 50 km and a single information processing node. Down and to the right in the plot indicate better parameters. All the other parameters are fixed to those of set 2 (Table II) or the base parameters (Table I). The target fidelities are (a)  $F = 0.7$ , (b)  $F = 0.8$ , and (c)  $F = 0.9$ , respectively. We also plot the gradient, indicating the direction and magnitude of steepest ascent. The blue ring and yellow diamond indicate the schemes we investigate in Fig. 9.

successfully emitting a photon of the correct frequency ( $p_{pps}$ ) simultaneously from 0.8 to 1.

From Fig. 8 we observe that increasing the gate fidelities has a bigger impact on the ability to generate entanglement than increasing the coherence times. In the bottom plot of Fig. 8 we choose two points, indicated by a blue ring and

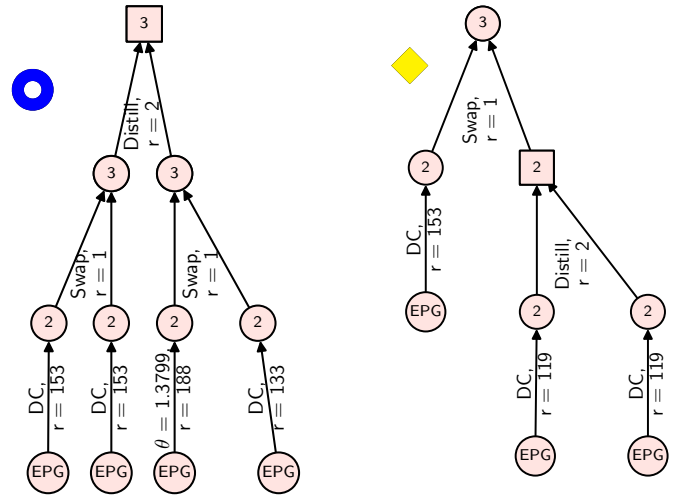


FIG. 9. Visualization of the two schemes indicated in the bottom of Fig. 8 by the blue ring (left) and the yellow diamond (right). The numbers indicate the number of nodes over which entanglement has been established, or elementary pair generation (EPG) has been performed. The “DC” indicates the double-click protocol, and the “ $\theta = \theta^*$ ” indicates a single-click protocol with the  $\theta$  parameter set to  $\theta^*$ . The “ $r$ ” here indicates the number of rounds the corresponding subtree is attempted. Note the necessity of combining disparate schemes—in both cases the EPG protocols used are not the same, and the yellow diamond scheme requires a swap on a distilled and undistilled pair.

a yellow diamond. The schemes corresponding to those two points are visualized in Fig. 9. The nonmonotonicity of the maximum generation rate most noticeable in Fig. 8(a) is an artifact from the heuristics occasionally leading to worse protocols, even with improved experimental parameters.

We make two observations about the algorithm from Fig. 9. First, the two schemes in Fig. 9 require swaps and distillation on states that have been created in different ways. This shows that already for only a single node entanglement distribution benefits from combining schemes in asymmetric fashion, even if the repeater chain itself is symmetric. Second, the algorithm is sensitive to parameter changes. We see that a small change in the parameters allows the diamond scheme to achieve a generation rate approximately four times as large as the ring scheme. This demonstrates further that the large space of explored schemes can provide a benefit.

The tradeoff between the success probability and the gate fidelities in Fig. 10 appears more complex. Not surprisingly, we observe that increasing the success probabilities has the greatest effect on the generation time and the ability to generate entangled states. In contrast to the previous scenario where only varying the gate fidelities leads to jumps in the generation time, we do not observe a similar phenomenon when varying the success probabilities. This is due to the fact that changing the success probabilities changes the generation time primarily by reducing the required number of attempts. Thus, if the minimal number of attempts  $r_{\min}$  is well approximated by a continuous function  $\hat{r}_{\min}(p)$  in  $p$ , we expect to see no jumps in the generation time as we vary  $p$ . More formally, we say that  $r_{\min}$  approximates  $\hat{r}_{\min}$  well if  $\frac{r_{\min}(p) - \hat{r}_{\min}(p)}{r_{\min}(p)} \approx 0$ .

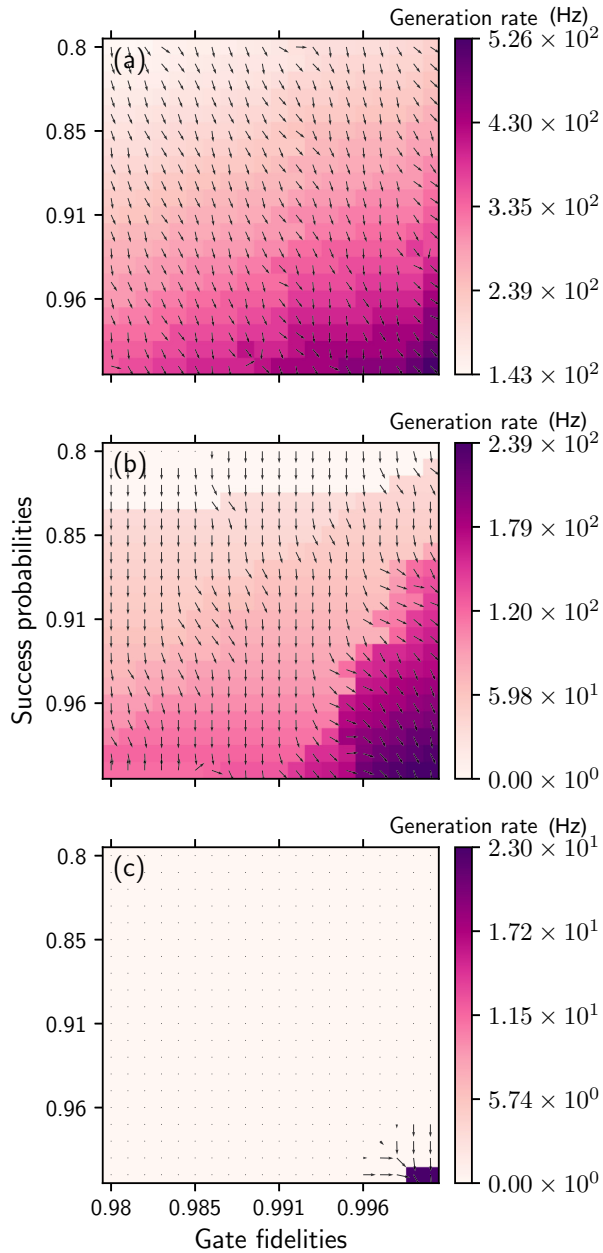


FIG. 10. Maximum achieved generation rates for several different values of the success probabilities (i.e., we vary  $p_{\text{det}} = p_{\text{em}} = p_{\text{ps}}$  simultaneously from 0.8 to 1) and gate fidelities (0.98 to 1), and for several different target fidelities, for a distance of 50 km and a single intermediate node for information processing platforms. Down and to the right in the plot indicate better parameters. All the other parameters are fixed to those of set 2 (Table I) and the base parameters (Table IV). The target fidelities are (a)  $F = 0.7$ , (b)  $F = 0.8$ , and (c)  $F = 0.9$ , respectively. We also plot the gradient, indicating the direction and magnitude of steepest ascent.

Since  $r_{\min}(p) = \lceil \frac{\log_2(1-p_{\min})}{\log_2(1-p)} \rceil$ , an obvious choice for  $\hat{r}_{\min}$  is  $\frac{\log_2(1-p_{\min})}{\log_2(1-p)}$ . Note that we then have that  $|r_{\min}(p) - \hat{r}_{\min}(p)| \leq 1$ , and that for  $p$  small enough  $\lceil \frac{\log_2(1-p_{\min})}{\log_2(1-p)} \rceil \gg 1$ . Since the total success probability of establishing an elementary pair is small, we have indeed that  $\frac{r_{\min}(p) - \hat{r}_{\min}(p)}{r_{\min}(p)} \approx 0$ , explaining the lack of sudden jumps. Furthermore, we find from Fig. 10(c)

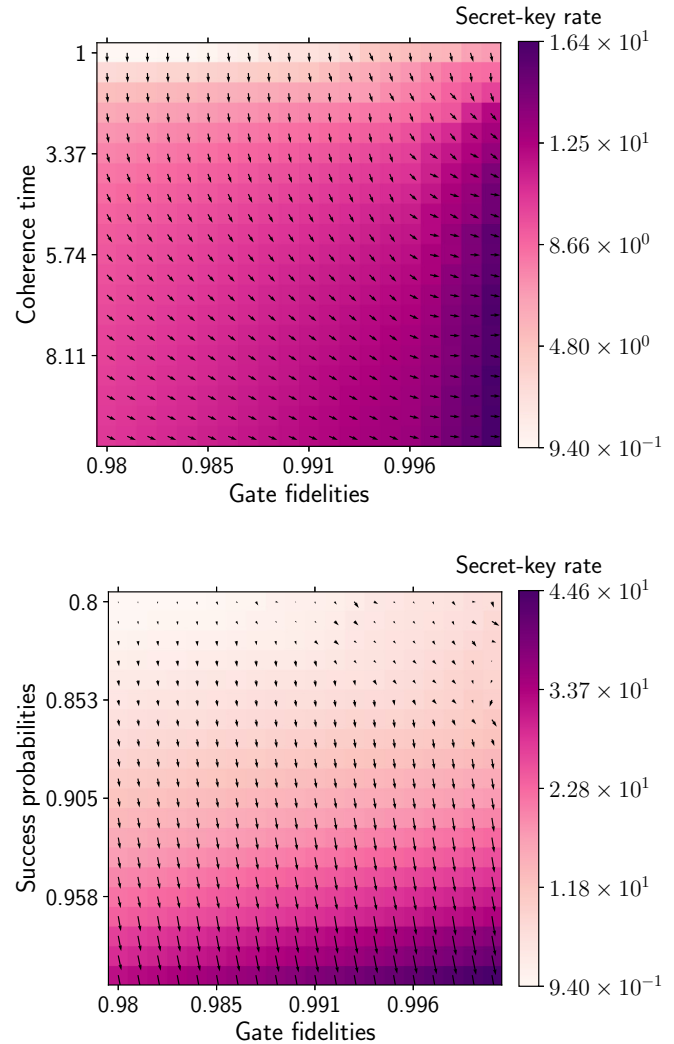


FIG. 11. Secret-key generation using the six-state protocol, for several different values of (a) the coherence times (1–10 s) and gate fidelities (0.98 to 1) and (b) the success probabilities (i.e., we vary  $p_{\text{det}} = p_{\text{em}} = p_{\text{ps}}$  simultaneously from 0.8 to 1) and gate fidelities (0.98 to 1) for a distance of 50 km and a single intermediate node for information processing platforms. Down and to the right in the plot indicate better parameters. All the other parameters are fixed to those of set 2 (Table II) and the base parameters (Table IV). We also plot the gradient, indicating the direction and magnitude of steepest ascent.

that, for almost all values of success probabilities and gate fidelities, it is impossible to generate a state with a fidelity of 0.9.

One of the near-term applications of a quantum repeater chain is the generation of a secret key. This motivates investigating the rate at which a secret key can be generated per unit time for several parameter ranges. Concretely, in Figs. 11(a) and 11(b) we investigate the same experimental settings and parameters as in Figs. 8 and 10. Each point corresponds to the maximum achieved secret key per unit time generated using a six-state protocol with advantage distillation [86] for each of the schemes in the output of our algorithm.

As in Fig. 8, we find in Fig. 11(a) that for increasing both the generation rate and secret-key rate increasing the coherence times is most beneficial only up to a certain point, after which the gate fidelities become more important. As in Fig. 10, we observe in Fig. 11(b) that almost always the success probabilities are more critical than the gate fidelities for increasing the secret-key rate.

## 2. Intermediate-distance entanglement generation using IP platforms

We expect the addition of nodes to become more beneficial as the distance over which entanglement is generated increases, conditioned on the fact that the experimental parameters are sufficiently high. In this section, we aim to quantify how good the experimental parameters need to be for this to be true. This motivates us to perform the heuristic optimization for the entanglement generation for greater distances, and with improved parameter sets. More concretely, we investigate the achieved generation times and fidelities for intermediate distances (i.e., 50 to 200 km) for the different experimental parameters proposed in Table II. We start with Fig. 12(a), where we reexamine the scenario of Fig. 7 of a total distance of 50 km. We now perform the heuristic optimization with parameter sets 2 and 3, where we consider implementing either no or a single intermediate node. It is clear from Fig. 12(a) that introducing a node over a distance of 50 km only improves the generation time by a modest amount for low fidelities, even with increased parameters. If we increase the total distance to 100 km, where we now also include parameter set 4, we find in Fig. 12(b) that a single node proves advantageous for almost all fidelities over all three considered parameter sets. In Figs. 12(c) and 12(d) we consider greater distances of 150 and 200 km, where we also include the heuristic optimization with two intermediate nodes. We observe that while having no node is clearly inferior to having at least one, introducing two nodes also outperforms a single node for most fidelities and sets of parameters for these distances. This suggests that the values of parameter set 3 (see Table II) are a relevant objective to reach for fast near-deterministic entanglement generation with information processing platforms.

We investigate the schemes for the above scenario of 200 km in Fig. 13, where we depict the schemes that achieve the lowest (nontrivial) fidelity and the highest fidelity. Interestingly, the scheme that achieves the highest fidelity requires that the different elementary pairs are generated both with the double- and single-click protocol. This exemplifies the need for including such asymmetric schemes in our optimization, which appears to become more important for higher fidelities.

The numerical investigation until this point has been dedicated to symmetric repeater chains. However, realistic quantum networks will be inhomogeneous and nodes will not be equally separated. In Fig. 26 in Appendix G we show the optimization results when considering an asymmetric repeater chain over 200 km with three intermediate nodes equally separated. The parameters used are parameter set 4 for the three intermediate nodes, and parameter set 2 for the nodes corresponding to Alice and Bob (see Table II). Such a situation can arise if the end users have access to different technology

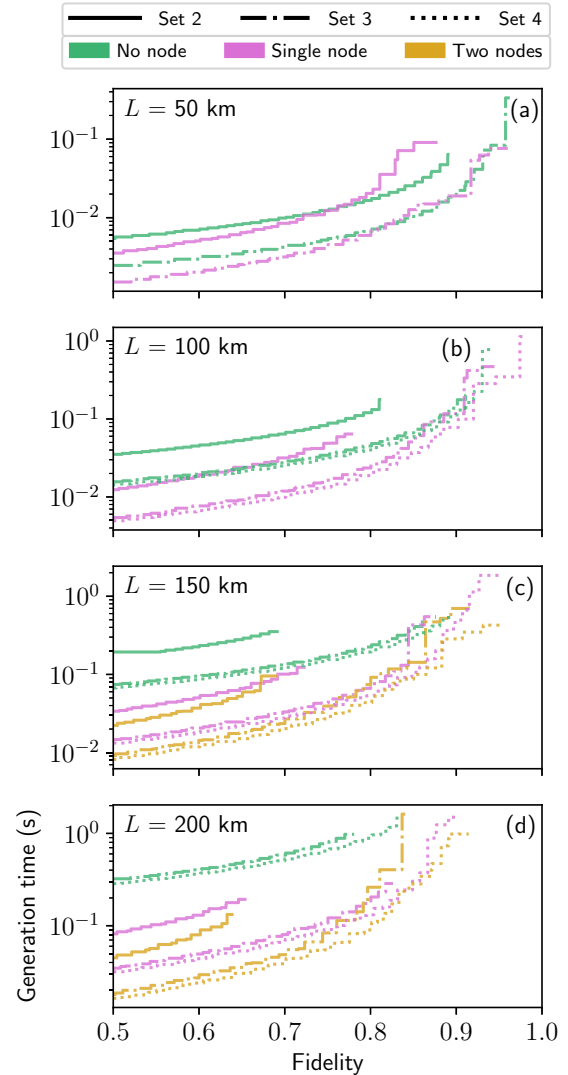


FIG. 12. Results of the achieved fidelity and generation time for total distances of 50 (a), 100 (b), 150 (c), and 200 (d) km using parameter sets 2 (solid), 3 (dashed-dotted), and 4 (dashed) (see Table II) for information processing nodes, where we consider having 0 (green), 1 (purple), or 2 (yellow) of such intermediate nodes.

than the network operator. In this setting, we compare the results of a full optimization with an optimization over BDCZ schemes, a class of schemes similar to the ones proposed in [13,14]. In particular, we include under the BDCZ class schemes that only combine identical pairs of schemes for connection and distillation. This class is different than the one in [20] as it allows optimization over the elementary pair generation protocols but, on the other hand, it does not include distillation schemes based on pumping [20]. We find that the full optimization gives an increased generation rate of up to a factor of 10 over BDCZ schemes.

