

ARTICLE OPEN



Designing quantum networks using preexisting infrastructure

Julian Rabbie^{1,2}, Kaushik Chakraborty^{1,2,3}, Guus Avis^{1,2,3} and Stephanie Wehner^{1,2}✉

We consider the problem of deploying a quantum network on an existing fiber infrastructure, where quantum repeaters and end nodes can only be housed at specific locations. We propose a method based on integer linear programming (ILP) to place the minimal number of repeaters on such an existing network topology, such that requirements on end-to-end entanglement-generation rate and fidelity between any pair of end-nodes are satisfied. While ILPs are generally difficult to solve, we show that our method performs well in practice for networks of up to 100 nodes. We illustrate the behavior of our method both on randomly-generated network topologies, as well as on a real-world fiber topology deployed in the Netherlands.

npj Quantum Information (2022)8:5; <https://doi.org/10.1038/s41534-021-00501-3>

INTRODUCTION

The quantum internet will provide an infrastructure for quantum communication between any two devices in the world^{1–5}. This can be used to perform tasks which are provably impossible with the classical internet. Many of these are cryptographic in nature⁶ and allow unconditional security, such as quantum key distribution^{7,8}, secure multi-party cryptography⁹ and blind quantum computation¹⁰. Other applications of the quantum internet include fast byzantine agreement¹¹ and clock synchronization¹².

A major challenge in the construction of terrestrial quantum networks is to overcome exponential loss in optical fibers. In order to enable quantum communication over large distances, quantum repeaters are required. These can form a quantum-repeater chain in which consecutive nodes are connected by elementary links. Quantum repeaters are a very active research area and major advances have been achieved recently^{13–18}. However, the technology is not yet at the stage of practical deployment, and we anticipate that the first practical quantum repeaters will be costly. It seems likely that before a global quantum internet is effected, smaller quantum networks connecting a limited number of end nodes are deployed. A cost-efficient way of deploying such networks is using existing classical infrastructure by converting already-deployed optical fiber and installing quantum repeaters at strategic locations.

We model a classical fiber network which forms the basis of a quantum network as an undirected, weighted graph $G = (\mathcal{N}, \mathcal{F}, \mathcal{L})$. The nodes \mathcal{N} are partitioned into a set of end nodes $\mathcal{C} \subset \mathcal{N}$ and a set of potential repeater locations $\mathcal{R} = \mathcal{N} \setminus \mathcal{C}$. The goal of the quantum network is to enable quantum communication between end nodes. Potential repeater locations are any location in the network where a quantum repeater could be placed. Such a location could, for example, be a hub in the classical network with the facilities required to run a quantum repeater. The edges of the graph are the fibers of the network, \mathcal{F} , where $\mathcal{L}(f)$ is the length of fiber $f \in \mathcal{F}$. In case a quantum repeater is installed at a potential repeater location, the potential repeater location becomes a quantum-repeater node. When deploying a quantum network based on a classical fiber network, it is essential to determine which potential repeater locations should be turned into quantum-repeater nodes.

In order to have an operational quantum network, nodes must be connected by elementary links. For many quantum-repeater schemes (such as those using heralded entanglement generation¹⁹), elementary links consist of fibers with active elements measuring qubits. Therefore, when deploying a quantum network based on a classical fiber network, it must also be determined which fibers to convert into elementary links. Here, we consider that elementary links can be constructed from any number of consecutively-adjacent fibers in the graph G (passing through potential repeater locations). Both fibers and potential repeater locations can be part of multiple elementary links, which is motivated by the fact that fibers are typically constructed in bundles (meaning that each elementary link could, in fact, use the same fiber bundle but a different fiber). Additionally, multiplexing over different wavelengths could be used to enable the use of a single fiber in multiple elementary links. For an example of how a (very) small classical fiber network can be used to create a quantum network, see Fig. 1.

Here, we introduce the problem of determining how to construct a quantum network using a pre-existing classical fiber network as the repeater-allocation problem. We define it as follows:

Given a classical fiber network corresponding to the undirected, weighted graph $G = (\mathcal{N}, \mathcal{F}, \mathcal{L})$ with end nodes $\mathcal{C} \subset \mathcal{N}$. Which of the potential repeater locations $\mathcal{R} = \mathcal{N} \setminus \mathcal{C}$ should be turned into quantum-repeater nodes, and which fibers should be converted into elementary links, such that a quantum network is obtained which satisfies a set of network requirements, while the associated costs are minimized?

In this paper we present a method which solves the repeater-allocation problem. Here, we only consider the costs associated to installing quantum repeaters, as we expect that the first practical quantum repeaters will come at a high cost. Furthermore, the set of network requirements that we consider are the following:

- **Rate and fidelity.** The quantum network must be able to distribute bipartite entangled quantum states between any pair of end nodes at some minimum rate, which we denote R_{\min} . Furthermore, the states must have some minimum fidelity to a maximally entangled state, which we denote F_{\min} . The network

¹QuTech, Delft University of Technology, Lorentzweg 1, 2628 CJ Delft, The Netherlands. ²Kavli Institute of Nanoscience, Delft University of Technology, Lorentzweg 1, 2628 CJ Delft, The Netherlands. ³These authors contributed equally: Kaushik Chakraborty, Guus Avis. ✉email: s.d.c.wehner@tudelft.nl

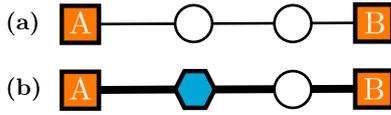


Fig. 1 Example of how a quantum network can be constructed. **a** Graph representing a simple fiber network. Nodes A and B are end nodes, while the other two nodes are potential repeater locations. **b** Quantum network that is constructed using the pre-existing fiber network. The first node from A is used as a quantum-repeater node (blue hexagon) and there are two elementary links. One elementary link is made from A to the quantum-repeater node, while the other starts at the quantum-repeater node and ends at B.

must be able to do this for every pair of end nodes simultaneously.

In a quantum-repeater chain with fixed hardware, the rate of entanglement distribution is limited by loss and noise in elementary links and in quantum repeaters. Therefore, it is generally possible to lower bound the rate by upper bounding the number of quantum repeaters (and thereby the number of elementary links), and the length of each elementary link (assuming the photon loss probability per unit length is constant). Similarly, fidelity is limited by noisy operations in quantum repeaters, while it can also be a decreasing function of the elementary link length (this can be, for example, due to dark counts in detectors). Therefore, fidelity too can be lower bounded by upper bounding the number of quantum repeaters and the elementary link length.

We use these bounds to assess whether the rate and fidelity between a pair of end nodes is sufficient. For any R_{\min} and F_{\min} , we can find N_{\max} and L_{\max} such that a repeater chain of N_{\max} repeaters and elementary links of length L_{\max} can deliver entangled states at rate R_{\min} with fidelity F_{\min} . Then, we consider two end nodes capable of receiving entangled states with at least rate R_{\min} and at least fidelity F_{\min} if there is a free path between them which contains at most N_{\max} repeaters and of which each elementary link is at most L_{\max} long.

How exactly N_{\max} and L_{\max} can be determined from R_{\min} and F_{\min} is specific to the quantum-repeater architecture and depends on various performance parameters. We give a toy-model calculation in Section “Toy-model calculation of N_{\max} and L_{\max} from R_{\min} and F_{\min} ” as an example. Note that when considering a quantum-repeater architecture which is not based on entanglement distribution, the method presented in this paper is still applicable if a performance metric like rate and fidelity can be determined which can be lower bounded by upper bounding the number of repeaters and the elementary link lengths of a repeater chain.

- **Robustness.** When a part of a quantum network breaks down, all other requirements should still be met. We quantify this using the minimum number of quantum-repeater nodes or elementary links (it can be any combination) that need to break down before one of the other requirements can no longer be met. Here, we use the symbol K to refer to this number.
- **Repeater capacity.** Quantum-repeater nodes should never be required to operate above their capacity in order to meet all other network requirements. We define the capacity of a quantum-repeater node as the maximum number of quantum-communication sessions it can facilitate simultaneously. In an entanglement-based network, this could be directly related to the number of entangled states that can be stored in memory or the number of Bell-state measurements that can be performed simultaneously. Here, we use the symbol D to refer to the capacity of the quantum-repeater nodes.

RESULTS

In this section we present a method, detailed below, which aids in the design of a quantum network using existing classical infrastructure. Specifically, given a fiber network, our method makes it possible to choose at which locations quantum repeaters should be installed. This is done such that entangled states can be distributed between all pairs of end nodes simultaneously with a minimum rate and fidelity. Furthermore, our method guarantees that the resulting quantum network is robust against failure of quantum repeaters and elementary links, and can take finite capacity of quantum repeaters into account. At the same time, our method minimizes the total number of quantum repeaters that need to be installed. We dub the problem that our method solves the repeater-allocation problem.

Method 1: Solving the Repeater-Allocation Problem Input

- Fiber network graph $G = (\mathcal{N}, \mathcal{F}, \mathcal{L})$.
- Set of end nodes $\mathcal{C} \subset \mathcal{N}$.
- Minimum rate R_{\min} and fidelity F_{\min} required by end nodes. Instead of a minimum rate and fidelity, one can also use the minimum value(s) for other performance metric(s), as long as these can be lower bounded by upper bounding the number of repeaters and elementary link lengths of a quantum-repeater chain.
- Required robustness parameter K (number of quantum-repeater nodes and elementary links that must be incapacitated before network operation is compromised).
- Capacity parameter D (number of quantum-communication sessions that one quantum repeater can facilitate simultaneously).

