On Designing A Quantum Network Topology: Model, Methodology, and Algorithm

Yuhang Gan*, Ruilin Zhou*, Shiyi Ling, Yi Liu, and Chen Qian
University of California, Santa Cruz
{ygan11, rzhou39, sling9, yliu634, cqian12}@ucsc.edu
*Authors contributed equally to this research.

***Abstract*—Quantum networking has been attracting growing attention in both academia and industry during recent years. Although many routing protocols have been proposed to optimize the network layer of quantum networks, another key network-layer problem is under-explored: the topology design problem. Existing work is limited to using structure topologies such as a ring or grid, using randomly generated topologies, or using existing Internet backbone infrastructure. Whether these choices are desired for quantum networks is not fully explained. We propose the quantum network topology design problem, for which a set of end users are given and we need to compute a set of repeaters and their locations and connectivities while ensuring the network performance, such as throughput, is optimized under the same infrastructure cost (e.g., number of repeaters). We present the methodology of designing topologies based on the unique characteristics of entanglement distribution networks and propose an algorithm called TriPlace to compute a topology for a given set of end users in different locations of a geographic region. The simulation results show that TriPlace incurs a lower infrastructure cost than all existing solutions while achieving similar throughput performance as the best of these solutions. We believe this work will start a new research problem for quantum networks and attract more future work.**

***Index Terms*—Quantum Networks, Network Topology Design**

I. Introduction

Quantum communication networks enable new applications that are not easy to build by conventional computer networks, including quantum key distribution (QKD) [1], [2], distributed quantum computing [3], [4], and blind quantum computing [5], [6]. Entanglement distribution networks [7]–[11] is one powerful and attractive type of quantum networks, because it allows end-to-end communication through a network of untrusted quantum repeaters using entanglement swapping.

Although we have observed a great number of studies focusing on the network layer of entanglement distribution networks during recent years – many of them are about the entanglement routing problem – we realize an under-explored yet important problem: here is little discussion on how the topology of entanglement distribution networks should be designed and organized. We summarize an (incomplete) list of recent works about quantum networks and the network topologies used in their analysis and experiments in Table I. Some assume structured topologies such as a ring [8], [12] or grid [9], [12]–[14]. Many use random topologies like the ones [10], [11], [15]–[19] generated by the Waxman model [20], where a link is placed between two nodes randomly based on a probability related to their distance. Some work uses the Internet backbone infrastructure [21]–[23], but no evidence shows quantum network nodes should follow the Internet backbone topology and they use different fibers. Quantum nodes only require normal Internet connections. There is little discussion on why they choose these topologies and whether these topologies are indeed preferred for quantum networks.

TABLE I: Topologies used in recent quantum network research

Topology Type	Related research
Random graph/Waxman	[10], [11], [15]–[18]
Square grid/lattice, star, ring, or chain	[8], [9], [12]–[14], [19], [24]–[27]
Internet backbone infrastructure	[21]–[23]

We argue that designing a network topology for quantum networks is a non-trial problem and worth careful studies, mainly due to the unique features existing in quantum networks. In a quantum network using entanglement distribution [7]–[11], end-to-end communication is achieved by a path of repeaters between the source and the destination. Each pair of neighbors on the path establish an entanglement pair, and all repeaters perform an operation called Bell state measurement (BSM) [28] (also called swapping) to generate an end-to-end (e2e) long-distance entanglement between the source and destination. Then a qubit can be transmitted between them. The success rate of generating a successful entanglement between two repeaters, such as through high-quality single-mode or ultra-low loss optical fiber [3], [29], [30], decreases exponentially with the distance [9], [10]. In addition, each BSM operation also succeeds with a probability related to the repeater hardware. Therefore the success rate of building one e2e entanglement can be modeled as a product of multiple probabilities including those for entanglement generation and BSM operations. Hence to optimize the e2e entanglement success rate, the number of repeaters on the path should neither be too small (which results in long links between neighbors and hence a low e2e success rate) nor too large (which results in more BSM operations and hence a low e2e success rate).