## 3. Long-distance entanglement generation using IP platforms

Generating near-deterministic entanglement over larger distances requires excellent experimental control. It is not clear how the number of nodes and the experimental parameters affect our ability to generate entanglement. To this end,

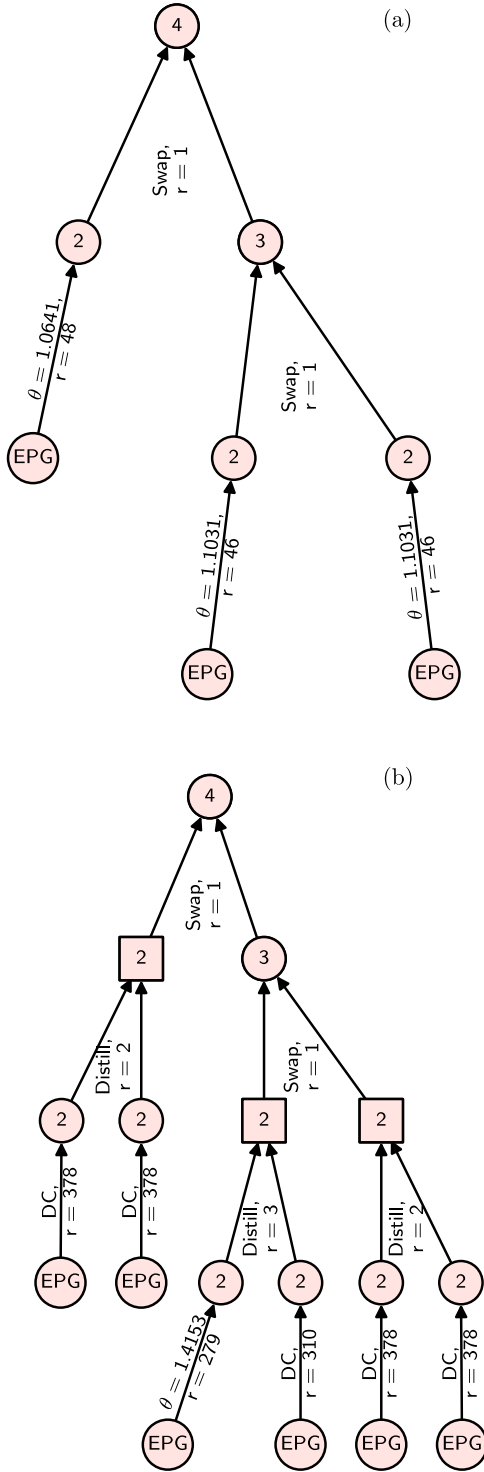


FIG. 13. Visual representation of the schemes with the lowest nontrivial fidelity (a) and highest fidelity (b), for a distance of 200 km with information processing platforms using parameter set 4 (see Table II) and two intermediate nodes or three hops. The numbers in the vertices indicate the number of nodes over which entanglement has been established. The “ $\theta = \theta^*$ ” indicates a single-click protocol with the  $\theta$  parameter set to  $\theta^*$ . The “ $r$ ” indicates the number of rounds the corresponding subtree is attempted. We find that the second scheme performs distillation between two elementary pairs generated with a single- and double-click protocol, demonstrating the benefit of including such distillation protocols in our optimization.

we consider here the generation of high fidelity entanglement over distances of 200, 400, 600, and 800 km. To gain an understanding of the relevant parameters, we study the effects of increasing gate fidelities and the memory coherence separately in Fig. 29 in Appendix G. We observe in Fig. 29 that increasing the coherence times yields a greater benefit than increasing the gate fidelities for these distances and parameters. In particular, increasing the coherence times allows for the generation of entanglement over larger distances, while increasing the gate fidelities effectively extends the ranges of fidelity over which entanglement is generated with the same generation time. We note here that the parameters  $p_{em}$ ,  $p_{pps}$ , and  $p_{det}$  (corresponding to the probability of emitting a photon from the memory, emitting in the correct mode or frequency, and detecting a photon successfully, respectively) remain fixed, which inhibits the potential benefits of including more nodes.

We have found that information processing platforms with sufficiently high parameters are a good candidate for near-term entanglement generation. In particular the success probabilities are an important factor for the generation of entanglement. However, even with multiple nodes, the maximum fidelity that can be reached is limited when attempting entanglement generation at large distances.

### B. Optimization results for MP platforms

Having investigated the performance of information processing platforms with regards to entanglement generation, we now explore entanglement generation with multiplexed elementary pair generation platforms. Not only are we interested in how well entanglement can be generated with a repeater chain built using a multiplexed elementary pair generation platform, but also in how the performance differs from information processing platforms. As explained in Sec. I, we expect that MP platforms perform better than IP platforms for larger distances, provided the experimental parameters are high enough. Our aim for this section is thus to investigate for which parameters and network configurations this becomes true.

First, let us discuss the set of protocols, the algorithm parameters, and the hardware parameters we will consider.

We consider a protocol for elementary pair generation with a tuneable parameter, one for swapping, and no protocol for distillation.

The elementary pair generation protocol (see Sec. III B) has one free parameter, the mean photon number  $N_s$ . Similar to information processing platforms, we also optimize over values of the mean photon number by considering a range of values of  $N_s$ . In this case, the range is from  $2 \times 10^{-4}$  to  $\frac{1}{2}(\sqrt{5 + \frac{2\sqrt{F_{\text{threshold}}(F_{\text{threshold}}+3)}}{F_{\text{threshold}}}} - 3)$ , in steps of  $10^{-4}$ . The lowest value of  $2 \times 10^{-4}$  was empirically found from the optimizations performed to be a good conservative lower bound, while the upper bound corresponds to achieving a fidelity of the elementary pair with fidelity equal to  $F_{\text{threshold}}$  when  $\eta \rightarrow 0$  [see Eq. (E1)] [87]. Setting  $\mathcal{F}_{\text{threshold}} = \frac{1}{2}$  leads to  $|\mathcal{E}| = 1039$ .

The swapping protocol is a photonic Bell state measurement with fixed efficiency depending on the number of ancillary photons (see Table IV). We thus have that  $|\mathcal{S}| =$

TABLE III. Base parameters used for the multiplexed elementary pair generation platforms considered in this paper.

$t_{\text{prep}}$ (entanglement preparation time)	$6 \mu\text{s}$
DcS (dark count rate)	10 per second
$L_0$ (attenuation length)	22 km
$n_{\text{RI}}$ (refractive index of the fiber)	1.44 [83]

1. Similar to the optimization with information processing platforms, to reduce the parameter space, we implement the banded swapping heuristic (see Sec. II D).

We use the same algorithm parameters as with the information processing platform optimization. For all of the schemes  $r$  ranges from  $r_{\text{min}}$  to  $r_{\text{max}}$  in (at most)  $r_{\text{discr}} = 200$  steps, where  $r_{\text{min}}$  and  $r_{\text{max}}$  are chosen such that the success probabilities are equal to  $p_{\text{min}}$  and  $p_{\text{max}}$ , respectively. We set  $\varepsilon_{\text{swap}} = \varepsilon_{\text{distill}} = 0.05$ ,  $\varepsilon_F = 0.01$ ,  $\varepsilon_p = 0.02$ , and  $p_{\text{min}} = 0.9$ . We consider only symmetric repeater chains, i.e., all the nodes have the same parameters and are equidistant.

Regarding the hardware parameters, the base parameters are given in Table III, while the four sets of parameters are given in Table IV.

### 1. Entanglement generation for short distances with MP platforms

We expect that multiplexed elementary pair generation platforms provide mostly a benefit over information processing platforms for larger distances. However, it is still of interest to investigate the performance of multiplexed elementary pair generation platforms for shorter distances. This is to gain an understanding of what can be done experimentally in the very near term. Thus, as in Sec. IV A 1, we first explore entanglement generation with MP platforms for short distances. We performed the heuristic optimization with parameter set 1 for distances of 15, 25, and 50 km, with zero, one, or two intermediate nodes. We found that, except for a distance of 15 km with no nodes, no entanglement could be generated. Even in the scenario of 15 km with no nodes, the maximum fidelity that could be generated was approximately 0.56. It is thus clear that, at least with the used parameters, information processing platforms are better than multiplexed elementary pair generation platforms for entanglement generation over short distances. We now investigate what are the relevant parameters to increase for MP platforms for entanglement generation over short distances. To this end, we perform a parameter exploration for a distance of 15 km. In particular, we vary the success probabilities and the efficiency coherence times from the values of parameter set 1 to those of set 2 in Table IV (see Fig. 14).

TABLE IV. The different sets of parameters considered for multiplexed elementary pair generation platforms in this paper.

	Set 1	Set 2	Set 3	Set 4
$T_{\text{coh}}$ (efficiency coherence times)	$10^{-2}$ s	$10^{-1}$ s	$10^0$ s	$10^1$ s
$N_{\text{modes}}$ (number of modes)	$10^4$	$10^5$	$10^6$	$10^7$
$p$ (success probabilities)	0.9	0.95	0.99	0.999
$p_{\text{BSM}}$ (BSM efficiency)	$\frac{1}{2}$	$\frac{3}{4}$	$\frac{7}{8}$	$\frac{15}{16}$

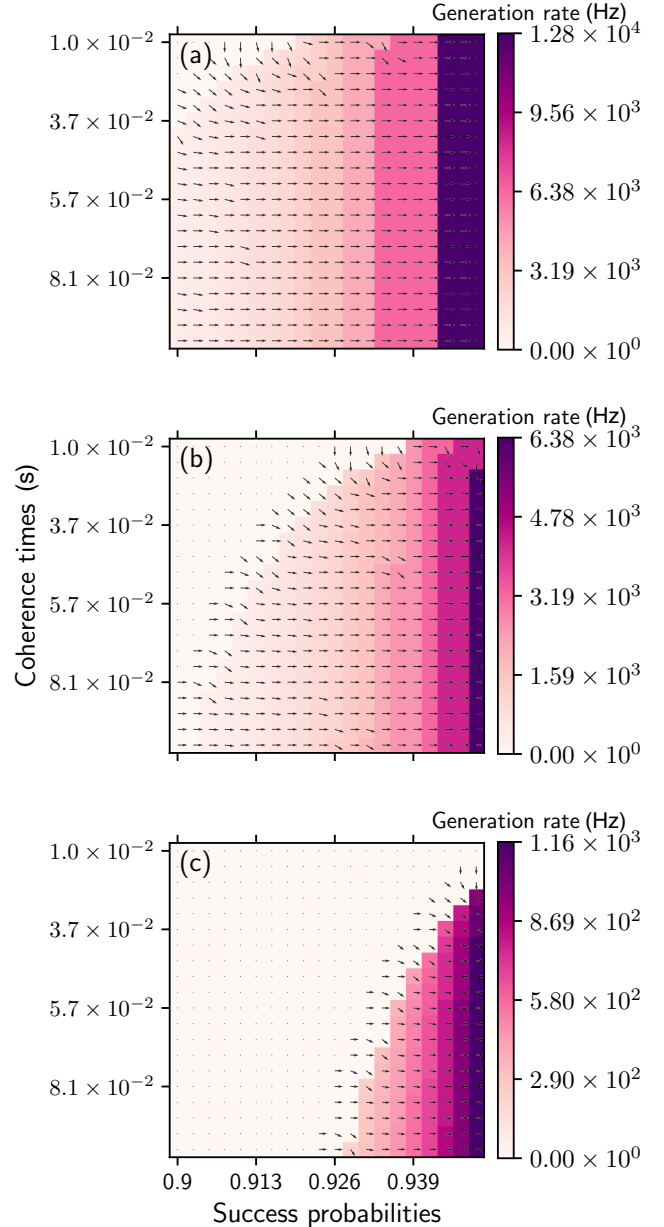


FIG. 14. Maximum generation rates for several different values of the success probabilities (i.e., we vary  $p_{\text{det}} = p_{\text{em}} = p_{\text{ps}}$  simultaneously) and efficiency coherence times, and for several different target fidelities, for a distance of 15 km and a single node for MP platforms. Down and to the right in the plot indicate better parameters. All the other parameters are fixed to those of set 2 (Table IV) or the base parameters (Table III). The target fidelities are (a)  $F = 0.7$ , (b)  $F = 0.8$ , and (c)  $F = 0.9$ , respectively. We also plot the gradient, indicating the direction and magnitude of steepest ascent.

We observe that with modest increases in the efficiency coherence times and success probabilities, entanglement generation becomes significantly more efficient. In particular, parameter set 1 (i.e., top left corner of the parameter plots) is only good enough for the generation of entanglement of very low fidelity ( $\approx 0.56$ ), while already a secret-key rate of  $\approx 500$  bits/s can be achieved for parameter set 2 (see Table IV). We conclude from the plots that, for current and



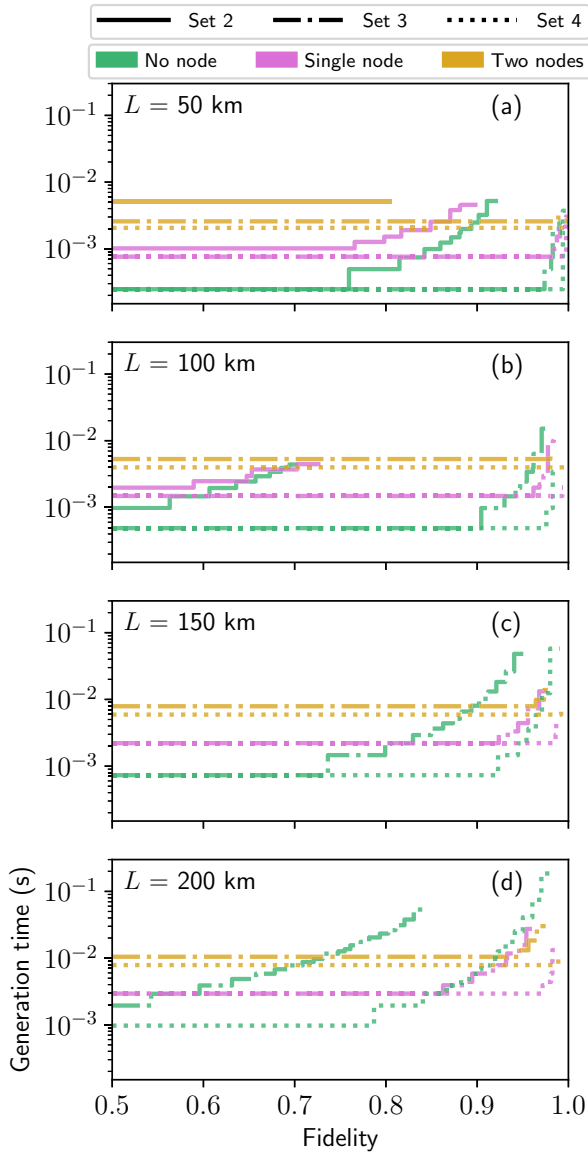


FIG. 15. Results of the achieved fidelity and generation time for total distances of 50 (a), 100 (b), 150 (c), and 200 (d) km using parameter sets 2 (solid), 3 (dashed-dotted), and 4 (dashed) (see Table IV) for multiplexed elementary pair generation platforms, where we consider having 0 (green), 1 (purple), or 2 (yellow) of such intermediate nodes.

near-term parameters and short distances, increasing the success probabilities is more important than increasing the efficiency coherence times.