Method

1. Determine values for the parameters L_{\max} and N_{\max} such that a quantum-repeater chain consisting of N_{\max} repeaters and elementary links of length L_{\max} is able to deliver entangled states at rate R_{\min} with fidelity F_{\min} to a maximally entangled state.
2. Construct the set of potential repeater locations

$$\mathcal{R} = \mathcal{N} \setminus \mathcal{C}. \quad (1)$$
3. Construct the set

$$\mathcal{Q} = \{(s, t) | (s, t) \in_R \{(i, j), (j, i)\}, i, j \in \mathcal{C}, i \neq j\}, \quad (2)$$
 where \in_R implies picked uniformly at random.
4. For every $(s, t) \in \mathcal{Q}$, construct the set

$$\mathcal{E}_{(s,t)} = \{(n_1, n_2) | n_1 \in \mathcal{R} \cup \{s\}, n_2 \in \mathcal{R} \cup \{t\}, n_1 \neq n_2\}, \quad (3)$$
 and then construct the set

$$\mathcal{E} = \bigcup_{q \in \mathcal{Q}} \mathcal{E}_q. \quad (4)$$
5. For every $(u, v) \in \mathcal{E}$, determine the shortest path from u to v in the fiber-network graph G . Store the length of the path as $L((u, v))$ and the fibers making it up as $F((u, v))$.
6. Solve the link-based formulation, provided in Section “Link-based formulation”, using an integer linear programming solver. Store the values of the variables $x_{uv}^{q,k}$ and y_u .
7. Apply the path extraction algorithm, i.e. Algorithm 1, to obtain the set \mathcal{P}^* . For every $(u, v) \in \mathcal{E}$, set $x_{uv}^{q,k} = 0$ if there is no $p \in \mathcal{P}^*$ such that $(u, v) \in p$.

Solution

- Every potential repeater location $u \in \mathcal{R}$ for which $y_u = 1$ should be used as a quantum-repeater node.

- For every $(u, v) \in \mathcal{E}$ for which $x_{uv}^{q,k} = 1$ for some value of q and k , an elementary link should be constructed using the fibers $F((u, v))$.

Key to our method is integer linear programming (ILP), which can be used to obtain the optimal repeater placement with an optimization solver such as Clp²⁰, Gurobi²¹ or CPLEX²². Our method has been tested both using a real fiber network and a large number of randomized graphs, on which we report in Discussion. The real network contains four end nodes and 50 potential repeater locations, and a solution was found in 74 s using a computer running a quad-core Intel Xeon W-2123 processor at 3.60 GHz and 16 GB of RAM, demonstrating that the method is feasible for realistically-sized networks.

Here, we put forward two different ILP formulations. The first, which we call the path-based formulation (see Section “Path-Based Formulation”), is based on enumerating and then choosing paths between end nodes of the quantum network. It is relatively easy to show and understand that this formulation indeed solves the repeater-allocation problem (see Section “Explanation of path-based formulation”). However, it is not efficient, as the number of variables and constraints in the formulation grows exponentially with the size of the network. The second formulation is the link-based formulation (see “Link-based formulation”). This formulation is much more efficient than the path-based formulation, as it only grows polynomially with the size of the network. Therefore, our method uses the link-based formulation. It is, however, harder to see that the link-based formulation can be used to solve the repeater-allocation problem. Yet, the link-based formulation is equivalent to the path-based formulation, as we show in Section “Proof of equivalence”.

The structure of the paper is as follows. In the remainder of this section, we introduce both the intuitive path-based formulation and the efficient link-based formulation. Next, in Discussion, we first give an example of the use of our method on a real fiber network in the Netherlands. We also study the behavior and performance of the method on a large number of randomly-generated network graphs. Furthermore, we present ways in which our method can be extended, and we discuss its limitations. Finally, in Methods, we argue that the path-based formulation can indeed be used to solve the repeater-allocation problem, we give an example of a rate-fidelity analysis, we sketch a proof of the equivalence of the path-based formulation and the link-based formulation, we explain how we generate random network graphs and we present the scaling of the two ILP formulations.

Path-based formulation

The main idea behind the path-based formulation is to enumerate and then choose paths for every $(s, t) \in \mathcal{Q}$, where \mathcal{Q} is the set of all ordered pairs of end nodes as defined in Eq. (2). A path between s and t is a sequence of elementary links that does not contain any loops and connects s and t . Quantum-repeater nodes are then allocated in such a way that they enable the chosen paths to be used. This can be considered an instance of the set cover problem²³. To guarantee a minimum rate R_{\min} and fidelity F_{\min} , we require every chosen path to contain at most N_{\max} quantum-repeater nodes, and we require every elementary link in the path to be at most L_{\max} long. N_{\max} and L_{\max} are functions of R_{\min} and F_{\min} , and what these functions look like depends on the specific quantum-repeater implementation under consideration. For an example of how N_{\max} and L_{\max} can be derived from R_{\min} and F_{\min} , see Section “Toy-model calculation of N_{\max} and L_{\max} from R_{\min} and F_{\min} ”. Furthermore, to guarantee the network is robust, we choose K different paths per source-destination pair. They are chosen such that none of the K paths share a quantum-repeater node or an elementary link. Finally, to account for the finite capacity of quantum repeaters, we choose the paths such that every quantum-repeater node is only used by at most D

different paths. It can be intuitively understood that any quantum network accommodating the use of all these paths, will satisfy all network requirements considered in this paper.

Linear Programming Formulation 1: Path-Based Formulation

$$\min \sum_{u \in \mathcal{R}} y_u \quad (5)$$

$$\text{s.t. } L((u, v))x_p \leq L_{\max} \quad \forall (u, v) \in \mathcal{E}, p \in \mathcal{P} \quad (6)$$

$$|p| x_p \leq N_{\max} + 1 \quad \forall p \in \mathcal{P} \quad (7)$$

$$\sum_{p \in \mathcal{P}_q} x_p = K \quad \forall q \in \mathcal{Q} \quad (8)$$

$$\sum_{p \in \mathcal{P}_q} r_{up} x_p \leq 1 \quad \forall u \in \mathcal{R}, q \in \mathcal{Q} \quad (9)$$

$$\sum_{p \in \mathcal{P}} r_{up} x_p \leq D y_u \quad \forall u \in \mathcal{R} \quad (10)$$

$$x_p \in \{0, 1\} \quad \forall p \in \mathcal{P} \quad (11)$$

$$y_u \in \{0, 1\} \quad \forall u \in \mathcal{R} \quad (12)$$

$$\text{where } r_{up} = \begin{cases} 1 & \text{if path } p \text{ uses } u \text{ as a quantum-repeater node} \\ 0 & \text{otherwise} \end{cases} \quad \forall u \in \mathcal{R}, p \in \mathcal{P} \quad (13)$$

Key to the path-based formulation are the binary decision variables x_p , which are defined for every path $p \in \mathcal{P} = \cup_{(s,t) \in \mathcal{Q}} \mathcal{P}_{(s,t)}$, where $\mathcal{P}_{(s,t)}$ is the set of all possible paths from end node s to end node t . The elementary links that can be contained by a path $p \in \mathcal{P}_{(s,t)}$ must all be in $\mathcal{E}_{(s,t)}$, which is defined in Eq. (3). Each x_p has value 1 when p is considered part of the chosen set of paths, and 0 otherwise. Furthermore, there are the binary decision variables y_u for all $u \in \mathcal{R}$. y_u is 1 if a quantum repeater is placed at potential repeater location u , and 0 otherwise. Constraints (6) to (10) guarantee that these variables are chosen such that all network requirements are satisfied. The objective function (5) ensures that they are chosen such that the total number of quantum-repeater nodes is minimized. It is argued that solutions to the path-based formulation are indeed solutions to the repeater-allocation problem in Section “Explanation of path-based formulation”.

The path-based formulation requires us to define one variable x_p corresponding to each path $p \in \mathcal{P}$. Hence, the total number of variables as well as the number of constraints are at least $|\mathcal{P}|$, which is $O(|\mathcal{N}|!)$. Therefore the size of the input to the ILP solver scales exponentially with the number of nodes. This makes the path-based formulation unsuitable for designing quantum networks based on large fiber networks. Our implementation of the path-based formulation in CPLEX can be found in the repository²⁴. In the next section, we give a more efficient formulation.

Link-based formulation

Here we present the link-based formulation, which is inspired by the capacitated facility location problem²³. Instead of choosing which paths to use, we choose which elementary links to use. Quantum repeaters can then be placed such that each chosen elementary link is enabled. To this end, for each end-node pair $q \in \mathcal{Q}$, for every elementary link $(u, v) \in \mathcal{E}_q$ and for $k = 1, 2, \dots, K$, we define the binary decision variable $x_{uv}^{q,k}$. It can be thought of as indicating whether elementary link (u, v) is used in the k^{th} path used to connect end node s to end node t , where $q = (s, t)$. Furthermore, we again use the variables y_u that indicate whether node $u \in \mathcal{R}$ is used as a quantum-repeater node.