## 2. Intermediate-distance entanglement generation using MP platforms

In the previous section we have found that at short distances MP platforms do not fare as well as IP platforms. This motivates us to investigate for which parameters and distances this does become the case. We thus investigate here entanglement distribution over distances of 50, 100, 150, and 200 km, where we consider the improved parameters found in sets 2, 3, and 4 in Tables III and IV in Fig. 15.

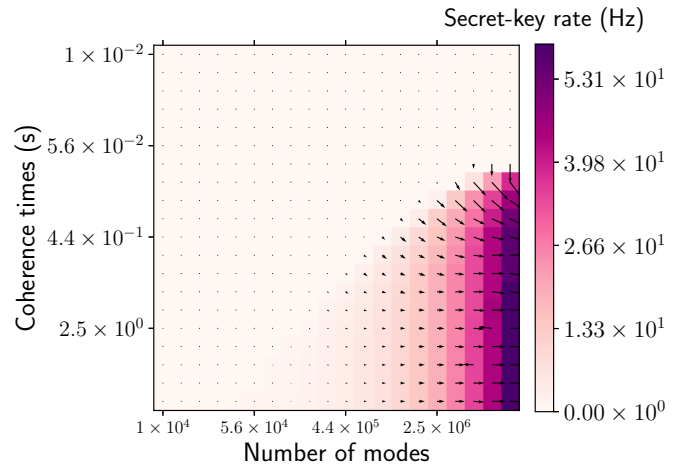


FIG. 16. Secret-key generation using the six-state protocol, for several different values of the efficiency coherence times ( $10^{-2}$ – $10^0$  s), and number of modes ( $10^4$ – $10^7$ ), for a distance of 200 km and a single node for MP platforms. Down and to the right in the plot indicate better parameters. All the other parameters are fixed to those of set 2 (Table IV) and the base parameters (Table III). We also plot the gradient, indicating the direction and magnitude of steepest ascent.

We find that, for most target fidelities in Figs. 15(a), 15(b) and 15(c), the generation time is relatively independent of the desired fidelity. We now explain this behavior. The fidelity is most strongly controlled by the parameter  $N_s$ —lowering  $N_s$  allows us to increase the fidelity, but lowers the success probability  $p$  of the elementary pair generation. However, the total success probability of generating at least one elementary pair  $1 - (1 - p)^{N_{\text{modes}}}$  does not decrease significantly, due to the large number of modes  $N_{\text{modes}}$ . In Appendix E we investigate how the minimum number of modes changes, as a function of the desired fidelity of the elementary pairs. We find that the required number of modes scales at least as  $\frac{\exp(\frac{L}{L_0})}{(1-F)^2}$ , where  $L$  is the distance between nodes and  $L_0$  is the attenuation length.

Since MP platforms are expected to have an advantage over IP platforms for longer distances, we investigate the secret-key rate per unit time for a total distance of 200 km (instead of 50 km for information processing platforms, see Figs. 8 and 11), where we vary the number of modes and the efficiency coherence time. In Fig. 16 we find that for most parameters the secret-key rate per unit time is zero. As in the previous parameter explorations performed, we observe that increasing the efficiency coherence times is only (strongly) beneficial up to a certain point (which depends on the number of modes in this case), after which increasing the efficiency coherence times further does not help. Interestingly, increasing the number of modes has the greatest effect on the secret key per unit time. Increasing the number of modes allows us to push the mean photon number to smaller numbers, effectively increasing the fidelity that can be generated within the same time window.

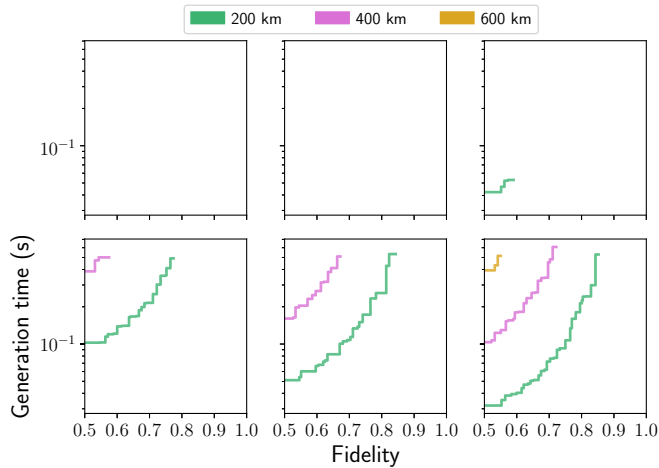


FIG. 17. Results of the heuristic optimization for total distances of 200, 400, and 600 km, for MP platforms and ten intermediate nodes. We use parameter set 2 for MP platforms (see Table III) as a baseline, where we set the success probability of the Bell state measurements to  $\frac{3}{4}$ ,  $\frac{7}{8}$ , and  $\frac{15}{16}$  in the first, second, and third column, respectively. We set the efficiency coherence time  $T_{\text{coh}}$  to 1 and 10 in the first and second row, respectively.

### 3. Long-distance entanglement generation with MP platforms

We observe by comparing Figs. 12 and 15 that MP platforms start to outperform IP platforms for distances of around  $\approx 200$  km. Here we are interested in whether multiplexed elementary pair generation platforms still perform well for even greater distances, which is the relevant scenario for large-scale quantum networks.

Let us first focus on the effect of the efficiency coherence times and Bell state measurement efficiency on long-distance entanglement generation. In Fig. 17 we investigate a repeater chain with ten nodes with the parameters from set 2; the success probabilities of the Bell state measurements given by  $\frac{3}{4}$ ,  $\frac{7}{8}$ , or  $\frac{15}{16}$  (corresponding to a number of ancillary photons 2, 6, and 14, respectively); and the efficiency coherence time  $T_{\text{coh}}$  set to 1 or 10. We find that even with the most optimistic parameters it is not possible to generate entanglement for distances of 800 km with ten nodes.

This leads to our results shown Fig. 18, where we plot the heuristic optimization results using parameter set 4, for distances of 200, 400, 600, and 800 km, and the number of nodes running from 1 to 4. We find indeed that, even for a distance of 800 km, entanglement can still be generated at a high fidelity ( $\approx 0.95$ ). This, combined with the fact that entanglement generation for the same distance is not possible in Fig. 17, suggests that it is essential to also increase the number of modes and the success probabilities to generate entanglement over large distances.

We give more detail of two schemes found from the optimization of Fig. 18. In particular, in Fig. 19 we give the schemes that achieve the lowest nontrivial fidelity and highest fidelity, respectively. As expected, the second scheme uses smaller values of the mean photon number  $N_s$  for the elementary pair generation. This increases the fidelity of the elementary pairs, at the cost of a lower success probability. Indeed, the number of attempts for the elementary pair gener-

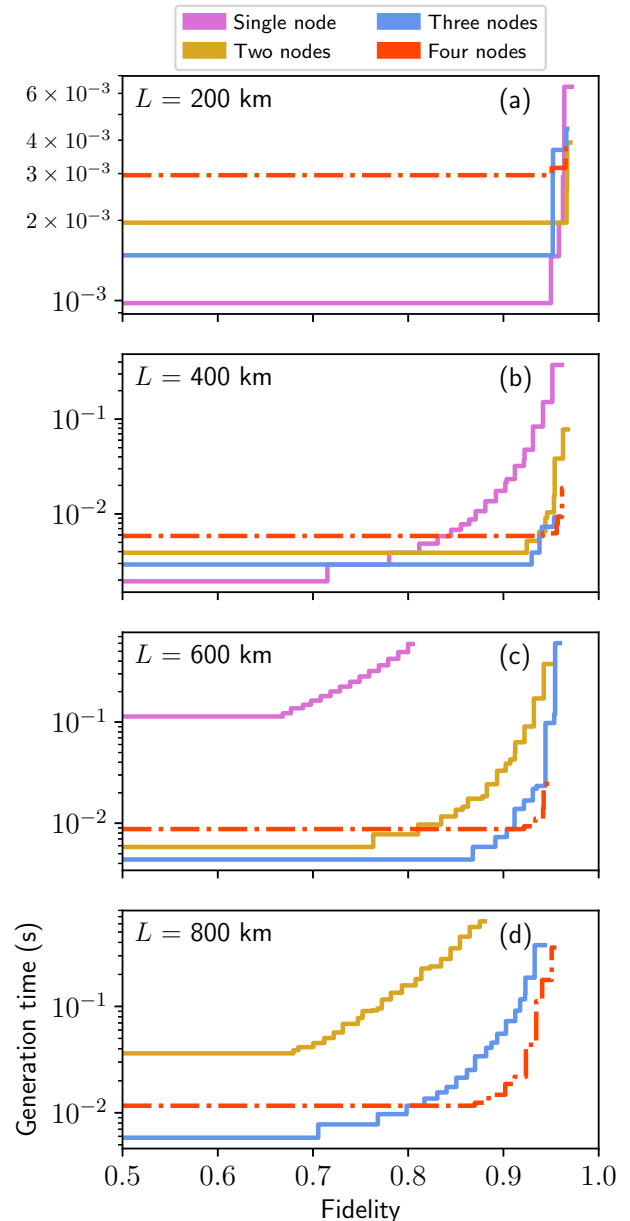


FIG. 18. Results of the achieved fidelity and generation time for total distances of 200 (a), 400 (b), 600 (c), and 800 (d) km using parameter set 4 (see Table IV) for MP platforms, where we consider having 1 (purple), 2 (yellow), 3 (blue), or 4 (orange) of such intermediate nodes.

ation ranges from 1 to 5 and from 8 to as high as 128, for the schemes in Figs. 19(a) and 19(b), respectively.

Here, we also note that there is a nontrivial interplay between the exponential decrease in output efficiency and performing more rounds (i.e., attempting more times to generate the elementary pairs) to increase the success probability. As we show in Appendix F, the requirement that each step succeeds with probability at least  $p_{\text{min}}$  can lead to a scenario where under a slight change of the network or parameters entanglement suddenly cannot be generated anymore.

Interestingly, we observe that the second scheme in Fig. 19 requires a swap between multihop links of lengths 5 and 2 at

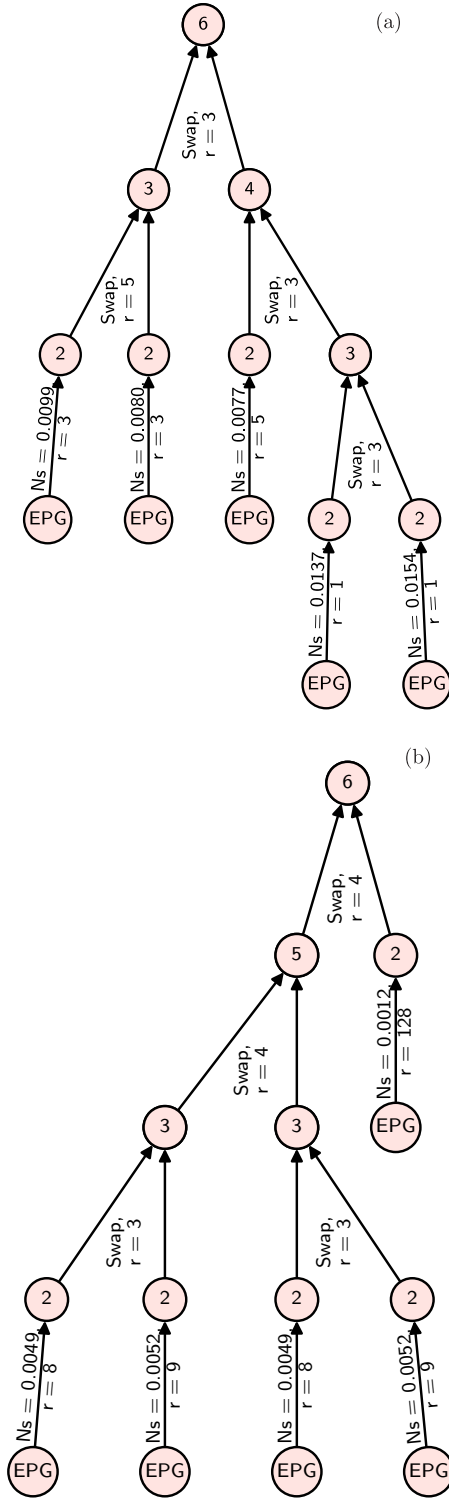


FIG. 19. Visual representation of the schemes with the lowest nontrivial fidelity (a) and highest fidelity (b), respectively, for a distance of 800 km with MP platforms using parameter set 4 (see Table IV) and four intermediate nodes or five hops. The “ $N_s = N_s^*$ ” indicates the elementary pair generation (EPG) protocol with mean photon number  $N_s^*$  used for MP platforms discussed in the main text. The “ $r$ ” here indicates the number of rounds the corresponding subtree is attempted. Note that the second scheme requires a swap between two schemes over multihop links of length 5 and 2 at the end.

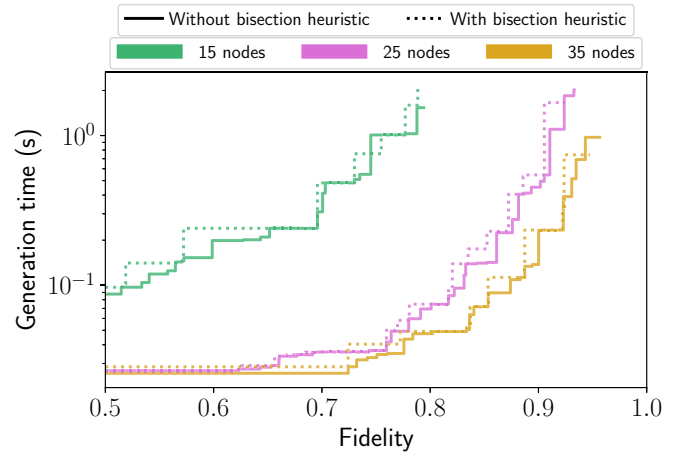


FIG. 20. Optimization results for a total distance of 4000 km, using a combination of multiplexed and information processing platforms. We use parameter sets 4 from both the multiplexed and information processing part of the platform (see Tables II and IV). The solid lines are the optimization without the bisection heuristic discussed in Sec. IID, while the dotted lines are with the bisection heuristic.

the end. This shows that, as with information processing platforms, exploring more complex asymmetric schemes provides a benefit over more simplistic schemes.

**C. Long-distance entanglement generation using a combination of IP and MP platforms**

Here we investigate combining the strengths of IP platforms with those of MP platforms. For this, we generate the elementary pairs with MP platforms, after which all the operations are performed with IP platforms. We optimize then over the same protocols as was done for IP and MP platforms (see Secs. IVA and IVB). We expect that, with sufficiently good parameters, the combination of the two outperforms the individual platforms, and we can distribute entanglement over significantly larger distances.

Using the parameter set 4 of both platforms, we plot the results for 15, 25, and 35 nodes in Fig. 20, for a total distance of 4000 km. Furthermore, we also plot a comparison here when the optimization includes the bisection heuristic (see Sec. IID).

From Fig. 20 we observe that, by combining both the strengths from multiplexing and information processing platforms, it is possible to generate entanglement with a high fidelity near-deterministically over large distances by using a large number of nodes. We find that the optimization results with the bisection heuristic are similar to the results without, while being significantly faster to perform. We find for the cases of 15, 25, and 35 intermediate nodes that the algorithm runtime drops from an order of magnitude of  $\approx 100$  min to  $\lesssim 10$  min. We thus find that the bisection heuristic allows for a faster heuristic optimization, without the resultant schemes becoming significantly worse than without the bisection heuristic.

We conclude our results with a plot comparing entanglement generation with the three implementations considered in

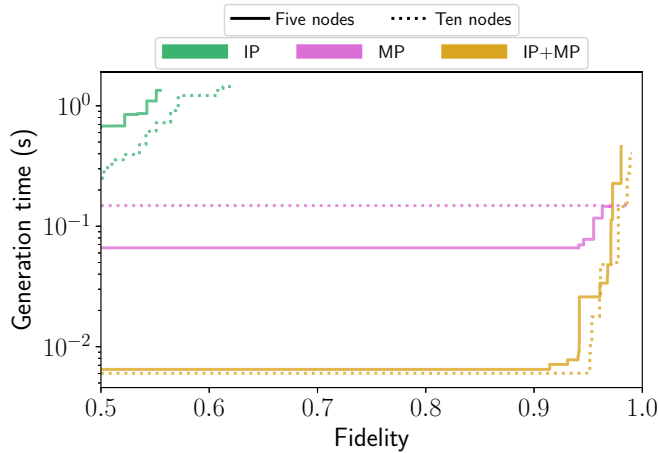


FIG. 21. Results of the heuristic optimization for a total distance of 800 km, where we compare the three implementations considered in this paper, using five (solid) or ten (dashed) intermediate nodes. We use parameter sets 4 from information processing (IP) platforms, multiplexed elementary pair generation (MP) platforms, and the combination of the two (IP+MP). The two crosses in the plot indicate the schemes depicted in Figs. 27 and 28, respectively.

this paper for a distance of 800 km and five or ten intermediate nodes. We find in Fig. 21 that, for large distances, the combination of IP and MP platforms outperforms the individual platforms. In fact, it can generate target fidelities below  $\approx 0.9$  an order of magnitude faster than the MP platform. We see that, as expected, information processing platforms perform significantly worse, where the maximum fidelity is limited to  $\approx 0.6$ . This is due to the effects of losses during elementary pair generation becoming too strong. This can of course be counteracted by using more nodes, but this results in too much decoherence. This suggests that, for large distances, MP platforms outperform information processing platforms for near-deterministic entanglement generation.

We depict the two schemes corresponding to the two crosses found in Fig. 21 in Figs. 27 and 28 in Appendix G, respectively. The first of these (blue cross) corresponds to the lowest nontrivial fidelity achieved, while the second one (red cross) corresponds to a state with fidelity  $F = 0.9605$ , generated in time  $T = 17.7$  ms. A higher fidelity was not chosen, due to those schemes becoming too big to fit on a page, demonstrating the nontrivial nature of the optimization performed here.

## V. CONCLUSIONS

The future quantum internet has the potential to change our information society by enabling the implementation of quantum communication tasks. For many of these tasks the key resource is the availability of high fidelity entanglement at the necessary rates. However, given the complex relation between experimental parameters, entanglement distribution protocols, and quantum network design, it is unclear what are the necessary parameters to distribute entanglement except for the most basic scenarios. Here, we develop an algorithm to partially answer this question. In particular, our algorithm

optimizes the near-deterministic distribution of entanglement over chains of quantum repeaters which are abstractly characterized by a small set of relevant parameters.

Even in this abstract setting, the number of possible protocols for a given quantum repeater chain is too large to attempt brute-force optimization. To make optimization feasible, we introduce a number of heuristics that render optimization feasible by dramatically reducing the runtime of the algorithm. Moreover, the heuristics can also be interpreted as approximate rules for protocol design as numerical results show that optimal protocols follow the heuristics. We could expect these heuristics to apply to more dynamic schemes, where the information of the current present entanglement in the network is used to make decisions on the fly by the network.

Any realistic quantum repeater network will be asymmetric in the distances between the nodes and the experimental parameters. We have applied our algorithm to an asymmetric repeater chain, and have found that our optimization results strongly outperform the results from a simplified optimization over symmetric or hierarchical schemes, such as those presented in [13,14].

We have used the algorithm not only for optimizing entanglement distribution, but also for parameter exploration. In particular, we have optimized entanglement distribution for several parameter regimes investigating the most relevant parameters for both information processing and multiplexed elementary pair generation platforms. For both, we find that success probabilities (e.g., the emission probabilities, detector efficiencies, etc.) have a strong impact on performance.

In contrast with previous work, our focus on near-deterministic schemes allowed us to make exact statements about the generation time and fidelities of the distributed states. The ability to deliver states with high probability at specific times could be of benefit for routing entanglement in a network.

In conclusion, here we have developed an algorithm that allows one to efficiently optimize and explore the parameter space for near-deterministic entanglement distribution over repeater chains. We have investigated a number of representative platforms but the algorithm is not particular to these choices. We make the source code publicly available [25] to facilitate the investigation of other implementations, parameters, and/or error models.

## ACKNOWLEDGMENTS

The authors would like to thank Filip Rozpędek, Tim Coopmans, Matthew Skrzypczyk, Joshua Slater, Guus Avis, Mohsen Falamarzi, Daniël Bouman, and Gláucia Murta Guimarães for helpful discussions for this project. This work was supported by the Dutch Technology Foundation, the Netherlands Organization for Scientific Research (NWO) through a VIDI grant (S.W.), the European Research Council through a starting grant (S.W.), the QIA project (funded by European Union's Horizon 2020, Grant No. 820445), and the Netherlands Organization for Scientific Research (NWO/OCW) as part of the Quantum Software Consortium program (Project No. 024.003.037/3368).

## APPENDIX A: COMPLEXITY OF THE ALGORITHM

Here we discuss the complexity of the algorithm. For this, first we bound from below the number of schemes that a brute-force approach without heuristics would need to explore. We then incorporate the heuristics and derive an upper bound on the number of schemes of the algorithm as described in Sec. II D. We finalize by deriving an upper bound on the number of schemes in the particular case of “symmetric” repeater chains, that is, chains where each node has the same parameters and adjacent nodes are connected by identical elementary links.

### 1. Lower bound on the complexity of the brute-force algorithm

Here we derive two lower bounds on the number of schemes considered by a brute-force algorithm. The two lower bounds are given by  $O[(r_{\text{discr}}|\mathcal{E}||\mathcal{S}||\mathcal{D}|)^{2^{mn}}]$  and  $O\{[(r_{\text{discr}})^2|\mathcal{E}||\mathcal{S}|]^n\}$ . These bounds correspond to the case with and without distillation protocols considered, respectively. Here,  $n$  denotes the number of elementary links in the repeater chain (i.e., one less than the number of nodes),  $m$  denotes the maximum number of distillation rounds,  $r_{\text{discr}}$  denotes the maximum different values of the number of attempts, and  $|\mathcal{E}|$ ,  $|\mathcal{S}|$ , and  $|\mathcal{D}|$  denote the number of elementary pair generation, swapping, and distillation protocols, respectively.

To make the analysis tractable, while still obtaining a strict lower bound on the number of schemes, we analyze a simpler algorithm that explores a reduced set of swapping schemes. At level  $i$ , instead of exploring all combinations of swapping between schemes for every pair of adjacent (multihop) links with a combined length  $i$ , this algorithm only considers swapping between schemes on the leftmost link of length  $i - 1$  and one of length 1, i.e., the entanglement is propagated by one elementary link at each level. Furthermore, we will assume the worst-case scenario, where all generated schemes have success probability greater than  $p_{\text{min}}$ , meaning that all of them will be stored.

We first present a sketch of our derivation of the lower bound. We find two maps  $f_{\text{swap}}$  and  $f_{\text{distill}}$  which send the number of schemes  $\zeta$  before the swap operation and  $m$  distilla-

tion rounds to a lower bound on the number of schemes after the swap and distillations, respectively. Denoting by  $\zeta_{\text{init}}$  the number of schemes over an elementary link after distillation, a lower bound for the number of schemes after two hops is then given by

$$(f_{\text{distill}} \circ f_{\text{swap}})(\zeta_{\text{init}}). \quad (\text{A1})$$

Similarly, after  $n - 1$  hops we find the following lower bound:

$$[(f_{\text{distill}} \circ f_{\text{swap}})]^{n-1}(\zeta_{\text{init}}). \quad (\text{A2})$$

In what follows, we will find the maps  $f_{\text{swap}}$ ,  $f_{\text{distill}}$ , and  $\zeta_{\text{init}}$ . The map  $f_{\text{swap}}$  will depend implicitly on  $r_{\text{discr}}$ ,  $|\mathcal{S}|$ , and  $\zeta_{\text{init}}$ , while  $f_{\text{distill}}$  depends implicitly on  $r_{\text{discr}}$ ,  $|\mathcal{D}|$ , and  $m$ .

Let us start with  $f_{\text{swap}}$ . As mentioned above, for each multihop link of length  $i$ , the simplified algorithm combines each of the  $\zeta$  schemes of the multihop link of length  $i - 1$  with each of the  $\zeta_{\text{init}}$  schemes stored for an elementary link. We obtain the following map on the number of schemes:

$$\zeta \xrightarrow{f_{\text{swap}}} r_{\text{discr}}|\mathcal{S}|\zeta_{\text{init}}\zeta. \quad (\text{A3})$$

Let us now find the map  $f_{\text{distill}}$ . Assuming we start with  $\zeta$  schemes, each of the  $|\mathcal{D}|$  distillations will generate a new scheme for each possible pair of schemes, and for each of the  $r_{\text{discr}}$  possible values of attempts. Thus, after a single distillation round, we end up with  $r_{\text{discr}}|\mathcal{D}|\zeta^2 + \zeta \geq r_{\text{discr}}|\mathcal{D}|\zeta^2$  schemes, where we have only kept the schemes which had a distillation step at the end.

We can now repeat the above for  $m = 2$  distillation rounds, by setting  $\zeta = r_{\text{discr}}|\mathcal{D}|\zeta^2$ . We then find that for  $m = 2$  distillation rounds a lower bound is given by  $r_{\text{discr}}|\mathcal{D}|(r_{\text{discr}}|\mathcal{D}|\zeta^2)^2$ . In general, after  $m$  distillation rounds we find that

$$\zeta \xrightarrow{f_{\text{distill}}} (r_{\text{discr}}|\mathcal{D}|)^{2^m-1}\zeta^{2^m}. \quad (\text{A4})$$

Thus, starting from a number  $\zeta$  of schemes, the composition of swapping with a scheme on an elementary link [Eq. (A3)] and then distilling [Eq. (A4)] gives us the following map for the lower bound on the number of schemes:

$$\zeta \xrightarrow{f_{\text{distill}} \circ f_{\text{swap}}} (r_{\text{discr}}|\mathcal{D}|)^{2^m-1}(r_{\text{discr}}|\mathcal{S}|\zeta_{\text{init}}\zeta)^{2^m} = \Omega\zeta^{2^m}, \quad (\text{A5})$$

where we define  $\Omega \equiv (r_{\text{discr}}|\mathcal{D}|)^{2^m-1}(r_{\text{discr}}|\mathcal{S}|\zeta_{\text{init}})^{2^m}$ , which is independent of  $\zeta$ .

Repeating the above map in Eq. (A5)  $n - 1$  times on  $\zeta_{\text{init}}$  (the number of schemes stored over an elementary link) yields the following lower bound:

$$\zeta_{\text{init}} \xrightarrow{f_{\text{distill}} \circ f_{\text{swap}}} \Omega(\zeta_{\text{init}})^{2^m} \xrightarrow{f_{\text{distill}} \circ f_{\text{swap}}} \Omega[\Omega(\zeta_{\text{init}})^{2^m}]^{2^m} \xrightarrow{f_{\text{distill}} \circ f_{\text{swap}}} \dots = \Omega^{1+2^m+2^{2m}+\dots+2^{m(n-2)}}\zeta_{\text{init}}^{2^{m(n-1)}} = \Omega^{\frac{2^{m(n-1)}-1}{2^m-1}}(\zeta_{\text{init}})^{2^{m(n-1)}}. \quad (\text{A6})$$

The only ingredient missing from our analysis now is  $\zeta_{\text{init}}$ , the number of schemes on an elementary link after  $m$  distillation rounds. First, for elementary pair generation there are  $r_{\text{discr}}$  different values of attempts per elementary pair generation protocol. In other words, for each of the  $|\mathcal{E}|$  elementary pair generation protocols that can be performed, there are  $r_{\text{discr}}$  different choices of  $r$ , leading to a total of  $|\mathcal{E}|r_{\text{discr}}$  schemes. To find the number of schemes after  $m$  distillation rounds we apply our map  $f_{\text{distill}}$  to  $|\mathcal{E}|r_{\text{discr}}$  and find that

$$\zeta_{\text{init}} \equiv (r_{\text{discr}}|\mathcal{D}|)^{2^m-1}(|\mathcal{E}|r_{\text{discr}})^{2^m}. \quad (\text{A7})$$

Inserting Eq. (A7) into (A6) and expanding gives

$$\begin{aligned} \Omega^{\frac{2^{m(n-1)}-1}{2^m-1}}(\zeta_{\text{init}})^{2^{m(n-1)}} &= \{(r_{\text{discr}}|\mathcal{D}|)^{2^m-1}[r_{\text{discr}}|\mathcal{S}|((r_{\text{discr}}|\mathcal{D}|)^{2^m-1}(|\mathcal{E}|r_{\text{discr}})^{2^m})^{2^m}]^{\frac{2^{m(n-1)}-1}{2^m-1}}[(r_{\text{discr}}|\mathcal{D}|)^{2^m-1}(r_{\text{discr}}|\mathcal{E}|)^{2^m}]^{2^{m(n-1)}}\} \\ &= (r_{\text{discr}})^{(2^{m(n-2)})(4^{m+1}-1)}|\mathcal{E}|^{(2^{m(n-1)})(2^{m+1})}|\mathcal{S}|^{2^{m(n-1)}}|\mathcal{D}|^{(2^{m(n-2)})(2^{2m+1}-2^m-1)} = O[(r_{\text{discr}}|\mathcal{E}||\mathcal{S}||\mathcal{D}|)^{2^{mn}}]. \end{aligned} \quad (\text{A8})$$

We note that this bound becomes trivial when no distillation is performed, i.e.,  $m = 0$ . This is due to the fact that lower-order terms were ignored in the number of schemes after distilling. We treat the  $m = 0$  case separately here. For the case of no distillation, we perform  $r_{\text{discr}}|\mathcal{S}|$  different swap protocols for  $n - 1$  times. Since we start with a total number of  $r_{\text{discr}}|\mathcal{E}|$  schemes on the elementary links, the total number of schemes is then given by

$$(r_{\text{discr}}|\mathcal{S}|)^{n-1}(r_{\text{discr}}|\mathcal{E}|)^n = (r_{\text{discr}})^{2n-1}(|\mathcal{E}||\mathcal{S}|)^{n-1} = O\{[(r_{\text{discr}})^2|\mathcal{E}||\mathcal{S}|]^n\}. \quad (\text{A9})$$

We see that with distillation (i.e.,  $m \geq 1$ ) the number of schemes to consider grows superexponentially in the number of elementary links  $n$ , while without distillation  $m = 0$  the number of schemes grows exponentially. It is clear that a brute-force optimization becomes infeasible for any reasonable number of protocols (i.e.,  $|\mathcal{E}|$ ,  $|\mathcal{S}|$ ,  $|\mathcal{D}|$ ), number of distillation rounds  $m$ , and elementary links  $n$ .

## 2. Upper bound on the complexity of the heuristic algorithm

In this section we consider the complexity with the heuristics implemented. The upper bound we find scales as  $O[n^2 \log_2(n)]$  for an arbitrary repeater chain, where  $n$  is the number of elementary links in the repeater chain. As discussed in the main text, the optimization can be simplified for the scenario of a repeater chain where every node has exactly the same parameters and the distance between each of the repeaters is equal. For such a symmetric repeater chain, we find a scaling of  $O[n \log_2(n)]$ .