Linear Programming Formulation 2: Link-Based Formulation

$$\min \sum_{u \in \mathcal{R}} y_u \quad (14)$$

$$\text{s.t.} \quad \sum_{(u,v) \in \mathcal{E}_q} x_{uv}^{q,k} - \sum_{(v,u) \in \mathcal{E}_q} x_{vu}^{q,k} = \begin{cases} 1, & \text{if } u = s \\ -1, & \text{if } u = t \\ 0, & \text{if } u \in \mathcal{R} \end{cases} \quad \forall u \in \mathcal{R} \cup \{s, t\}, q = (s, t) \in \mathcal{Q}, k = 1, 2, \dots, K \quad (15)$$

$$L((u, v)) x_{uv}^{q,k} \leq L_{\max} \quad \forall (u, v) \in \mathcal{E}_q, q \in \mathcal{Q}, k = 1, 2, \dots, K \quad (16)$$

$$\sum_{(u,v) \in \mathcal{E}_q} x_{uv}^{q,k} \leq N_{\max} + 1 \quad \forall q \in \mathcal{Q}, k = 1, 2, \dots, K \quad (17)$$

$$\sum_{(u,v) \in \mathcal{E}_q} \sum_{k=1}^K x_{uv}^{q,k} \leq 1 \quad \forall u \in \mathcal{R}, q \in \mathcal{Q} \quad (18)$$

$$\sum_{k=1}^K x_{st}^{q,k} \leq 1 \quad \forall q \in \mathcal{Q} \quad (19)$$

$$\sum_{q \in \mathcal{Q}} \sum_{(u,v) \in \mathcal{E}_q} \sum_{k=1}^K x_{uv}^{q,k} \leq D y_u \quad \forall u \in \mathcal{R} \quad (20)$$

$$x_{uv}^{q,k} \in \{0, 1\} \quad \forall (u, v) \in \mathcal{E}_q, q \in \mathcal{Q}, k = 1, 2, \dots, K \quad (21)$$

$$y_u \in \{0, 1\} \quad \forall u \in \mathcal{R} \quad (22)$$

Because the number of elementary links scales polynomially with the number of nodes, both the number of variables and the number of constraints also scale polynomially with the number of nodes $|\mathcal{N}|$. In particular, they are $O(|\mathcal{N}|^2)$ (see Section “Scaling of the formulations” for a derivation). Our implementation of the link-based formulation in CPLEX can be found in the repository²⁴.

In Section “Proof of equivalence”, we sketch the proof of the equivalence of the path-based formulation and the link-based formulation. Furthermore, we sketch why the variables $x_{uv}^{q,k}$ and y_u still provide a solution to the link-based formulation after performing step 7 of Method 1. The reason this step is included in our method is because, otherwise, elementary links could be included in the solution which are not necessary to meet the network requirements. The detailed version of the proof can be found in Supplementary Note 2. Since the link-based formulation scales much more favorably with the size of the fiber network under consideration, it is more efficient to use this formulation when solving the repeater-allocation problem for large networks.

DISCUSSION

In this section we illustrate our method as implemented by the link-based formulation using the Python API of CPLEX version 12.9²². The corresponding code can be found in the repository²⁴. Furthermore, we investigate the effect of varying network-requirement parameters and discuss possible extensions and limitations of our method.

We first demonstrate our method by solving the repeater-allocation problem for a real fiber network. The fiber network that we consider is the core network of SURFnet. The latter is a network provider for Dutch educational and research institutions and has provided us with the network data, which is available in the repository²⁴. The network graph is depicted in Fig. 2.

As end nodes of the network, we have chosen the cities of Delft, Enschede, Groningen and Maastricht. In this example, we consider an entanglement-based quantum network utilizing massive multiplexing as described in e.g.²⁵. For the end nodes, we require

a minimum rate of $R_{\min} = 1$ Hz (one entangled state per second) and a fidelity to a maximally entangled state $F_{\min} = 0.93$. Furthermore, we set the robustness parameter to $K = 2$ (thus requiring that any single quantum repeater or elementary link in the network can break down without compromising network functionality), and we set the capacity parameter to $D = 4$ (which, in this case, means that we assume each quantum repeater can perform four Bell-state measurements simultaneously).

The first step of our method requires us to calculate the L_{\max} and N_{\max} corresponding to the minimal rate and fidelity we have chosen. This requires us to study the behavior of a quantum-repeater chain consisting of $N + 1$ elementary links of length L each. L_{\max} and N_{\max} then have to be chosen as the largest possible values for L and N respectively such that the repeater chain still achieves the required rate and fidelity. Here, we make a couple of simplifying assumptions to make the calculations more tractable. Particularly, we assume elementary links generate Werner states, and we assume that the only losses are due to fiber attenuation and probabilistic Bell-state measurements (which we take to have a 50% success probability). In Section “Toy-model calculation of N_{\max} and L_{\max} from R_{\min} and F_{\min} ”, we perform the calculation and find that for an elementary-link fidelity $F_{\text{link}} = 0.99$, number of multiplexing modes $M = 1000$, speed of light in fiber $c_{\text{fiber}} = 200,000$ km s⁻¹ and attenuation length $L_{\text{att}} = 22$ km, we have $N_{\max} = 6$ and $L_{\max} = 136$ km.

The rest of the steps of the method in Results have been performed using a Python script and CPLEX²⁴. The resulting solution is shown graphically in Fig. 3. All chosen repeater nodes are shown as blue hexagons, while all fibers that are used in elementary links are drawn as thick lines. We see that repeaters are placed around Groningen in order to bridge the large distance to the other end nodes without exceeding the maximum elementary-link length L_{\max} . Additionally, placing quantum-repeater nodes close to Groningen means they can be used for several of Groningen’s outgoing connections. There are multiple such nodes close together because each only has a limited capacity ($D = 4$), and the redundancy increases the robustness of the network.

On our setup (see Results), it took us approximately 74 s to find the optimal solution to the link-based formulation for this network. Note that a feasible solution is a combination of decision variable values that satisfy all the constraints, while the optimal solution is a feasible solution that also minimizes the objective function.

Next, we demonstrate and investigate the effect of the different network-requirement parameters on the outcome of our method. The network-requirement parameters are, in principle, the minimum rate R_{\min} , the minimum fidelity F_{\min} , the robustness parameter K and the capacity parameter D . However, since R_{\min} and F_{\min} are translated into a maximum number of repeaters N_{\max} and a maximum elementary-link length L_{\max} in our method, we here consider the network-requirement parameters to be N_{\max} , L_{\max} , K and D . This way, we can keep our discussion agnostic about the exact hardware used to create a quantum network and how R_{\min} and F_{\min} are mapped to N_{\max} and L_{\max} .

First we give a visual demonstration on how the different network-requirement parameters affect the repeater placement. To this end, we have created a network graph with end nodes in the corners of the network and 10 possible repeater locations randomly distributed in between the end nodes. For details on how the graph was obtained, see Section “Generating random networks”. While keeping the network fixed, we vary the network-requirement parameters D , K and L_{\max} . In Fig. 4a to c, we explore how the robustness parameter influences the total number of required quantum repeaters. Since each repeater has a capacity of $D = 6$ to distribute entanglement between the six end-node pairs, and because the network is set up in such a way that each path needs exactly one quantum-repeater node to connect end nodes without elementary links exceeding $L_{\max} = 0.9$, the optimal solution always contains K repeaters. In Fig. 4d to f on the other hand, we see that as the capacity of quantum repeaters is varied from $D = 1$ to $D = 3$, the required