Let us first briefly discuss the effects the heuristics have on the complexity, before upper bounding the number of schemes. First, we note here that in the worst-case scenario all the schemes are incomparable, leading to no pruning. Second, the coarse-graining of the fidelity and probability imposes an upper limit on the considered schemes. The coarse-graining fixes the maximum stored schemes to be  $\lceil \frac{(1-F_{\text{threshold}})}{\varepsilon_F} \rceil \lceil \frac{(1-p_{\text{min}})}{\varepsilon_p} \rceil$  per (multihop) link. For instance, the number of schemes for elementary pair generation does not change with the heuristic, namely, it remains  $n|\mathcal{E}|r_{\text{discr}}$  in total. However, at most  $\lceil \frac{(1-F_{\text{threshold}})}{\varepsilon_F} \rceil \lceil \frac{(1-p_{\text{min}})}{\varepsilon_p} \rceil$  of these are stored per elementary link.

Let us now consider swapping. The algorithm restricts the creation of a scheme on a multihop link of length  $i$  (equivalently, a link requiring  $i$  hops) to swapping two links of length  $\frac{i}{2} \pm \log_2(i - 1)$ , leading to at most  $2\lfloor \log_2(i - 1) \rfloor + 1$  different options [see Eq. (5)]. The banded swapping heuristic further reduces swapping to schemes that verify Eq. (6) from the main text:

$$\left| \frac{\log_2(F_1)}{i_1} - \frac{\log_2(F_2)}{i_2} \right| \leq \varepsilon_{\text{swap}}.$$

This equation becomes in the asymptotic limit

$$\frac{F_1}{F_2} \leq \exp\left(\frac{i\varepsilon_{\text{swap}}}{2}\right),$$

since  $i_1 \sim i_2 \sim \frac{i}{2}$ , and where we have assumed without loss of generality that  $F_1 \geq F_2$ . Now note that  $F_{\text{threshold}} \geq \frac{1}{2}$ , which implies that  $\frac{1}{2} \leq \frac{F_1}{F_2} \leq 2$ . We thus have that, in the asymptotic limit, the banded swapping heuristic becomes void if  $\varepsilon_{\text{swap}}$  is fixed.

This means that asymptotically the algorithm considers the full  $r_{\text{discr}} \frac{(1-F_{\text{threshold}})^2(1-p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2}$  schemes for swapping.

The last heuristic is banded distillation. For a fixed distillation protocol, it reduces the number of schemes for performing distillation to at most  $2\lceil \frac{\varepsilon_{\text{distill}}}{\varepsilon_F} \rceil \lceil \frac{1-p_{\text{min}}}{\varepsilon_p} \rceil$ . Since there are  $\lceil \frac{(1-F_{\text{threshold}})}{\varepsilon_F} \rceil \lceil \frac{(1-p_{\text{min}})}{\varepsilon_p} \rceil$  stored schemes, we find the following upper bound on the considered schemes for distillation for a single multihop link,  $2r_{\text{discr}}|\mathcal{D}|m \frac{\varepsilon_{\text{distill}}(1-F_{\text{threshold}})(1-p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2}$ . We have removed here and in what follows the ceiling functions, since we are interested in the asymptotic complexity and increasing readability.

Combining the previous arguments, we find the following upper bound:

$$\begin{aligned} & n|\mathcal{E}|r_{\text{discr}} + r_{\text{discr}} \sum_{i=2}^n (n-i+1) \left( (2\lfloor \log_2(i-1) \rfloor + 1)|\mathcal{S}| \frac{(1-F_{\text{threshold}})^2(1-p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2} \right. \\ & \left. + 2m|\mathcal{D}| \frac{\varepsilon_{\text{distill}}(1-F_{\text{threshold}})(1-p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2} \right) \\ & = r_{\text{discr}} \left( n|\mathcal{E}| + \frac{(1-F_{\text{threshold}})(1-p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2} \sum_{i=2}^n (n-i+1)|\mathcal{S}|((2\lfloor \log_2(i) \rfloor + 1)(1-F_{\text{threshold}}) + 2m|\mathcal{D}|\varepsilon_{\text{distill}}) \right) \\ & \sim r_{\text{discr}} \left( n|\mathcal{E}| + \frac{(1-F_{\text{threshold}})(1-p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2} \left( 2(1-F_{\text{threshold}})|\mathcal{S}|n^2 \log_2(n) + 2m|\mathcal{D}|\varepsilon_{\text{distill}} \frac{n^2}{2} \right) \right) \\ & = r_{\text{discr}} \left( n|\mathcal{E}| + n^2 \frac{(1-F_{\text{threshold}})(1-p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2} (2(1-F_{\text{threshold}})|\mathcal{S}| \log_2(n) + m|\mathcal{D}|\varepsilon_{\text{distill}}) \right). \quad (\text{A10}) \end{aligned}$$

We observe that the algorithm is  $O[n^2 \log_2(n)]$ , where the prefactor is given by  $2|\mathcal{S}|r_{\text{discr}} \frac{(1-F_{\text{threshold}})(1-p_{\text{min}})^2}{\varepsilon_F \varepsilon_p}$ .

In the case of a symmetric repeater chain (i.e., every node has exactly the same parameters and the distance between each of the repeaters is equal) we can simplify the optimization by exploiting the symmetry. That is, the optimization done over a (multi-hop) link of length  $i$  only needs to be done once, as opposed to  $n - i + 1$  times in the general setting. Furthermore, there are only  $\lfloor \log_2(i - 1) \rfloor + 1$  unique ways to perform swapping. The number of schemes is then upper bounded by

$$|\mathcal{E}|r_{\text{discr}} + r_{\text{discr}} \sum_{i=2}^n \left( \lfloor \log_2(i - 1) \rfloor + 1 \right) |\mathcal{S}| \frac{(1 - F_{\text{threshold}})^2 (1 - p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2} + 2m|\mathcal{D}| \frac{\varepsilon_{\text{distill}} (1 - F_{\text{threshold}}) (1 - p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2} \Big) \sim r_{\text{discr}} |\mathcal{E}| + r_{\text{discr}} \frac{(1 - F_{\text{threshold}}) (1 - p_{\text{min}})^2}{(\varepsilon_F \varepsilon_p)^2} ((1 - F_{\text{threshold}}) |\mathcal{S}| n \log_2(n) + nm|\mathcal{D}| \varepsilon_{\text{distill}}). \tag{A11}$$

We observe that for the symmetric scenario the algorithm is  $O[n \log_2(n)]$ , where the prefactor is given by  $r_{\text{discr}} |\mathcal{S}| \left( \frac{(1 - F_{\text{threshold}}) (1 - p_{\text{min}})^2}{\varepsilon_F \varepsilon_p} \right)^2$ . As mentioned before, the algorithm developed supports both the general and symmetric case.

**APPENDIX B: ANALYSIS OF THE HEURISTICS**

The algorithm detailed in this paper uses four different parameters to reduce the search space of the optimization, namely,  $\varepsilon_F$ ,  $\varepsilon_p$ ,  $\varepsilon_{\text{swap}}$ , and  $\varepsilon_{\text{distill}}$  (see Sec. IID). The parameters  $\varepsilon_F$  and  $\varepsilon_p$  are responsible for the coarse-graining in the algorithm, while the parameters  $\varepsilon_{\text{swap}}$  and  $\varepsilon_{\text{distill}}$  govern the restrictions on the states used for swapping and distillation, respectively. In this section we investigate the heuristics and how they affect the algorithm runtime and accuracy of the optimization, which we use to settle on values for  $\varepsilon_F$ ,  $\varepsilon_p$ ,  $\varepsilon_{\text{swap}}$ , and  $\varepsilon_{\text{distill}}$ . The objective here is to find a good tradeoff between the algorithm runtime and the accuracy of the algorithm. We first investigate the coarse-graining, i.e., we vary  $\varepsilon_F$  and  $\varepsilon_p$ . Afterwards, we investigate the effects of  $\varepsilon_{\text{swap}}$  and  $\varepsilon_{\text{distill}}$  on the optimization results. Finally, we compare the banded swapping heuristic with the naive heuristic, i.e., where we require the two states to be close in fidelity.

We first vary  $\varepsilon_F$  and  $\varepsilon_p$  simultaneously when optimizing over schemes for a distance of 6 km and a single repeater with

the IP parameter set 2, of which the results can be seen in Figs. 22 and 23. As expected, there is a tradeoff between the accuracy of the algorithm and its running time as  $\varepsilon_F$  and  $\varepsilon_p$  are varied. While a good tradeoff between the accuracy and the runtime depends on each specific case, we use these results to settle in this paper for  $\varepsilon_F = 0.01$  and  $\varepsilon_p = 0.02$ . We settle for these parameters since the important characteristics of the generation time as a function of the fidelity appear to be similar when a more fine-grained optimization is implemented, without the runtime becoming infeasible.

In Figs. 24(a) and 24(b) we perform an optimization for several different values of  $\varepsilon_{\text{swap}}$  and  $\varepsilon_{\text{distill}}$ , using parameter set 4 with four intermediate nodes for a distance of 300 km. In Fig. 24(a) we use the banded swapping heuristic [see Eq. (6)], while in Fig. 24(b) we only swap between states that are  $\varepsilon_{\text{swap}}$  close in fidelity. We observe that the optimization results in Fig. 24(b) are significantly worse than those in Fig. 24(a), while Fig. 25 indicates that the runtimes are comparable for both heuristics. We use these results to settle on  $\varepsilon_{\text{swap}} = \varepsilon_{\text{distill}} = 0.05$ . Furthermore, we find that the heuristic plays primarily a role for smaller fidelities. This implies that only for

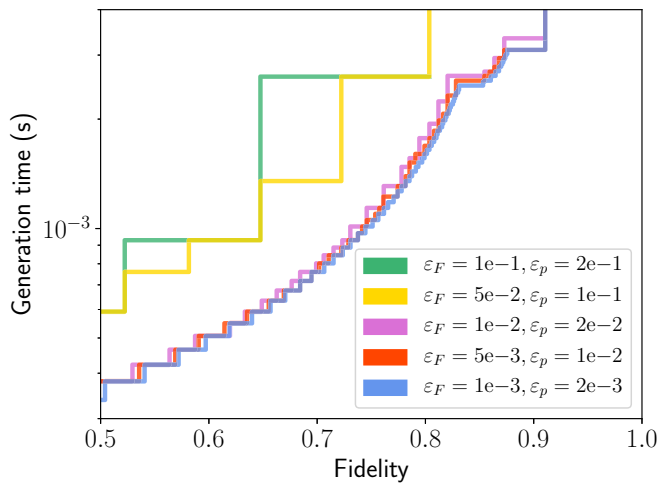


FIG. 22. Optimized schemes for a distance of 15 km using a single node with the IP parameter set 2 (see Table II) for several different pairs of  $\varepsilon_F$  and  $\varepsilon_p$ . Note that as  $\varepsilon_F$  and  $\varepsilon_p$  approach zero, the curves converge.

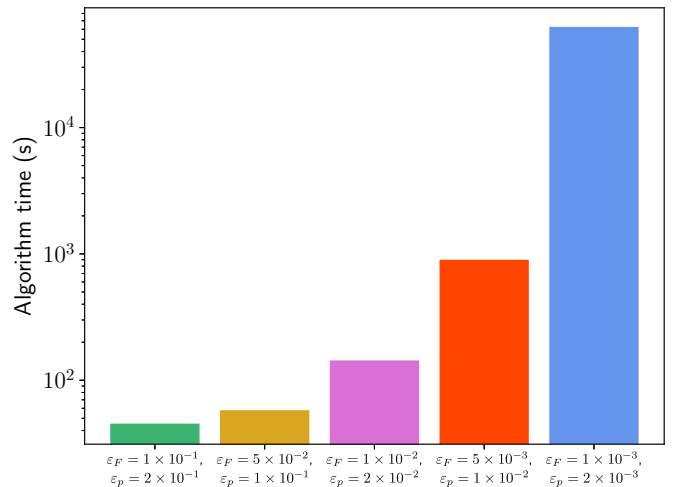


FIG. 23. The runtime of the algorithm for the optimizations performed in Fig. 22. Notice the logarithmic scale, indicating the strong growth rate as  $\varepsilon_F$  and  $\varepsilon_p$  become smaller.

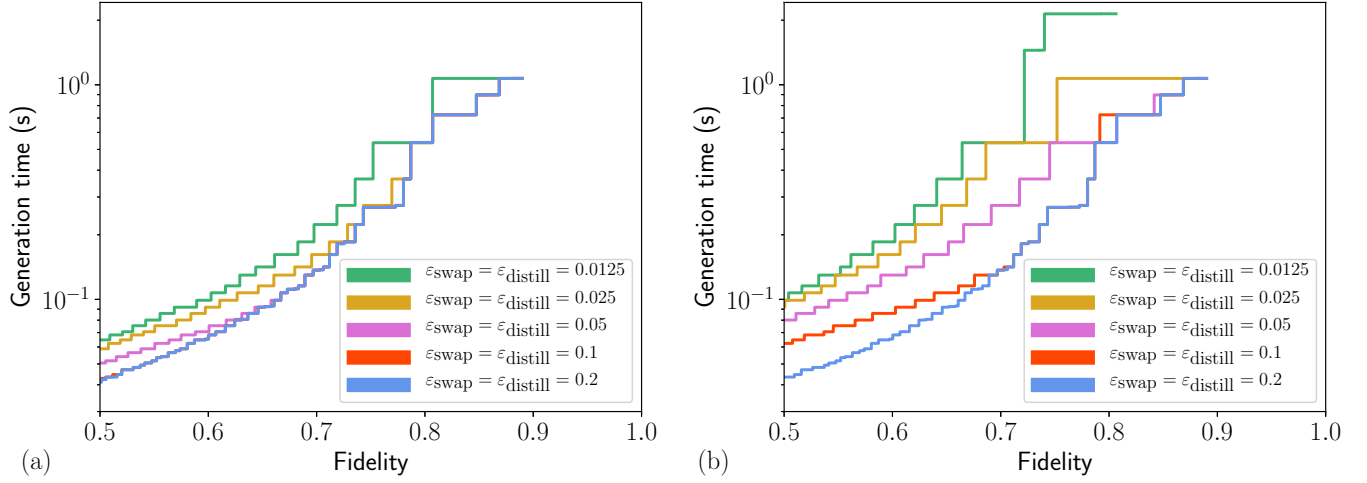


FIG. 24. Optimized schemes for a distance of 300 km using four intermediate nodes with the IP parameter set 4 (see Table II) for several different pairs of  $\epsilon_{\text{swap}}$  and  $\epsilon_{\text{distill}}$ . (a) Optimization results with banded swapping heuristic. The difference between  $\epsilon_{\text{swap}} = \epsilon_{\text{distill}} = 0.1$  and  $0.2$  is minimal, differing only slightly for very low fidelities. (b) Optimization results with naive swapping heuristic. Note that the curves converge in a significantly poorer fashion than in part level (a).

small fidelities there is a benefit in swapping between states with disparate fidelities.

### APPENDIX C: AVERAGE NOISE DUE TO STORAGE

Here we discuss the average noise induced when repeating a protocol with success probability  $p$  until success or until a maximum number of  $r$  attempts. Denote the quantum channel corresponding to storing for a single round by  $\Lambda$ . The average noise channel  $\mathbb{E}[\Lambda]$  corresponds to having the channel

$\underbrace{\Lambda \circ \Lambda \dots \circ \Lambda}_{r-j}$  with probability  $\frac{p(1-p)^{j-1}}{1-(1-p)^r}$  for  $1 \leq j \leq r$ . Note

that we can calculate the average channel instead of the average density matrix at the output, due to the linearity of quantum channels.