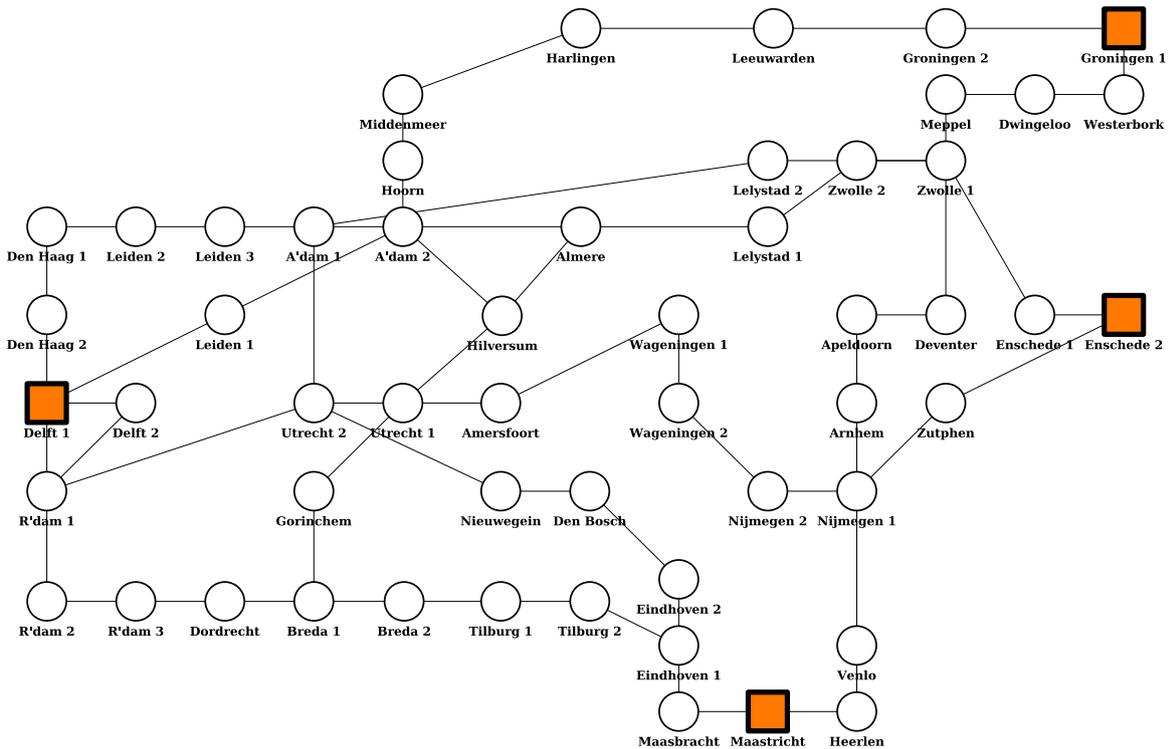


Fig. 2 Graph representation of SURFnet core network. Node locations roughly correspond to geographical locations but have been adjusted for readability. Lengths of fibers connecting nodes are not shown. Nodes that are used as end nodes are shown as orange squares. Potential repeater locations are shown as white circles. A'dam and R'dam are used as abbreviations for Amsterdam and Rotterdam respectively.

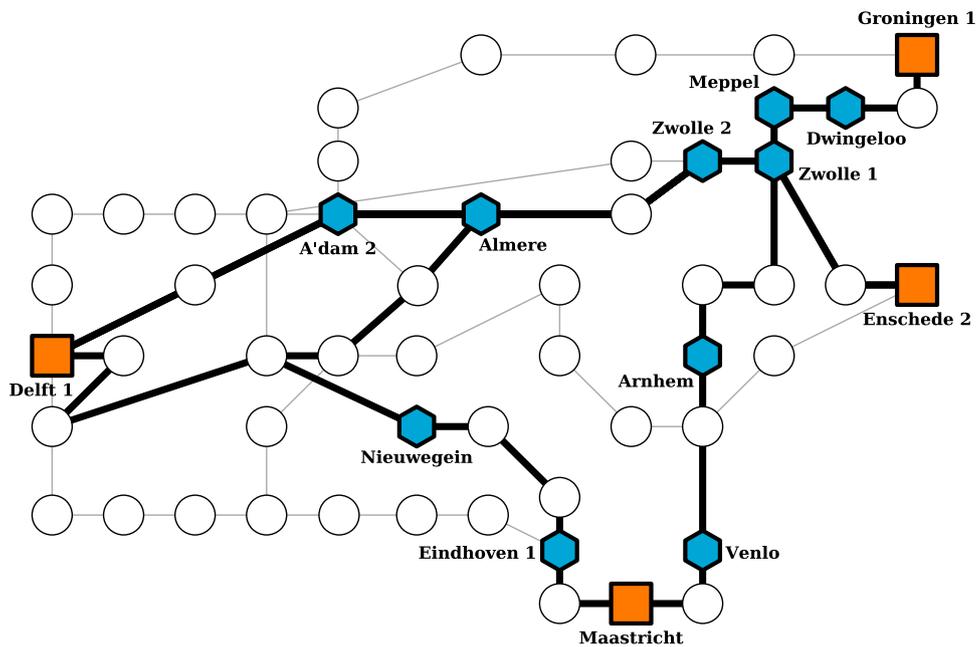


Fig. 3 Solution to the repeater-allocation problem on SURFnet network. Solution to the repeater-allocation problem for $R_{\min} = 1$ Hz, $F_{\min} = 0.93$, $K = 2$ and $D = 4$. The network graph used as input corresponds to the SURFnet network, depicted in Fig. 2. End nodes are shown as orange squares, quantum-repeater nodes are shown as blue hexagons and the fibers that are used in the elementary links are highlighted with thick lines.

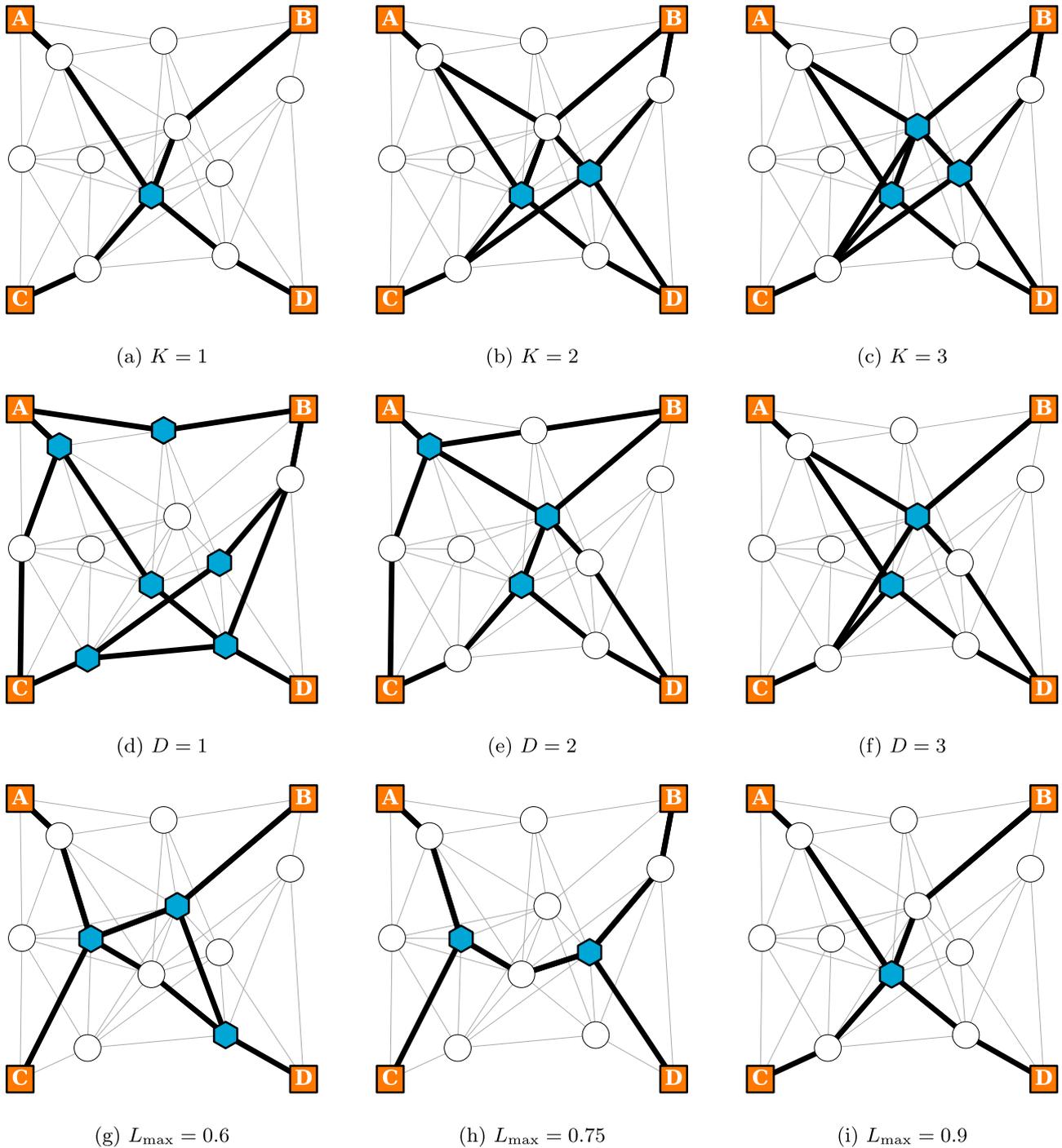


Fig. 4 Solutions obtained using our method for an example network graph. Solutions obtained using our method for an example network graph using the network-requirement parameters $L_{\max} = 0.9$, $N_{\max} = 3$, $K = 1$ and $D = 6$, unless noted otherwise in the caption of a specific solution. **a–c** Visualization of the effect of K . A higher robustness implies that we require more repeaters. **d–f** Visualization of the effect of D . As the capacity of quantum-repeater nodes increases, multiple paths can use the same repeater and hence the overall number of repeaters decreases. **g–i** Visualization of the effect of L_{\max} . When longer elementary-link lengths are allowed, less quantum-repeater nodes are required to bridge the distance between end nodes.

number of quantum repeaters decreases when D increases. Note that since $K = 1$, the optimal solution here always happens to contain $|Q|/D$ repeaters. Finally, in Fig. 4g–i, we see that as we allow for longer elementary links to be used, the total number of repeaters is decreased. If we would increase L_{\max} even further, at a certain point every end node can be connected to another end node with a direct

elementary link and hence the number of repeaters will drop to zero. The degeneracy of the optimal solution is visible from the fact that the solutions with two repeaters for $K = 2$ (Fig. 4b), $D = 3$ (Fig. 4f) and $L_{\max} = 0.75$ (Fig. 4h) are not equal. Later, in this section, it is discussed how this degeneracy can be lifted (see Linear Programming Formulation 3). We do not show the effect of N_{\max} . Since the

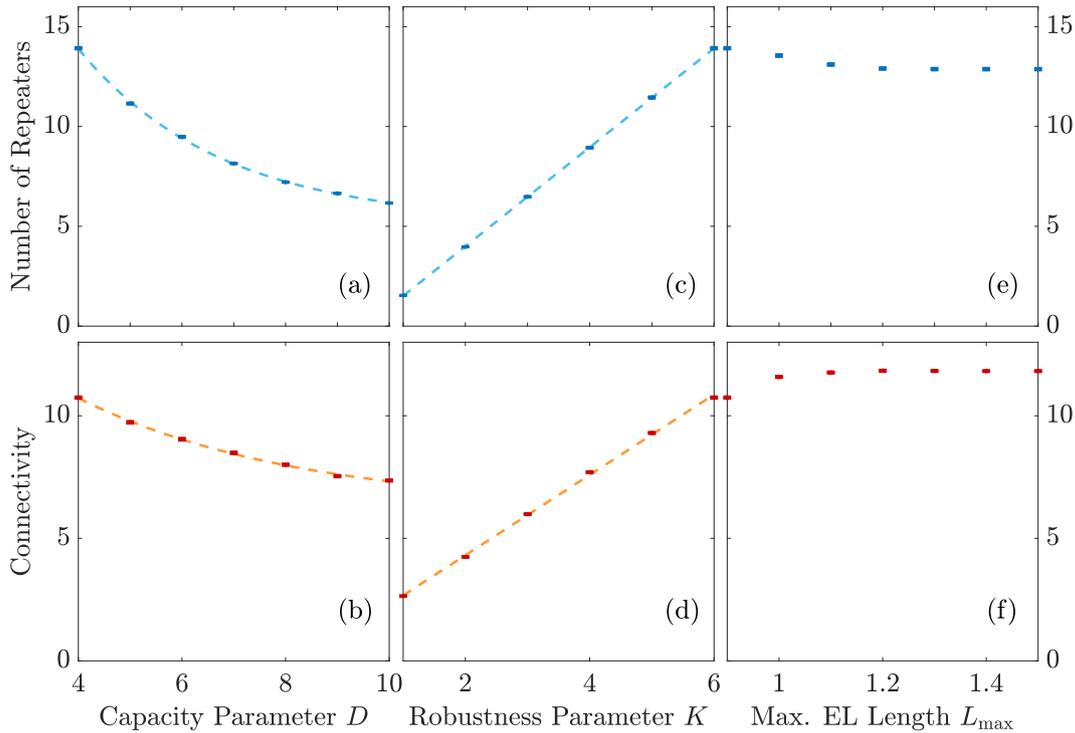


Fig. 5 Simulation on random geometric graphs. Simulation on 1000 random geometric graphs with a radius of $d_{\max} = 0.9$ and $n = 25$ nodes for varying network parameters. We use $L_{\max} = 0.9$, $N_{\max} = 6$, $K = 6$ and $D = 4$, except for the varied parameter. In the plots, each of the points represents the average number of placed repeaters or average connectivity over all samples for each value of (a, b) the capacity parameter D , (c, d) the robustness parameter K or (e, f) the maximum elementary-link length L_{\max} . We either use a linear or an exponential function for the fits. The error bars represent one standard deviation of the mean. Solving an instance to optimality requires approximately 30 s on average.

total number of repeaters is already minimized, changing the value of N_{\max} does not change the repeater allocation, but only determines whether a feasible solution exists at all.

Considering how the repeater placement on a single network varies with the network-requirement parameters can offer insight into how our method operates. However, it does not provide a general investigation into the properties of our method. In order to make more general and quantitative statements about our method, we will next consider the effect of varying network-requirement parameters on the repeater allocation for an ensemble of random networks. In this work, we construct random network graphs using random geometric graphs. That is, network graphs are constructed by scattering nodes randomly on a unit square. Edges are put only between nodes if the Euclidean distance separating them is smaller than some number, which is called the radius of the random geometric graph. The nodes which form the convex hull of the network are chosen as end nodes, so that the others are potential repeater locations. This choice is motivated by the fact that any potential-repeater locations that do not lie between end nodes would probably not play an important role anyway. For a more elaborate account of how we generate random network graphs, see Section “Generating random networks”.

We here report how the number of placed repeaters and the (vertex) connectivity of quantum networks designed using our method vary as a function of the network-requirement parameters. The number of placed repeaters is interesting to consider since the aim of our method is to minimize this. On the other hand, the connectivity is interesting since it lower bounds the minimum number of quantum repeaters that need to break down before any pair of end nodes becomes disconnected, thereby giving an indication of how robust a quantum network is. Note that

connectivity is not the same as the robustness parameter K , which lower bounds the minimum number of quantum repeaters or elementary links that need to break down before end nodes can no longer distribute entanglement with a minimum rate and fidelity, while at the same time taking repeater capacity into account. We have first generated 1000 random network graphs for which our method was able to find solutions for the parameter values $L_{\max} = 0.9$, $N_{\max} = 6$, $K = 6$ and $D = 4$. Then, while keeping all other parameters constant, we have varied each of the parameters D , K and L_{\max} . This has been done in such a way that all considered values are less restrictive than the original values, such that we can be sure that a solution exists for each parameter value. Of each resulting quantum network, we determine the number of repeaters and the connectivity, and for each parameter value we determine the average number of repeaters and the average connectivity over all 1000 quantum networks.

In Fig. 5a, b we show the number of repeaters and the connectivity as a function of the repeater capacity D . We see that both the number of repeaters and the connectivity decrease as D increases, and they both accurately follow an exponential fit in the domain under consideration. In Fig. 5c, d, we show how the number of repeaters and connectivity vary as a function of the robustness parameter K . We see that both increase linearly in the domain under consideration. For D (K) the number of repeaters decreases (increases) following the same line of reasoning as we mentioned above for the visual demonstration. Generally, we expect the connectivity to follow the change in the number of repeaters, because a network with less quantum repeaters is easier to disconnect. Finally, in Fig. 5e, f, we investigate the effect of L_{\max} on the number of repeaters and connectivity. While the number of repeaters decreases, the connectivity increases, although they both

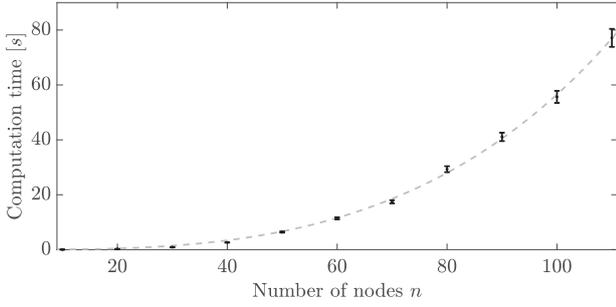


Fig. 6 Computation time for random geometric graphs. Computation time in seconds for 100 random geometric graphs with $d_{\max} = 0.9$, $L_{\max} = 1$, $N_{\max} = 6$, $K = 2$ and $D = 8$ for varying number of nodes n . The error bars represent one standard deviation of the mean. For the fit we have used an exponential function of the form $a(e^{bn^3} - 1)$, where a and b are free parameters.

flatten from $L_{\max} = 1.2$. The number of repeaters does not decrease to zero because $K = 6$. Therefore, even if L_{\max} is large enough to allow for paths between end nodes with zero quantum-repeater nodes, there are still at least five quantum-repeater nodes required to make the network robust against the breakdown of direct elementary links between end nodes. On the other hand, the connectivity increases since it also takes paths through other end nodes into account in its computation, and with an increasing value of L_{\max} , we expect more direct elementary links to appear.

Even though the link-based formulation has a scaling of $O(|\mathcal{N}|^2)$ in terms of the number of variables and constraints, it remains an ILP. In general, ILP's are NP-hard and thus generally require an exponentially increasing amount of time to solve. In order to investigate the performance of our method for varying network sizes, we determined the computation time for finding an optimal solution as a function of the number of nodes. The result is shown in Fig. 6, in which we see that the computation time indeed increases exponentially. Nonetheless, instances on random geometric graphs with 100 nodes can be solved to optimality in about 1 min on our setup (see Results).

The computation time can be strongly affected by the network topology and the chosen parameter values, since these can alter the difficulty of finding an optimal solution as well as the number of variables and constraints (see Section “Scaling of the formulation”). However, the parameters that we use for Fig. 6 are neither very strict nor loose and provide us with insight into the approximate scaling of the computation time, rather than the worst-case behavior. Note that we expect that, in practical use cases, the topology and the parameter values will be determined once and remain more or less fixed, which implies that the repeater-allocation problem will not need to be solved repeatedly. This makes the increasingly large computation time for sizable graphs or stringent parameters less problematic.

There are various ways in which our method can be extended. Here, we present two possible extensions. Such extensions change the ILP formulation in Section “Link-based formulation”. The result of these is the generalized link-based formulation, which is presented in Linear Programming Formulation 3. In order to incorporate the extensions into Method 1, the generalized link-based formulation must be used where otherwise the link-based formulation would be used.