We consider two types of noise in this paper, depolarizing and dephasing. These types of noise occur naturally in quantum information processing systems, and have the following exponential behavior:

$$\mathcal{N}_d(\rho) = e^{-\frac{t}{\lambda_d}} \rho + (1 - e^{-\frac{t}{\lambda_d}}) \frac{\mathbb{I}}{2}, \quad (\text{depolarizing})$$

$$\mathcal{N}_Z(\rho) = \frac{1 + e^{-\frac{t}{\lambda_Z}}}{2} \rho + \frac{1 - e^{-\frac{t}{\lambda_Z}}}{2} \frac{\mathbb{I}}{2}, \quad (\text{dephasing}).$$

Thus, if we want to calculate the average amount of noise for depolarizing and dephasing, it suffices to calculate the average of  $e^{-c(k-j)}$  with probability distribution  $\frac{p(1-p)^{j-1}}{1-(1-p)^r}$ ,  $j = \{1, \dots, r\}$ , where  $c \equiv \frac{T_{\text{attempt}}}{T_{\text{depol/deph}}}$  quantifies the noise experienced in a single attempt for depolarizing and dephasing, respectively. We find thus that the average channels correspond to having the exponential terms in the above channels set to

$$\begin{aligned} \mathbb{E}[e^{-c(r-j)}] &= \sum_{j=1}^r \frac{p(1-p)^{j-1}}{1-(1-p)^r} e^{-c(r-j)} \\ &= \frac{pe^c((1-p)^r - e^{-cr})}{(1-(1-p)^r)(e^c(1-p) - 1)}. \end{aligned} \quad (\text{C1})$$

Finally, the decay in the success probability for retrieving a state from a memory for MP platforms is given by  $\mathbb{E}[e^{-c(r-j)}]$ , where  $c = \frac{T_{\text{attempt}}}{T_{\text{coh}}}$ .

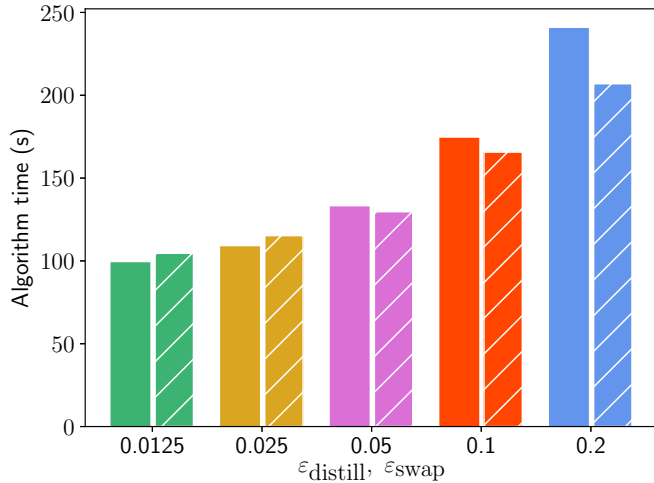


FIG. 25. Algorithm runtimes for several different values of  $\epsilon_{\text{swap}}$  and  $\epsilon_{\text{distill}}$ . The right, solid bars are for the optimization with the heuristic for swapping found in Eq. (6), while the left, hatched bars are for the optimization where we only swap between states that are  $\epsilon_{\text{swap}}$  close in fidelity. We observe that both heuristics lead to approximately the same runtime behavior, while the results with the banded swapping heuristic are closer to optimal.



#### APPENDIX D: MODELING OF ELEMENTARY PAIR GENERATION FOR MP PLATFORMS

Here we detail the calculations performed to derive the analytical form of the resultant state during elementary pair generation for MP platforms, and the success probability [see Eq. (D1)]. We first discuss the effects of the losses on the state emitted by the PDC sources. Second, the Bell state measurement and the resulting postmeasurement state are discussed. We will close with a brief discussion on the postselection of having zero photons. We model all losses in the setup as a pure-loss channel. Since we restrict ourselves to at most two-photon excitations in each mode, we truncate the Kraus operators from [88] to the  $\{|0\rangle, |1\rangle, |2\rangle\}$  subspace and find the explicit matrix form of the truncated Kraus operators. They are

$$A_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \sqrt{1-\gamma} & 0 \\ 0 & 0 & 1-\gamma \end{bmatrix},$$

$$A_1 = \begin{bmatrix} 0 & \sqrt{\gamma} & 0 \\ 0 & 0 & \sqrt{2(1-\gamma)\gamma} \\ 0 & 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 & \gamma \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

where  $\gamma = 1 - \eta$  is the loss parameter. Note that, even after truncation, these Kraus operators still form a channel since  $\sum_{i=0}^2 A_i^\dagger A_i = \mathbb{I}$ . We now let four such channels act on the state:

$$|\psi_{N_s}\rangle = \sqrt{p_0}|00, 00\rangle + \sqrt{\frac{p_1}{2}}(|10, 01\rangle + |01, 10\rangle) + \sqrt{\frac{p_2}{3}}(|20, 02\rangle - |11, 11\rangle + |02, 20\rangle),$$

where the early and late photonic modes in the direction towards the memory each evolve under a truncated pure-loss channel with parameter  $\gamma_1$ , and similarly for the two modes going towards the beamsplitter station with parameter  $\gamma_2$ . This results in a state  $\rho(N_s, \gamma_1, \gamma_2)_{a_0 a_1 b_0 b_1}$  between the memory and the photon just before the beamsplitter. The same situation holds for the other PDC source, such that the total state just before the beamsplitter is  $\rho(N_s, \gamma_1, \gamma_2)_{a_0 a_1 b_0 b_1} \otimes \rho(N_s, \gamma_1, \gamma_2)_{c_0 c_1 d_0 d_1}$ , where we have assumed the prepared states have equal mean photon number and experience equal losses. Instead of applying the unitary corresponding to the beamsplitter and then applying the positive operator-valued measures (POVMs) for the detectors, we can apply the inverse of the beamsplitter unitary on the POVMs corresponding to success. Since we assume photon number resolving detectors, we find our POVM elements corresponding to success to be

$$\mathbb{I}_{a_0 a_1} \otimes |\Psi^+\rangle\langle\Psi^+|_{b_0 c_0} \otimes |\Psi^+\rangle\langle\Psi^+|_{b_1 c_1} \otimes \mathbb{I}_{d_0 d_1},$$

$$\mathbb{I}_{a_0 a_1} \otimes |\Psi^-\rangle\langle\Psi^-|_{b_0 c_0} \otimes |\Psi^+\rangle\langle\Psi^+|_{b_1 c_1} \otimes \mathbb{I}_{d_0 d_1},$$

$$\mathbb{I}_{a_0 a_1} \otimes |\Psi^+\rangle\langle\Psi^+|_{b_0 c_0} \otimes |\Psi^-\rangle\langle\Psi^-|_{b_1 c_1} \otimes \mathbb{I}_{d_0 d_1},$$

$$\mathbb{I}_{a_0 a_1} \otimes |\Psi^-\rangle\langle\Psi^-|_{b_0 c_0} \otimes |\Psi^-\rangle\langle\Psi^-|_{b_1 c_1} \otimes \mathbb{I}_{d_0 d_1}.$$

We find that the postmeasurement states for each of these POVM elements are equivalent up to local unitaries, such that we only have to consider the first one. While one could call the whole process described so far elementary pair generation, the

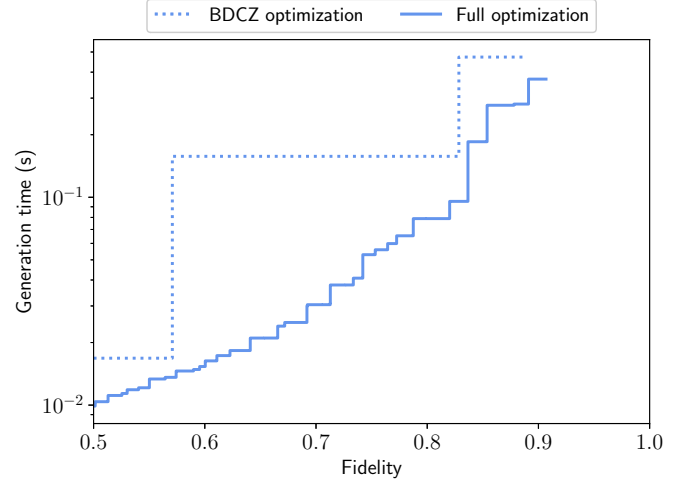


FIG. 26. Comparison between the optimization results of the full optimization with the optimization over BDCZ schemes. We consider a repeater chain over 200 km with three intermediate nodes. The parameters used are parameter set 4 for the intermediate nodes and parameter set 2 for Alice and Bob. BDCZ schemes are those schemes that only perform swapping and distillation between two schemes that have used the same sequences of protocols. Contrary to the comparison with BDCZ schemes in [20] we allow for an optimization over the different ways of generating elementary pairs, i.e., we vary the number of attempts  $r$  and the  $\theta$  parameter. We observe that the schemes found with the full optimization outperform BDCZ schemes, achieving a faster generation time by a factor of  $\approx 10$ , and extending the maximal achievable fidelity by a small margin.

state will have a fidelity equal to half or less for any  $N_s > 0$ . The reason for this is that there has not been any postselection on detecting a valid click pattern on the detectors when performing, say, Bell state measurements. For this reason, we apply the following POVM to postselect on having nonzero photons at each side of the memory:

$$\mathbb{I}_{a_0 a_1 b_0 b_1} - (|00\rangle\langle 00|_{a_0 a_1} \otimes \mathbb{I} + \mathbb{I} \otimes |00\rangle\langle 00|_{d_0 d_1} - |0000\rangle\langle 0000|_{a_0 a_1 d_0 d_1}).$$

The resultant state is too cumbersome to report here, but can be found in the accompanying MATHEMATICA [89] and PYTHON scripts. We find the success probability to be given by

$$p_{\text{succ}} = 4p_{\text{BSM}}p_{\text{nonzero photons}}$$

$$= 4\eta^2 \frac{3p_1^2 - 4(4\eta - 3)p_1 p_2 + 4p_2(1 + (3 - 8\eta + 4\eta^2))}{24} p_{\text{app}}^2 \frac{(p_1 + 4(\eta - 1)p_2(p_{\text{app}} - 2))(3p_1 + 4(\eta - 1)p_2(p_{\text{app}} - 2))}{4p_2 + (p_1 + (2 - 4\eta)p_2)(3p_1 + (6 - 4\eta)p_2)}. \quad (\text{D1})$$

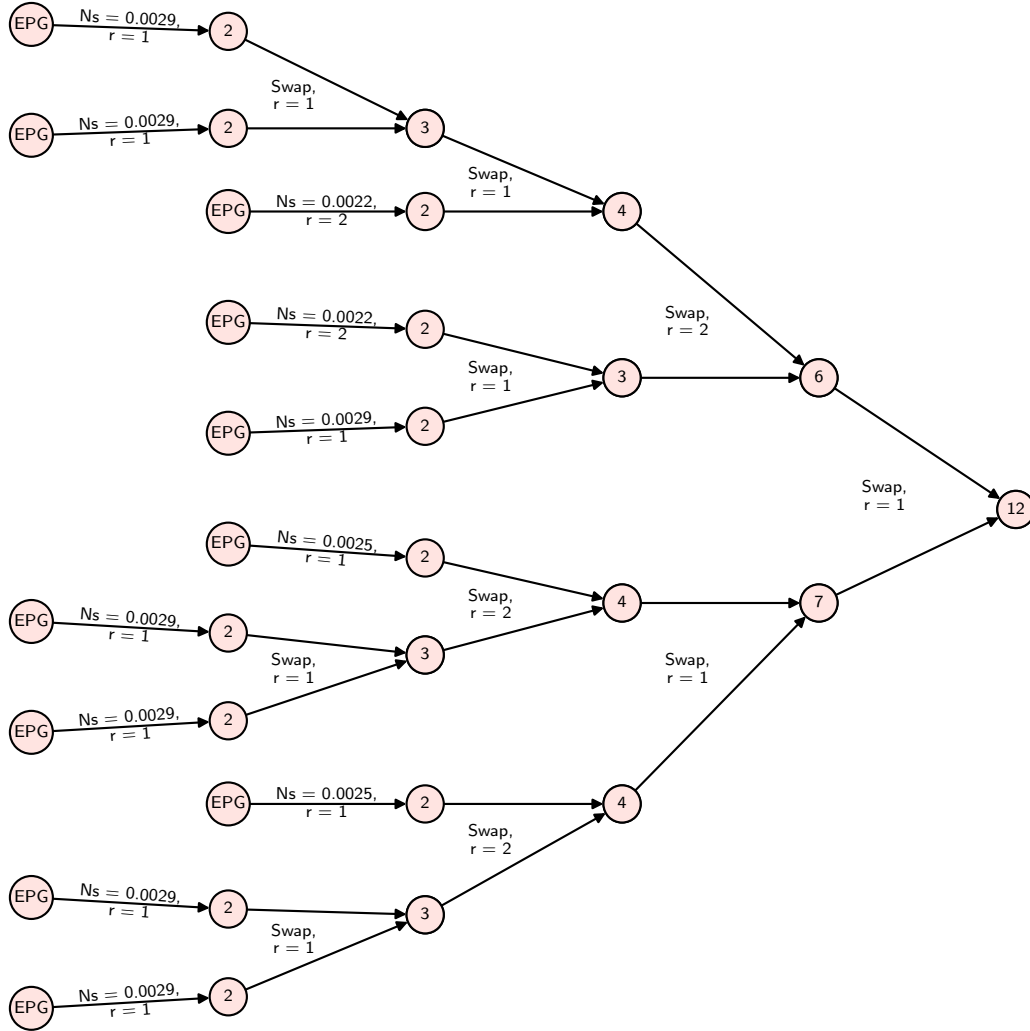


FIG. 27. Visual representation of the scheme with the lowest nontrivial achieved fidelity for a distance of 800 km with a combination of IP and MP platforms using parameter sets 4 (see Tables II and IV) and ten intermediate nodes or eleven hops. Elementary pair generation is indicated by EPG, and the mean photon number used is indicated by the  $N_s$ .