Linear Programming Formulation 3: Generalized Link-Based Formulation

$$\min \sum_{u \in \mathcal{R}} y_u + a \sum_{q \in \mathcal{Q}} \sum_{(u,v) \in \mathcal{E}_q} \sum_{k=1}^{K^q} L((u,v)) x_{uv}^{q,k} \quad (23)$$

$$\text{s.t.} \quad \sum_{(u,v) \in \mathcal{E}_q} x_{uv}^{q,k} - \sum_{(v,u) \in \mathcal{E}_q} x_{vu}^{q,k} = \begin{cases} 1, & \text{if } u = s \\ -1, & \text{if } u = t \\ 0, & \text{if } u \in \mathcal{R} \end{cases} \quad \forall u \in \mathcal{R} \cup \{s, t\}, q = (s, t) \in \mathcal{Q}, k = 1, 2, \dots, K^q \quad (24)$$

$$L((u,v)) x_{uv}^{q,k} \leq L_{\max}^q \quad \forall (u,v) \in \mathcal{E}_q, q \in \mathcal{Q}, k = 1, 2, \dots, K^q \quad (25)$$

$$\sum_{(u,v) \in \mathcal{E}_q} x_{uv}^{q,k} \leq N_{\max}^q + 1 \quad \forall q \in \mathcal{Q}, k = 1, 2, \dots, K^q \quad (26)$$

$$\sum_{(u,v) \in \mathcal{E}_q} \sum_{k=1}^{K^q} x_{uv}^{q,k} \leq 1 \quad \forall u \in \mathcal{R}, q \in \mathcal{Q} \quad (27)$$

$$\sum_{k=1}^{K^q} x_{st}^{q,k} \leq 1 \quad \forall q \in \mathcal{Q} \quad (28)$$

$$\sum_{q \in \mathcal{Q}} \sum_{(u,v) \in \mathcal{E}_q} \sum_{k=1}^{K^q} x_{uv}^{q,k} \leq D_u y_u \quad \forall u \in \mathcal{R} \quad (29)$$

$$x_{uv}^{q,k} \in \{0, 1\} \quad \forall (u,v) \in \mathcal{E}_q, q \in \mathcal{Q}, k = 1, 2, \dots, K^q \quad (30)$$

$$y_u \in \{0, 1\} \quad \forall u \in \mathcal{R} \quad (31)$$

The first extension we can make is solving the repeater-allocation problem in case of heterogeneous network requirements. So far, we have considered the network requirements to be homogeneous, i.e. the same throughout the network. However, it can be the case that some end nodes require a higher rate and fidelity, that some end nodes need access to more robust quantum communication, or that quantum repeaters with a larger capacity can be placed at some potential repeater locations than at other. Then, we can define the network-requirement parameters on a per-end-node-pair or per-node basis. Specifically, for every pair of end nodes $q \in \mathcal{Q}$, we define the minimum rate R_{\min}^q and fidelity F_{\min}^q of entanglement generation, and the required robustness parameter K^q (in order to break communication between the end nodes q , at least K^q quantum repeaters or elementary links must be incapacitated). Furthermore, for every potential repeater location $u \in \mathcal{R}$, we define the quantum-repeater capacity D_u . To incorporate this into the method, the input parameters must be adapted accordingly, and the maximum number of repeaters and maximum elementary-link length must be calculated for every pair of end nodes separately (i.e. L_{\max}^q and N_{\max}^q must be determined from R_{\min}^q and F_{\min}^q for each $q \in \mathcal{Q}$).

A second extension has to do with the fact that the link-based formulation in Section “Link-based formulation” typically has a highly-degenerate optimal solution. That is, often there are multiple possible quantum-repeater placements for which all constraints are satisfied and the total number of quantum-repeater nodes is minimal. However, it might be the case that some solutions are more desirable than others. To pick out these solutions, one can define a secondary objective. This secondary objective can then be taken into account by defining a corresponding objective function, and adding it to the existing objective function, while scaling it such that it does not influence the optimal number of repeaters. In particular, the scale factor a should be chosen such that the secondary objective value does not exceed 1. This can be seen as a form of weighted goal programming²⁶. As an example, in Linear

Programming Formulation 3, we use as secondary objective to minimize the total length of all used elementary links. Other secondary objectives, such as minimizing the largest elementary-link length, could be implemented in a similar fashion.

Finally, we discuss some of the limitations of the method we present in this work. Each limitation represents a way that our method could be further extended, but is beyond the scope of this paper.

A first major limitation is the complexity of ILP's. While we provide an efficient ILP formulation, in which the number of variables and constraints scales polynomially with the network size, it remains an ILP. This cannot be helped, as choosing whether a repeater should be placed at a certain potential repeater location is inherently binary. In general, it is NP-hard to solve an ILP. While we indeed observe exponential scaling of the computation time in Fig. 6, we are able to find optimal solutions of realistically-sized networks within tractable time using CPLEX, which is also demonstrated using a real network in Fig. 3. Conceivably, one can use heuristics or approximation algorithms to obtain solutions faster, although the solutions then may no longer be optimal.

Another limitation that we consider here is the fact that our method is agnostic about how elementary links are constructed. We assume that any number of fibers can be combined to form an elementary link. However, quantum-repeater protocols relying on heralded entanglement generation typically require the presence of a midpoint station with the capability to perform Bell-state measurements¹⁹. If there are constraints on the placement of such stations, our method is insufficient. Conceivably, if such stations can only be placed at potential repeater locations, a modified version of our method could be used. Furthermore, we assume that an elementary link between two nodes is always constructed from the fibers which minimize the elementary-link length such that rate and fidelity are maximized. However, if one would like to incorporate the number of fibers (rather than elementary links) that need to be disabled before the quantum network is incapacitated as an additional network requirement (thereby guaranteeing more robustness), this may no longer be a useful assumption. It may then be better to try to construct different elementary links from different fibers as much as possible, such that individual fibers do not become too critical.

METHODS

Explanation of path-based formulation

In Section "Path-based formulation", we introduced the path-based formulation and we claim that solutions to the path-based formulation can be used to construct solutions to the repeater-allocation problem. Here, we show how and why this can be done.

The idea behind the path-based formulation is to choose a combination of feasible paths that minimize the overall number of utilized repeaters. If a path is chosen that uses potential repeater location $u \in \mathcal{R}$ as a quantum-repeater node, a repeater should be placed at u . The binary variables x_p are used to parameterize the chosen paths, while the binary variables y_u are used to parameterize where quantum repeaters should be placed. A coupling between these variables is realized by Constraints (10): if a path p is chosen in which a node $u \in \mathcal{R}$ is used as quantum-repeater node, the corresponding y_u variables must have value 1. Conversely, when $y_u = 1$ for a given repeater node $u \in \mathcal{R}$, up to D paths can use this repeater node in order for the corresponding constraint to hold, thereby also imposing a limit on the repeater capacity. After all, if $\sum_{p \in \mathcal{P}_q} r_{up} x_p > D$ then more than D paths are chosen in which node $u \in \mathcal{R}$ is used as a repeater, which renders the solution infeasible.

Paths are moreover only considered useful if they can be used to deliver entanglement between end nodes with the minimum required rate R_{\min} and fidelity F_{\min} . In the path-based formulation, this is implemented by requiring chosen paths to contain at most $N_{\max} + 1$ elementary links, each with a length of at most L_{\max} . The values of N_{\max} and L_{\max} can be determined from R_{\min} and F_{\min} as detailed in Results. These requirements are straightforwardly enforced by Constraints (6) and (7). Constraints (6) can only hold when $x_p = 0$ for all paths that contain an elementary link

$((u, v) \in p)$ which is too long ($L((u, v)) > L_{\max}$). Similarly, Constraints (7) can only hold when $x_p = 0$ for all paths for which the number of elementary links ($|p|$) exceeds the maximum ($N_{\max} + 1$).

Furthermore, the choice of paths must be such that it is guaranteed that up to K potential repeater nodes or elementary links can break down before there is no path available between any pair of end nodes that can deliver entanglement at the required rate and fidelity. This is implemented by choosing, per pair of end nodes, K different paths. All of these paths are chosen such that none of them share a quantum-repeater node. Since elementary links connect quantum-repeater nodes, this automatically also means that none of the paths share an elementary link. Therefore, when a quantum-repeater node or elementary link becomes incapacitated, this can disrupt at most one path between a pair of end nodes. When there are K break downs, in the worst case, this can disrupt all paths between a pair of end nodes. But as long as there are fewer break downs, there will be at least one path available.

Since every chosen path can deliver entanglement at the required rate and fidelity, this guarantees robustness of the quantum network against up to K break downs. It is enforced by Constraints (8) that there are exactly K paths chosen between every pair of end nodes. Furthermore, Constraints (9) make sure that the number of chosen paths connecting a pair of end nodes using u as a quantum-repeater node ($\sum_{p \in \mathcal{P}_q} r_{up} x_p$) is at most one, thereby guaranteeing that all K paths are disjoint. Note that, when considering the quantum-repeater capacity, all chosen paths are taken into account. In other words, Constraints (10) guarantee that the repeater capacity is not exceeded when all paths are used simultaneously. Therefore, if one path between a pair of end nodes is disrupted and they are forced to switch to another path, it is guaranteed that none of the quantum repeaters along that path are overloaded.

It is now easy to obtain a solution to the repeater-allocation problem from the solution to the path-based formulation. Every potential repeater location $u \in \mathcal{R}$ for which $y_u = 1$ in the solution to the path-based formulation should be used as a quantum-repeater node. Furthermore, each elementary link which is part of a chosen path $((u, v) \in p$ for which $x_p = 1$) should be constructed. This is done using the fibers making it up ($F((u, v))$). Then, the resulting quantum network will be such that all network requirements are satisfied. Furthermore, the number of quantum-repeater nodes will be minimal. This is because this number, which is exactly $\sum_{u \in \mathcal{R}} y_u$, is minimized by the objective function (5) of the path-based formulation. Therefore, the path-based formulation can indeed be used to solve the repeater-allocation problem.

Toy-Model calculation of N_{\max} and L_{\max} from R_{\min} and F_{\min}

In this section we calculate the maximum number of repeaters and maximum elementary-link length from the minimum required rate R_{\min} and fidelity F_{\min} using a toy model of a quantum-repeater chain. The quantum-repeater architecture that we consider is of the massively-multiplexed type as described in e.g.²⁵. The toy model that we consider here makes the following simplifying assumptions:

- the states distributed over elementary links are Werner states,
- the noise in the states distributed over elementary links is the only noise,
- the only sources of photon loss are fiber attenuation and non-deterministic Bell-state measurements,
- all processes except light traveling through fiber are instantaneous.

It is shown in Supplementary Note 1 that in this model, a repeater chain with N quantum repeaters, M entanglement-distribution attempts per round per elementary link, elementary-link length L , elementary-link fidelity F_{link} , speed of light in fiber c_{fiber} and a 50% Bell-state measurement success probability has the following end-to-end rate R and fidelity F :

$$R = \frac{c_{\text{fiber}}}{L} \left(\frac{1}{2} \right)^N \left[1 - \left(1 - \frac{1}{2} e^{-L/L_{\text{att}}} \right)^M \right]^{N+1}, \quad (32)$$

$$F = \frac{1}{4} \left[1 + 3 \left(\frac{4F_{\text{link}} - 1}{3} \right)^{N+1} \right]. \quad (33)$$

N_{\max} can now be obtained from the fidelity. Specifically, it is the lowest-integer solution to the equation

$$F > F_{\min}. \quad (34)$$

To find L_{\max} , we can put the resulting value of N_{\max} into the equation

$$R > R_{\min}. \quad (35)$$

The smallest value for L that solves Equation (35) is then L_{\max} . Note that the calculation here is somewhat simplified because the fidelity is not a function of L_{\max} . If both fidelity and rate would be functions of N and L , there would not exist a unique solution. In that case, there is some freedom in choosing N_{\max} and L_{\max} .

The calculation of N_{\max} and L_{\max} for the example parameters $F_{\min} = 0.93$, $R_{\min} = 1$ Hz, $F_{\text{link}} = 0.99$, $c_{\text{fiber}} = 200,000$ km s⁻¹, $M = 1000$ and $L_{\text{att}} = 22$ km results in $N_{\max} = 6$ and $L_{\max} = 136$ km (rounded down).

Proof of equivalence

In this section we briefly outline the proof of why the path-based formulation and the link-based formulation are equivalent. The main idea is to use an optimal solution to the path-based formulation to construct a feasible solution to the link-based formulation and vice versa. We prove that this is always possible in such a way that the value of the objective function of the constructed feasible solution is the same as that of the original optimal solution. This can be used to show that the optimal objective values of both formulations are always the same. Therefore, the feasible solution to one formulation constructed from an optimal solution to another formulation is itself an optimal solution. We say that two ILP formulations are equivalent if optimal solutions to one can be obtained from the other and vice versa, and therefore we conclude that the path-based formulation and the link-based formulation are equivalent.

To construct a solution to the link-based formulation using a solution to the path-based formulation, we use the elementary links that appear in chosen paths. More specifically, for each $q = (s, t) \in \mathcal{Q}$ and $k = 1, 2, \dots, K$, we set $x_{uv}^{q,k} = 1$ if elementary link $(u, v) \in \mathcal{E}_q$ is in the k^{th} chosen path connecting s and t . Conversely, Constraints (15) guarantee that, for every $q = (s, t) \in \mathcal{Q}$ and $k = 1, 2, \dots, K$, the elementary links $(u, v) \in \mathcal{E}_q$ for which $x_{uv}^{q,k} = 1$ can be used to form exactly one path between s and t . These paths can be obtained by using Algorithm 1, which outputs the set \mathcal{P}^* that contains the extracted paths over all $q \in \mathcal{Q}$ and $k = 1, 2, \dots, K$. Thus, we can construct a solution to the path-based formulation from a solution to the link-based formulation by setting $x_p = 1$ for all $p \in \mathcal{P}^*$. Furthermore, the repeater-placement variables y_u are kept the same when translating between formulations.

By comparing the different constraints, it can be understood that if a solution to one formulation is feasible, the solution to the other formulation that can be obtained from it is also feasible. Constraints (6) and (16) both guarantee that elementary-link lengths do not exceed L_{\max} , while Constraints (7) and (17) both guarantee that each path includes N_{\max} quantum-repeater nodes at maximum. Constraints (8) and (15) make sure there are K paths between each pair of end nodes. These paths are guaranteed to be disjoint for the path-based formulation by Constraints (9) and for the link-based formulation by Constraints (18) and (19). Lastly, Constraints (10) and (20) couple the x variables to the y variables and make sure the quantum-repeater capacity is taken into account.

In step 7 of Method 1, we manually set $x_{uv}^{q,k} = 0$ for all elementary links $(u, v) \in \mathcal{E}$ which are not in one of the paths $p \in \mathcal{P}^*$. We do this because, on some occasions, the variables $x_{uv}^{q,k}$ are allowed to have value 1 in such a way that they form loops (which are disjoint from the path between s and t). For example, it could be the case that for some $q \in \mathcal{Q}$, $k = 1, 2, \dots, K$ and $u_1, u_2 \in \mathcal{R}$, it holds that $x_{u_1 u_2}^{q,k} = x_{u_2 u_1}^{q,k} = 1$, which does not violate any of the constraints in Section "Link-based formulation", and also does not influence the objective function (14). Since these loops do not connect end nodes, they do not contribute to realizing any of the network requirements. Therefore, any variable $x_{uv}^{q,k}$ with value 1 such that it is part of a loop can safely be set to 0 without violating any constraint. This is shown rigorously in Supplementary Note 2. Only allowing for elementary links which are part of paths between end nodes realizes the removal of such loops. Since the method in Results recommends the construction of elementary link $(u, v) \in \mathcal{E}$ if $x_{uv}^{q,k} = 1$, setting them to 0 whenever this is possible helps to prevent the construction of unnecessary elementary links. One way in which the appearance of loops in optimal solutions can be prevented in the first place by is to use the generalized link-based formulation in Discussion. In this formulation, the minimization of the total elementary-link length is used as secondary objective and hence the presence of loops will increase the objective function.

Algorithm 1. Path extraction algorithm.

```

 $\mathcal{P}^* = \emptyset;$ 
for  $q = (s, t) \in \mathcal{Q}$  do
  for  $k = 1, 2, \dots, K$  do
     $u_0 = s;$ 
     $n = 0;$ 
    while  $u_n \neq t$  do
      Find the unique node  $v \in \mathcal{R} \cup \{t\}$  for which  $x_{u_n v}^{q,k} = 1;$ 
       $n = n + 1;$ 
       $u_n = v;$ 
    end
     $p = ((s, u_1), (u_1, u_2), \dots, (u_{n-1}, t));$ 
     $\mathcal{P}^* = \mathcal{P}^* \cup p;$ 
  end
end
end

```

Generating random networks

Here, we describe how we generate random network graphs based on random geometric graphs. These networks are used to demonstrate our method and study the effect of different network-requirement parameters in Discussion.

The recipe for generating a random geometric graph on a two-dimensional Euclidean space with n nodes and radius d_{\max} is as follows²⁷. First, n points are distributed uniformly at random on a unit square, by sampling both their horizontal and vertical coordinates uniformly at random. To every two points p_1, p_2 we associate $d(p_1, p_2)$, which is the Euclidean distance between the two points. From this, an undirected weighted graph is constructed in which every node corresponds to one of the points, and edges between nodes corresponding to points p_1, p_2 are added if $d(p_1, p_2) \leq d_{\max}$. The weight that is given to the edge is $d(p_1, p_2)$.

To turn a random geometric graph into a suitable network graph, it must be decided which of the nodes are end nodes, and which are potential repeater locations. To this end, we determine the convex hull of the graph. We choose to use nodes corresponding to vertices of the convex hull of the graph as end nodes, i.e. they make up the set \mathcal{C} . All other nodes are thus considered potential repeater locations, i.e. they make up the set \mathcal{R} . This method is used because it is expected that potential repeater locations lying outside of the area spanned by the end nodes will only rarely be chosen as quantum-repeater nodes. When the end nodes form the convex hull, there are no such potential repeater locations, and the number of nodes that are not of relevance to the repeater-allocation problem is minimized. We generate the random geometric graphs using NetworkX²⁸ and determine the convex hull using an algorithm²⁹ which is included in SciPy³⁰.