**APPENDIX E: THE INTERPLAY BETWEEN THE NUMBER OF MODES AND THE FIDELITY FOR MP PLATFORMS**

Here we investigate the interplay between the number of modes, the fidelity, and the losses in the fiber for MP platforms. We assume here that the only source of noise is from the PDC source, and there are no or negligible losses locally. We take the state derived in the previous section (but which is too cumbersome to report here), and set  $p_{\text{app}} = 1$ . The fidelity of the resultant state is then calculated to be

$$F = \frac{3}{4(\eta - 1)^2 N_s^4 + 24(\eta - 1)^2 N_s^3 + 4(9\eta^2 - 20\eta + 11)N_s^2 - 24(\eta - 1)N_s + 3}.$$

Solving for  $N_s$ , we find that

$$N_s = \frac{1}{2} \left( \sqrt{\frac{-9\eta F + 5F + 2\sqrt{F(F+3)}}{F - \eta F}} - 3 \right).$$

Let us now input the above relation into Eq. (D1) where we set  $p_1$  and  $p_2$  according to Eqs. (10). We find a success probability of

$$p = \frac{32\eta^2 \left( \sqrt{\frac{-9F\eta + 5F + 2\sqrt{F(F+3)}}{F - \eta F}} - 3 \right)^2}{F \left( \sqrt{\frac{-9F\eta + 5F + 2\sqrt{F(F+3)}}{F - \eta F}} - 1 \right)^6}. \tag{E1}$$

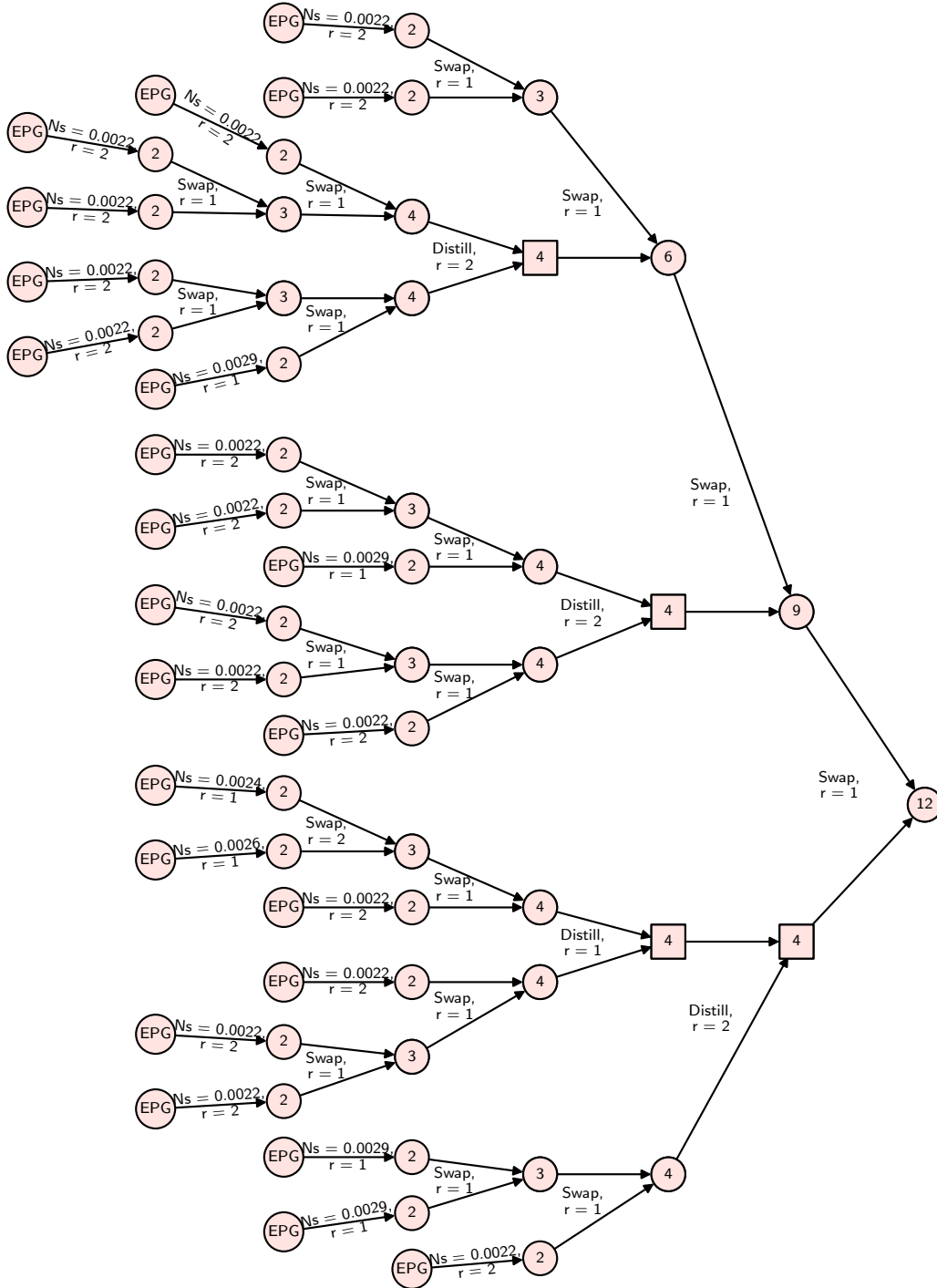


FIG. 28. Visual representation of the scheme that achieves a fidelity of  $F = 0.9605$  in time  $T = 17.7$  ms (indicated by the cross in Fig. 21), for a distance of 800 km with a combination of IP and MP platforms using parameter sets 4 (see Tables II and IV) and ten intermediate nodes or eleven hops. Elementary pair generation is indicated by EPG, and the mean photon number used is indicated by the  $N_s$ . The structure of the scheme is nonhierarchical, which can most clearly be seen in the final swap operation, which happens between two multihop links of lengths 4 and 9.

Since we need on the order of  $\frac{1}{p}$  modes, we find that we need on the order of

$$\frac{1}{p} = \frac{F \left( \sqrt{\frac{-9F\eta + 5F + 2\sqrt{F(F+3)}}{F - F\eta}} - 1 \right)^6}{32\eta^2 \left( \sqrt{\frac{-9F\eta + 5F + 2\sqrt{F(F+3)}}{F - F\eta}} - 3 \right)^2} = \frac{F \left( \sqrt{\frac{2\sqrt{F(F+3)}}{F} + 5 - 1} \right)^6}{32\eta^2 \left( \sqrt{\frac{2\sqrt{F(F+3)}}{F} + 5 - 3} \right)^2} + O(\eta^{-1})$$

modes to achieve a fidelity of  $F$  for  $\eta \approx 0$ . The  $\eta^2 = \exp(-\frac{L}{L_0})$  term in the denominator is given by the total losses of the fiber. The contribution due to the fidelity is then given by

$$\frac{F \left( \sqrt{\frac{2\sqrt{F(F+3)}}{F}} + 5 - 1 \right)^6}{32 \left( \sqrt{\frac{2\sqrt{F(F+3)}}{F}} + 5 - 3 \right)^2} = \frac{32}{(1-F)^2} + O((1-F)^{-1}).$$

We thus find that the number of minimum required modes scales as  $\frac{e^{\frac{L}{L_0}}}{(1-F)^2}$ , where  $L$  is the internode distance.

#### APPENDIX F: THE EFFECT OF EFFICIENCY DECOHERENCE FOR MP PLATFORMS

In this section we explore the effects the exponential decrease of the output efficiency has on the ability of performing schemes with probability greater than  $p_{\min}$ . While for information processing platforms it is always possible to achieve any success probability by performing as many attempts  $r$  as required, this is not the case for MP platforms due to the decrease in output efficiency over time. Here we derive conditions on the efficiency coherence times of the memories for a given  $p_{\min}$ , generation time  $T$ , and success probability  $p$  of the underlying schemes, such that  $p_{\min}$  can be achieved.

Since there are two memories used for state storage, the success probability of emitting both states again is modeled as given by

$$\begin{aligned} p_{\text{single success}} &= (1 - (1-p)^r) \mathbb{E}[e^{-(c_1+c_2)(r-j)}] \\ &= \frac{pe^{(c_1+c_2)}((1-p)^r - e^{-(c_1+c_2)r})}{e^{(c_1+c_2)}(1-p) - 1}. \end{aligned} \quad (\text{F1})$$

We are interested in when the above quantity cannot be larger than  $p_{\min}$ . To this end, we take the derivative of Eq. (F1) with respect to  $r$  and set it to zero to find the maximum value of success probability. Setting  $c = c_1 + c_2$ , we find

$$\begin{aligned} \frac{e^c p (ce^{-cr} + (1-p)^r \ln(1-p))}{e^c(1-p) - 1} &= 0, \\ \rightarrow r &= \frac{c - \ln\left(-\frac{e^c \ln(1-p)}{c}\right)}{c + \ln(1-p)}. \end{aligned} \quad (\text{F2})$$

However, since  $r$  needs to be an integer equal to or greater than 1, we choose the ceiling or floor of Eq. (F2), whichever maximizes the resultant  $p_{\text{succ}}$ . Furthermore, since we cannot perform distillation, our main concern is the drop in success probability after performing a Bell state measurement. This motivates us to set  $p = 1 - \frac{1}{2^{N+1}}$  [68,69,71–73]. Setting Eq. (F1) equal to  $p_{\min} = 0.9$ , we numerically find that  $N = 0$  gives  $c \approx 0.023$ ,  $N = 1$  gives  $c \approx 0.053$ ,  $N = 2$  gives  $c \approx 0.101$ , and  $N = 3$  gives  $c = \infty$ . Obviously, the assumption here is that the initial success probability is given by  $1 - \frac{1}{2^{N+1}}$ ,

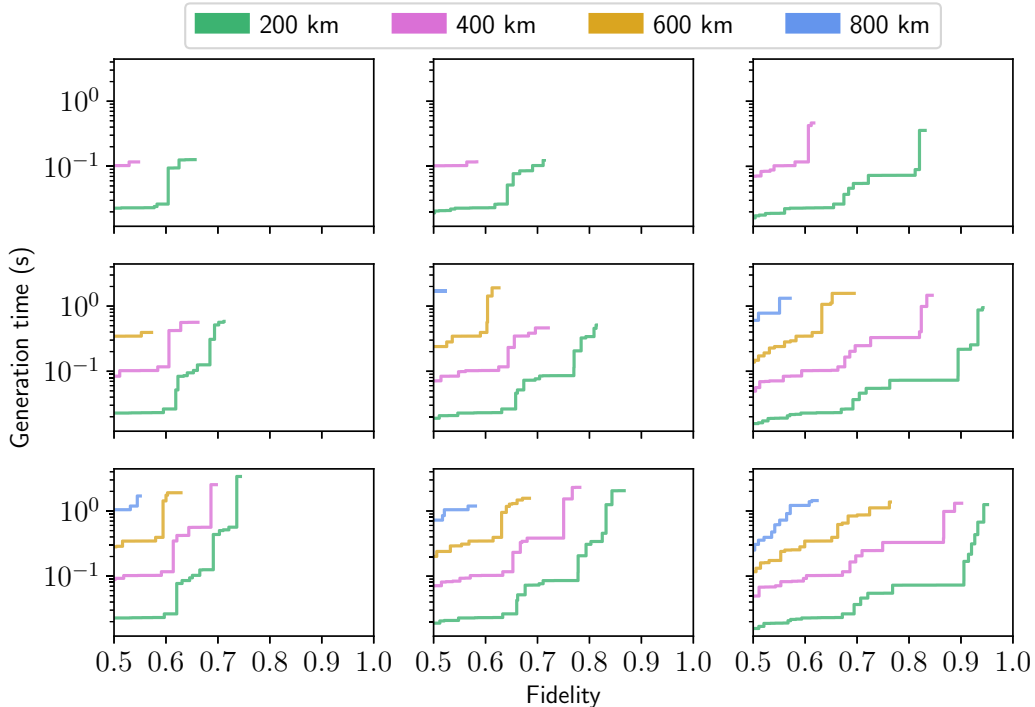


FIG. 29. Optimization results for total distances of 200, 400, 600, and 800 km, using ten intermediate nodes. We use IP parameter set 2 as a baseline, where we set the gate fidelities to be 0.99, 0.995, and 0.999 in the first, second, and third column, respectively. We set the coherence times  $T_{\text{deph}}$  and  $T_{\text{depol}}$  to 10, 50, and 100 s in the first, second, and third row, respectively.

which is not true due to other losses in the system. However, it is clear that increasing  $N$  can increase the total time significantly during which entanglement can be generated in a near-deterministic fashion. In particular, we find that the sum of the reciprocals of the efficiency coherence times of the memories should be *at least*  $\frac{1}{c}$  times the generation time of a scheme for MP platforms to successfully generate entanglement near-deterministically. This results in factors of approximately 43, 19, and 10 times the generation time for Bell state measurement success probabilities of  $\frac{1}{2}$ ,  $\frac{3}{4}$ , and  $\frac{7}{8}$ , respectively.

## APPENDIX G: ADDITIONAL OPTIMIZATION RESULTS

This section contains the additional figures mentioned in the main text.

First, we compare the optimization results of the full optimization with an optimization over BDCZ schemes only in

Fig. 26. BDCZ schemes are those schemes that for each connection and distillation step only combine two schemes that have used the same sequence of protocols, as in [13,14]. We consider an asymmetric repeater chain with three intermediate nodes over a distance of 200 km. We model the behavior of the intermediate nodes with parameter set 4 and Alice and Bob with parameter set 2 (see Table II). The full optimization yields schemes that can achieve faster generation rates (by approximately a factor of 10) than achievable with the BDCZ schemes.

Second, we consider two visualizations (Figs. 27 and 28) of the schemes found for a distance of 800 km with a combination of IP and MP platforms using parameter sets 4 (see Tables II and IV) and ten intermediate nodes. Finally, Fig. 29 contains the results found while performing a parameter exploration for total distances of 200, 400, 600, and 800 km, using ten intermediate nodes for information processing platforms.

- 
- [1] K. Chen and H.-K. Lo, Conference key agreement and quantum sharing of classical secrets with noisy GHZ states, in *Proceedings of the International Symposium on Information Theory*, 2005 (unpublished), pp. 1607–1611.
  - [2] J. Ribeiro, G. Murta, and S. Wehner, Fully device-independent conference key agreement, *Phys. Rev. A* **97**, 022307 (2018).
  - [3] R. Jozsa, D. S. Abrams, J. P. Dowling, and C. P. Williams, Quantum Clock Synchronization based on Shared Prior Entanglement, *Phys. Rev. Lett.* **85**, 2010 (2000).
  - [4] R. Ben-Av and I. Exman, Optimized multiparty quantum clock synchronization, *Phys. Rev. A* **84**, 014301 (2011).
  - [5] V. Giovannetti, S. Lloyd, and L. Maccone, Quantum-enhanced positioning and clock synchronization, *Nature (London)* **412**, 417 (2001).
  - [6] C. Crépeau, D. Gottesman, and A. Smith, Secure multi-party quantum computation, in *Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing*, 2002 (unpublished), pp. 643–652.
  - [7] W. K. Wootters and W. H. Zurek, A single quantum cannot be cloned, *Nature (London)* **299**, 802 (1982).
  - [8] D. Dieks, Communication by EPR devices, *Phys. Lett. A* **92**, 271 (1982).
  - [9] S. Pirandola, R. Laurenza, C. Ottaviani, and L. Banchi, Fundamental limits of repeaterless quantum communications, *Nat. Commun.* **8**, 15043 (2017).
  - [10] M. Takeoka, S. Guha, and M. M. Wilde, The squashed entanglement of a quantum channel, *IEEE Trans. Inf. Theory* **60**, 4987 (2014).
  - [11] S. Pirandola, R. García-Patrón, S. L. Braunstein, and S. Lloyd, Direct and Reverse Secret-Key Capacities of a Quantum Channel, *Phys. Rev. Lett.* **102**, 050503 (2009).
  - [12] S. Pirandola, U. L. Andersen, L. Banchi, M. Berta, D. Bunandar, R. Colbeck, D. Englund, T. Gehring, C. Lupo, C. Ottaviani *et al.*, Advances in quantum cryptography, *Adv. Opt. Photon.* **12**, 1012 (2020).
  - [13] H. Briegel, W. Dür, J. I. Cirac, and P. Zoller, Quantum Repeaters: The Role of Imperfect Local Operations in Quantum Communication, *Phys. Rev. Lett.* **81**, 5932 (1998).
  - [14] W. Dür, H.-J. Briegel, J. I. Cirac, and P. Zoller, Quantum repeaters based on entanglement purification, *Phys. Rev. A* **59**, 169 (1999).
  - [15] L. Childress, J. Taylor, A. S. Sørensen, and M. D. Lukin, Fault-tolerant quantum repeaters with minimal physical resources and implementations based on single-photon emitters, *Phys. Rev. A* **72**, 052330 (2005).
  - [16] L. Childress, J. Taylor, A. S. Sørensen, and M. Lukin, Fault-Tolerant Quantum Communication based on Solid-State Photon Emitters, *Phys. Rev. Lett.* **96**, 070504 (2006).
  - [17] C. H. Bennett, H. J. Bernstein, S. Popescu, and B. Schumacher, Concentrating partial entanglement by local operations, *Phys. Rev. A* **53**, 2046 (1996).
  - [18] C. H. Bennett, G. Brassard, S. Popescu, B. Schumacher, J. A. Smolin, and W. K. Wootters, Purification of Noisy Entanglement and Faithful Teleportation via Noisy Channels, *Phys. Rev. Lett.* **76**, 722 (1996).
  - [19] N. Kalb, A. A. Reiserer, P. C. Humphreys, J. J. W. Bakermans, S. J. Kamerling, N. H. Nickerson, S. C. Benjamin, D. J. Twitchen, M. Markham, and R. Hanson, Entanglement distillation between solid-state quantum network nodes, *Science* **356**, 928 (2017).
  - [20] L. Jiang, J. M. Taylor, N. Khaneja, and M. D. Lukin, Optimal approach to quantum communication using dynamic programming, *Proc. Natl. Acad. Sci. USA* **104**, 17291 (2007).
  - [21] S. Brand, T. Coopmans, and D. Elkouss, Efficient computation of the waiting time and fidelity, *IEEE J. Selected Areas Comm.* **38**, 619 (2020).
  - [22] E. Shechukin, F. Schmidt, and P. van Loock, On the waiting time in quantum repeaters with probabilistic entanglement swapping, *Phys. Rev. A* **100**, 032322 (2019).
  - [23] S. Santra, L. Jiang, and V. S. Malinovsky, Quantum repeater architecture with hierarchically optimized memory buffer times, *Quantum Sci. Technol.* **4**, 025010 (2019).
  - [24] S. E. Vinay and P. Kok, Statistical analysis of quantum-entangled-network generation, *Phys. Rev. A* **99**, 042313 (2019).