The random network graph used in Fig. 4 has been based on a random geometric graph with $n = 10$ and $d_{\max} = 0.6$, but has been further edited to be made suitable for demonstration purposes. Some nodes were displaced manually. Additionally, end nodes have been added at the corners of the unit square and connected to the three closest potential repeater locations.

Scaling of the formulations

The path-based formulation relies on the enumeration of all the paths between two end nodes. For every pair $(s, t) \in \mathcal{Q}$ we must consider all possible permutations of intermediate nodes in which r repeaters are placed on a path. For $r = 0$, we get a single path directly from s to t and for $r = 1$ we should consider all possible paths that utilize one repeater, which are $|\mathcal{R}|$ in total. Next, when $r = 2$ we must consider all paths that contain exactly two repeaters and additionally all permutations of the repeater placements in these paths, which gives $|\mathcal{R}|(|\mathcal{R}| - 1)$ paths in total, et cetera. The number of y_u variables is $|\mathcal{R}|$, so that the number of variables $n_{\text{var}}^{\text{pbf}}$ of the path-based formulation is given by

$$n_{\text{var}}^{\text{pbf}} = |\mathcal{R}| + |\mathcal{Q}| |\mathcal{P}_q| \quad (36)$$

$$= |\mathcal{R}| + |\mathcal{Q}| \sum_{r=0}^{|\mathcal{R}|} \frac{|\mathcal{R}|!}{(|\mathcal{R}| - r)!}. \quad (37)$$

If $|\mathcal{R}| > 1$, this simplifies to

$$n_{\text{var}}^{\text{pbf}} = |\mathcal{R}| + |\mathcal{Q}| [e^{|\mathcal{R}|}], \quad (38)$$

where e denotes Euler's number and $\lceil \cdot \rceil$ represents the rounding operator. We assume that the number of end nodes, and therefore the number of end-node pairs $|\mathcal{Q}| = |\mathcal{C}|(|\mathcal{C}| - 1)/2$ is constant and therefore does not scale with the total number of nodes $|\mathcal{N}|$ in our graph. This implies that the number of possible repeater locations $\mathcal{R} = \mathcal{N} \setminus \mathcal{C}$ scales linearly with the number of nodes. The number of variables, as well as the number of constraints, is thus $O(|\mathcal{N}|^2)$.

One important detail of our implementation of the path-based formulation is that we take Constraints (6) and (7) into account while enumerating all the paths. If we encounter a path which contains an elementary link with a length that exceeds L_{\max} or which uses more than N_{\max} repeaters, we simply exclude it from the formulation. This can greatly reduce the total number of variables, although it will remain to scale exponentially with $|\mathcal{N}|$.

In the link-based formulation, we need to enumerate all the elementary links in the network. To this end, we need to count every elementary link from s to every node $v \in \mathcal{R} \cup \{t\}$, and from $u \in \mathcal{R}$ to t which results in $2|\mathcal{R}| + 1$ elementary links. Next, we also need to consider the elementary link from every node $u \in \mathcal{R}$ to $v \in \mathcal{R}$ and back, in order to allow for directional paths from s to t , which are $|\mathcal{R}|(|\mathcal{R}| - 1)$ in total. Additionally, since we use the index k for our $x_{uv}^{q,k}$ variables in order to keep track of the redundant paths that are required for the given level of robustness, we need to make a copy of these variables for every value of $k = 1, 2, \dots, K$. When we combine this with the $|\mathcal{R}|y_u$ variables, we get that the total number of variables of the link-based formulation is given by

$$n_{\text{var}}^{\text{lb}} = |\mathcal{R}| + K|\mathcal{Q}||\mathcal{E}_q| \quad (39)$$

$$= |\mathcal{R}| + K|\mathcal{Q}|(|\mathcal{R}|^2 + |\mathcal{R}| + 1), \quad (40)$$

which is $O(|\mathcal{N}|^2)$, if we assume that K is a fixed constant. Note that the link-based formulation therefore also has $O(|\mathcal{N}|^2)$ constraints.

DATA AVAILABILITY

The datasets generated during the current study that are used for the plots are available at <https://doi.org/10.4121/14914806.v1>. The data representing the SURFnet core network is available in the Github repository²⁴.

CODE AVAILABILITY

All the code used for generating the results can be found in the Github repository²⁴.

Received: 3 June 2020; Accepted: 20 October 2021;

Published online: 20 January 2022

REFERENCES

- Van Meter, R. Quantum networking (John Wiley & Sons, 2014).
- Lloyd, S. et al. Infrastructure for the quantum internet. *Comput. Commun. Rev.* **34**, 9–20 (2004).
- Kimble, H. J. The quantum internet. *Nature* **453**, 1023 (2008).
- Wehner, S., Elkouss, D. & Hanson, R. Quantum internet: a vision for the road ahead. *Science* **362**, 1–9 (2018).
- Pirandola, S. & Braunstein, S. L. Physics: unite to build a quantum internet. *Nat. N.* **532**, 169 (2016).
- Pirandola, S. et al. Advances in quantum cryptography. *Adv. Opt. Photonics* **12**, 1012–1236 (2020).
- Bennett, C. H. & Brassard, G. Quantum cryptography: public key distribution and coin tossing. *Theor. Comput. Sci.* **560**, 7–11 (2014).
- Ekert, A. K. Quantum cryptography based on bell theorem. *Phys. Rev. Lett.* **67**, 661 (1991).
- Broadbent, A. & Schaffner, C. Quantum cryptography beyond quantum key distribution. *Des. Codes Cryptogr.* **78**, 351–382 (2016).
- Broadbent, A., Fitzsimons, J. & Kashefi, E. Universal blind quantum computation. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, 517–526 (2009).
- Ben-Or, M. & Hassidim, A. Fast quantum byzantine agreement. In *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing, STOC '05*, 481–485 (Association for Computing Machinery, Baltimore, MD, USA, 2005).
- Kómár, P. et al. A quantum network of clocks. *Nat. Phys.* **10**, 582–587 (2014).
- Bhaskar, M. K. et al. Experimental demonstration of memory-enhanced quantum communication. *Nature* **580**, 60–64 (2020).
- Stephenson, L. et al. High-rate, high-fidelity entanglement of qubits across an elementary quantum network. *Phys. Rev. Lett.* **124**, 110501 (2020).
- Rozpędek, F. et al. Near-term quantum-repeater experiments with nitrogen-vacancy centers: Overcoming the limitations of direct transmission. *Phys. Rev. A* **99**, 052330 (2019).
- Humphreys, P. C. et al. Deterministic delivery of remote entanglement on a quantum network. *Nature* **558**, 268–273 (2018).
- Yu, Y. et al. Entanglement of two quantum memories via fibres over dozens of kilometres. *Nature* **578**, 240–245 (2020).
- Pirandola, S. End-to-end capacities of a quantum communication network. *Commun. Phys.* **2**, 1–10 (2019).
- Munro, W. J., Azuma, K., Tamaki, K. & Nemoto, K. Inside quantum repeaters. *IEEE J. Sel. Top. Quantum Electron.* **21**, 78–90 (2015).
- Lougee-Heimer, R. The common optimization interface for operations research: Promoting open-source software in the operations research community. *IBM J. Res. Dev.* **47**, 57–66 (2003).
- Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual (2021). <https://www.gurobi.com>.
- IBM. IBM ILOG CPLEX 12.9 User's Manual (2019). <http://www.cplex.com>.
- Daskin, M. S. Network and discrete location: models, algorithms, and applications (John Wiley & Sons, 2011).
- Rabbie, J. & Avis, G. Repalloc (2020). <https://github.com/jrabbie/RepAlloc>.
- Sinclair, N. et al. Spectral multiplexing for scalable quantum photonics using an atomic frequency comb quantum memory and feed-forward control. *Phys. Rev. Lett.* **113**, 053603 (2014).
- Jones, D., Tamiz, M. et al. Practical goal programming, vol. 141 (Springer, 2010).
- Penrose, M. et al. Random geometric graphs, vol. 5 (Oxford university press, 2003).
- Hagberg, A. A., Schult, D. A. & Swart, P. J. Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T. & Millman, J. (eds.) Proceedings of the 7th Python in Science Conference, 11–15 (Pasadena, CA USA, 2008).
- Barber, C. B., Dobkin, D. P. & Huhdanpaa, H. The quickhull algorithm for convex hulls. *ACM Trans. Math. Softw.* **22**, 469–483 (1996).
- Virtanen, P. et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**, 261–272 (2020).

ACKNOWLEDGEMENTS

We would like to thank SURFnet for sharing their data regarding the network topology. This publication was supported by the QIA-project that has received funding from the European Union's Horizon 2020 research and innovation program under grant Agreement No. 820445. This work was supported by an ERC Starting grant and NWO Zwaartekracht QSC.

AUTHOR CONTRIBUTIONS

This work is based on the master thesis of J.R. which was devised by S.W. and supervised by S.W. and G.A.; K.C. introduced some key conceptual and proof ideas. J.R. proposed the use of linear programming and implemented both formulations with CPLEX and G.A. wrote most of the code for the simulations. All authors contributed to the manuscript.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41534-021-00501-3>.

Correspondence and requests for materials should be addressed to Stephanie Wehner.

Reprints and permission information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the

article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022