- [25] <https://github.com/KDGoodenough/RepeaterchainOptimisation>.
- [26] C. Monroe and J. Kim, Scaling the ion trap quantum processor, *Science* **339**, 1164 (2013).
- [27] I. V. Inlek, C. Crocker, M. Lichtman, K. Sosnova, and C. Monroe, Multispecies Trapped-ion Node for Quantum Networking, *Phys. Rev. Lett.* **118**, 250502 (2017).
- [28] M. Bock, P. Eich, S. Kucera, M. Kreis, A. Lenhard, C. Becher, and J. Eschner, High-fidelity entanglement between a trapped ion and a telecom photon via quantum frequency conversion, *Nat. Commun.* **9**, 1998 (2018).
- [29] L. Childress and R. Hanson, Diamond NV centers for quantum computing and quantum networks, *MRS Bull.* **38**, 134 (2013).
- [30] K. Tsurumoto, R. Kuroiwa, H. Kano, Y. Sekiguchi, and H. Kosaka, Quantum teleportation-based state transfer of photon polarization into a carbon spin in diamond, *Commun. Phys.* **2**, 74 (2019).
- [31] S. Welte, B. Hacker, S. Daiss, S. Ritter, and G. Rempe, Photon-Mediated Quantum Gate between Two Neutral Atoms in an Optical Cavity, *Phys. Rev. X* **8**, 011018 (2018).
- [32] J. D. Siverns, J. Hannegan, and Q. Quraishi, Neutral-Atom Wavelength-Compatible 780 nm Single Photons from a Trapped Ion via Quantum Frequency Conversion, *Phys. Rev. Applied* **11**, 014044 (2019).
- [33] M. Razavi and J. H. Shapiro, Long-distance quantum communication with neutral atoms, *Phys. Rev. A* **73**, 042303 (2006).
- [34] R. Hanson, L. P. Kouwenhoven, J. R. Petta, S. Tarucha, and L. M. K. Vandersypen, Spins in few-electron quantum dots, *Rev. Mod. Phys.* **79**, 1217 (2007).
- [35] Robert, A. Žak, B. Röthlisberger, S. Chesi, and D. Loss, Quantum computing with electron spins in quantum dots, *La Rivista del Nuovo Cimento* **33**, 7 (2010).
- [36] C.-W. Chou, H. De Riedmatten, D. Felinto, S. V. Polyakov, S. J. Van Enk, and H. J. Kimble, Measurement-induced entanglement for excitation stored in remote atomic ensembles, *Nature (London)* **438**, 828 (2005).
- [37] D. Matsukevich, T. Chaneliere, M. Bhattacharya, S.-Y. Lan, S. Jenkins, B. Kennedy, and A. Kuzmich, Entanglement of a Photon and a Collective Atomic Excitation, *Phys. Rev. Lett.* **95**, 040405 (2005).
- [38] H. Krovi, S. Guha, Z. Dutton, J. A. Slater, C. Simon, and W. Tittel, Practical quantum repeaters with parametric down-conversion sources, *Appl. Phys. B* **122**, 52 (2016).
- [39] D. Deutsch, A. Ekert, R. Jozsa, C. Macchiavello, S. Popescu, and A. Sanpera, Quantum Privacy Amplification and the Security of Quantum Cryptography Over Noisy Channels, *Phys. Rev. Lett.* **77**, 2818 (1996).
- [40] J. Dehaene, M. Van den Nest, B. De Moor, and F. Verstraete, Local permutations of products of Bell states and entanglement distillation, *Phys. Rev. A* **67**, 022310 (2003).
- [41] R. Horodecki, P. Horodecki, M. Horodecki, and K. Horodecki, Quantum entanglement, *Rev. Mod. Phys.* **81**, 865 (2009).
- [42] As discussed later, we actually do consider distillation protocols taking three or more states to a single one, but these are composed of several distillation protocols taking two states to a single one.
- [43] Or, in the case of multiplexed elementary pair generation platforms, the probability of having a single success for elementary pair generation can be increased by having more modes.
- [44] Since the success of a scheme follows a geometric distribution, the average generation time can be computed from the success probability and the generation time of one attempt.
- [45] P. C. Humphreys, N. Kalb, J. P. J. Morits, R. N. Schouten, R. F. L. Vermeulen, D. J. Twitchen, M. Markham, and R. Hanson, Deterministic delivery of remote entanglement on a quantum network, *Nature (London)* **558**, 268 (2018).
- [46] 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* **5**, 25 (2019).
- [47] E. Schoute, L. Mančinska, T. Islam, I. Kerenidis, and S. Wehner, Shortcuts to quantum network routing, [arXiv:1610.05238](https://arxiv.org/abs/1610.05238).
- [48] W. Dür and H. J. Briegel, Entanglement purification and quantum error correction, *Rep. Prog. Phys.* **70**, 1381 (2007).
- [49] R. Van Meter, T. D. Ladd, W. J. Munro, and K. Nemoto, System design for a long-line quantum repeater, *IEEE/ACM Transactions on Networking* **17**, 1002 (2009).
- [50] B. Hensen, H. Bernien, A. E. Dréau, A. Reiserer, N. Kalb, M. S. Blok, J. Ruitenberg, R. F. L. Vermeulen, R. N. Schouten, C. Abellán *et al.*, Loophole-free Bell inequality violation using electron spins separated by 1.3 kilometres, *Nature (London)* **526**, 682 (2015).
- [51] B. Hensen, N. Kalb, M. S. Blok, A. E. Dréau, A. Reiserer, R. F. L. Vermeulen, R. N. Schouten, M. Markham, D. J. Twitchen, K. Goodenough *et al.*, Loophole-free Bell test using electron spins in diamond: Second experiment and additional analysis, *Sci. Rep.* **6**, 30289 (2016).
- [52] M. H. Abobeih, J. Cramer, M. A. Bakker, N. Kalb, M. Markham, D. J. Twitchen, and T. H. Taminiou, One-second coherence for a single electron spin coupled to a multi-qubit nuclear-spin environment, *Nat. Commun.* **9**, 2552 (2018).
- [53] J. Cramer, N. Kalb, M. Adriaan Rol, B. Hensen, M. S. Blok, M. Markham, D. J. Twitchen, R. Hanson, and T. H. Taminiou, Repeated quantum error correction on a continuously encoded qubit by real-time feedback, *Nat. Commun.* **7**, 11526 (2016).
- [54] C. T. Nguyen, D. D. Sukachev, M. K. Bhaskar, B. Machielse, D. S. Levonian, E. N. Knall, P. Stroganov, R. Riedinger, H. Park, M. Loncar and M. D. Lukin, Quantum Network Nodes based on Diamond Qubits with an Efficient Nanophotonic Interface, *Phys. Rev. Lett.* **123**, 183602 (2019).
- [55] C. T. Nguyen, D. D. Sukachev, M. K. Bhaskar, B. Machielse, D. S. Levonian, E. N. Knall, P. Stroganov, C. Chia, M. J. Burek, R. Riedinger, H. Park, M. Loncar and M. D. Lukin, An integrated nanophotonic quantum register based on silicon-vacancy spins in diamond, *Phys. Rev. B* **100**, 165428 (2019).
- [56] G. Burkard, H.-A. Engel, and D. Loss, Spintronics and quantum dots for quantum computing and quantum communication, *Fortschr. Phys.: Progress of Physics* **48**, 965 (2000).
- [57] Y. Chen, M. Zopf, R. Keil, F. Ding, and O. G. Schmidt, Highly-efficient extraction of entangled photons from quantum dots using a broadband optical antenna, *Nat. Commun.* **9**, 2994 (2018).
- [58] D. Huber, M. Reindl, J. Aberl, A. Rastelli, and R. Trotta, Semiconductor quantum dots as an ideal source of polarization-entangled photon pairs on-demand: A review, *J. Opt.* **20**, 073002 (2018).
- [59] C. Cabrillo, J. I. Cirac, P. Garcia-Fernandez, and P. Zoller, Creation of entangled states of distant atoms by interference, *Phys. Rev. A* **59**, 1025 (1999).

- [60] M. Lucamarini, Z. L. Yuan, J. F. Dynes, and A. J. Shields, Overcoming the rate–distance limit of quantum key distribution without quantum repeaters, *Nature (London)* **557**, 400 (2018).
- [61] F. Rozpedek, R. Yehia, K. Goodenough, M. Ruf, P. C. Humphreys, R. Hanson, S. Wehner, and D. Elkouss, Near-term quantum repeater experiments with NV centers: overcoming the limitations of direct transmission, *Phys. Rev. A* **99**, 052330 (2018).
- [62] S. D. Barrett and P. Kok, Efficient high-fidelity quantum computation using matter qubits and linear optics, *Phys. Rev. A* **71**, 060310(R) (2005).
- [63] E. T. Campbell and S. C. Benjamin, Measurement-Based Entanglement under Conditions of Extreme Photon Loss, *Phys. Rev. Lett.* **101**, 130502 (2008).
- [64] F. Rozpedek, T. Schiet, L. P. Thinh, D. Elkouss, A. C. Doherty, and S. Wehner, Optimizing practical entanglement distillation, *Phys. Rev. A* **97**, 062333 (2018).
- [65] F. Rozpedek, 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 Sci. Tech.* **3**, 034002 (2018).
- [66] N. Sangouard, C. Simon, H. De Riedmatten, and N. Gisin, Quantum repeaters based on atomic ensembles and linear optics, *Rev. Mod. Phys.* **83**, 33 (2011).
- [67] P. Kok and S. L. Braunstein, Postselected versus non-postselected quantum teleportation using parametric down-conversion, *Phys. Rev. A* **61**, 042304 (2000).
- [68] N. Lütkenhaus, J. Calsamiglia, and K.-A. Suominen, Bell measurements for teleportation, *Phys. Rev. A* **59**, 3295 (1999).
- [69] L. Vaidman and N. Yoran, Methods for reliable teleportation, *Phys. Rev. A* **59**, 116 (1999).
- [70] A. Olivo and F. Grosshans, Investigating the optimality of ancilla-assisted linear optical Bell measurements, *Phys. Rev. A* **98**, 042323 (2018).
- [71] F. Ewert and P. van Loock, 3/4-Efficient Bell Measurement with Passive Linear Optics and Unentangled Ancillae, *Phys. Rev. Lett.* **113**, 140403 (2014).
- [72] W. P. Grice, Arbitrarily complete Bell-state measurement using only linear optical elements, *Phys. Rev. A* **84**, 042331 (2011).
- [73] S.-W. Lee, K. Park, T. C. Ralph, and H. Jeong, Nearly Deterministic Bell Measurement for Multiphoton Qubits and its Application to Quantum Information Processing, *Phys. Rev. Lett.* **114**, 113603 (2015).
- [74] S.-W. Lee, K. Park, T. C. Ralph, and H. Jeong, Nearly deterministic Bell measurement with multiphoton entanglement for efficient quantum-information processing, *Phys. Rev. A* **92**, 052324 (2015).
- [75] S.-W. Lee and H. Jeong, Near-deterministic quantum teleportation and resource-efficient quantum computation using linear optics and hybrid qubits, *Phys. Rev. A* **87**, 022326 (2013).
- [76] H. A. Zaidi and P. van Loock, Beating the One-Half Limit of Ancilla-Free Linear Optics Bell Measurements, *Phys. Rev. Lett.* **110**, 260501 (2013).
- [77] A. Holzäpfel, J. Etesse, K. T. Kaczmarek, A. Tiranov, N. Gisin, and M. Afzelius, Optical storage on the timescale of a second in a solid-state atomic frequency comb memory using dynamical decoupling, *New J. Phys.* **22**, 063009 (2020).
- [78] N. Timoney, I. Usmani, P. Jobez, M. Afzelius, and N. Gisin, Single-photon-level optical storage in a solid-state spin-wave memory, *Phys. Rev. A* **88**, 022324 (2013).
- [79] E. Saglamyurek, N. Sinclair, J. Jin, J. A. Slater, D. Oblak, F. Bussières, M. George, R. Ricken, W. Sohler, and W. Tittel, Broadband waveguide quantum memory for entangled photons, *Nature (London)* **469**, 512 (2011).
- [80] M. Afzelius, C. Simon, H. De Riedmatten, and N. Gisin, Multimode quantum memory based on atomic frequency combs, *Phys. Rev. A* **79**, 052329 (2009).
- [81] N. Kalb, A. Reiserer, S. Ritter, and G. Rempe, Heralded Storage of a Photonic Quantum bit in a Single Atom, *Phys. Rev. Lett.* **114**, 220501 (2015).
- [82] K. Nemoto, M. Trupke, S. J. Devitt, B. Scharfenberger, K. Buczak, J. Schmiedmayer, and W. J. Munro, Photonic quantum networks formed from NV- centers, *Sci. Rep.* **6**, 26284 (2016).
- [83] R. Paschotta, Article on “fibers,” in *Encyclopedia of Laser Physics and Technology* (Wiley-VCH, Weinheim, 2008).
- [84] M. Blok, N. Kalb, A. Reiserer, T. Taminiiau, and R. Hanson, Towards quantum networks of single spins: analysis of a quantum memory with an optical interface in diamond, *Faraday Discuss.* **184**, 173 (2015).
- [85] A. Reiserer, N. Kalb, M. S. Blok, K. J. M. van Bemmelen, T. H. Taminiiau, R. Hanson, D. J. Twitchen, and M. Markham, Robust Quantum-Network Memory using Decoherence-Protected Subspaces of Nuclear Spins, *Phys. Rev. X* **6**, 021040 (2016).
- [86] S. Watanabe, R. Matsumoto, T. Uyematsu, and Y. Kawano, Key rate of quantum key distribution with hashed two-way classical communication, *Phys. Rev. A* **76**, 032312 (2007).
- [87] Obviously, when  $\eta \rightarrow 0$  the probability of getting a successful click pattern is zero. However, here we are only interested in the worst-case scenario or upper bound, which corresponds to detecting a successful click pattern as  $\eta \rightarrow 0$ .
- [88] J. S. Ivan, K. K. Sabapathy, and R. Simon, Operator-sum representation for bosonic Gaussian channels, *Phys. Rev. A* **84**, 042311 (2011).
- [89] See Supplemental Materials at <http://link.aps.org/supplemental/10.1103/PhysRevA.103.032610> for the Mathematica notebook.