To illustrate our argument point, we use an example of putting repeaters between two end users with distance D. If we deploy n repeaters on the path evenly with links have the same length, which can achieve a higher end-to-end success rate than uneven deployment [31], then the length of each

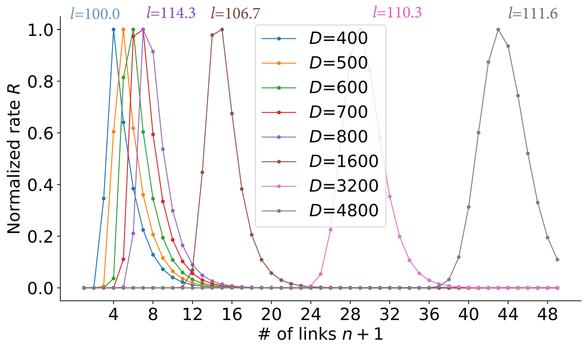

Fig. 1: e2e success rate R vs. # of links on a path

link is $l = \frac{D}{n+1}$ and the e2e entanglement success rate can be modeled as [21], [32]:

$$R = \frac{c_f}{l} q^n \left[1 - \left(1 - qe^{-l/L_{\mathrm{a}}}\right)^m\right]^{n+1} \tag{1}$$

where c_f is the speed of light in the fiber, q is the success probability of BSM, and L_{a} is the fiber attenuation length, and m is the entanglement-distribution attempts times per time slot on each link. Fig. 1 shows the numerical value of R for a different set of D (in km) by varying n and other reasonable parameters for optical fiber. We find that for each value of D, R can only achieve its optimal values for a very small range of n (and corresponding l). Outside the optimal value of n and l, R decreases significantly due to the nature of quantum networks. In addition, we display the value of l (in km) for each optimal R and find that for the same set of hardware parameters, the optimal link length l is similar across different e2e distances. Finding an optimal network topology for geographically distributed users is an even more challenging problem, which needs to consider many factors including the optimal link length, locations of repeaters, cost of repeaters, and multiple paths between each pair of e2e nodes.

In this work, we introduce the problem of *quantum network topology design*, which has never been studied in prior work. The problem is described as follows. Given a set of end users (each of which can represent a large population) in a 2-dimensional plane, the problem seeks for a placement of repeater nodes and their connection graph, such that 1) each link is close to its optimal length based on its hardware parameters; 2) the cost of repeaters is optimized (i.e., the number of repeater nodes should be limited), and 3) the expected throughput and fidelity, two metrics that are commonly used in existing work, are optimized compared to other topologies with the same cost of repeaters. There is no prior work studying the same problem. The closest study to this work is the methods to select a subset of nodes for repeaters from *existing high-speed Internet infrastructure* [21]–[23]. Our work provides a more generalized solution to allow the repeater nodes to be in any location and any connection topology, which matches practical quantum networks because quantum network hardware can be independent of the high-speed Internet infrastructure. **We believe the design of quantum network topology is a crucial step and impossible to bypass in developing future quantum networks, despite how detailed quantum network technology evolves.**

To provide a solution to this problem, we propose the topology design algorithm, called TriPlace. The main idea of TriPlace includes the following steps. 1) It divides the plane containing end users into a grid network of square lattices formed by a set of repeater candidates. 2) Then TriPlace constructs a Delaunay Triangulation (DT) [33] whose property is known to avoid extremely long edges and provide rich path diversity among each pair of arbitrary nodes while maintaining a limited number of node degrees (and hence a limited number of links). 3) To reduce the cost of the network, TriPlace further uses the k-means clustering algorithm [34], to reduce the number of repeater nodes by merging geographically close repeater candidates to avoid unnecessary repeaters that might introduce short links and more hops. A genetic algorithm [35] is also used to adjust the repeater locations such as all link lengths have small variations and are close to the optimal length.

We summarize the main contributions of this work:

- We propose the problem of quantum network topology design and the corresponding network model.
- We develop the methodology to find an optimal quantum network topology with limited cost, including finding the optimal link length, finding the locations of repeaters, and connecting certain pairs of repeaters.
- We present an algorithm for quantum network topology design, called TriPlace, for constructing large-scale quantum backbone repeater networks, which uses only around 25% of repeaters but achieves similar throughput compared to a very recent work that places repeaters using the Internet infrastructure [23].
- We conduct comprehensive simulation experiments by running the existing routing algorithms on different network topologies including TriPlace. The results demonstrate that TriPlace has significant advantages in the cost-performance tradeoff among existing solutions. **The simulation code will be open to the public to motivate more computer networking researchers to work on this problem, upon the acceptance of this paper.**

The rest of the paper is organized as follows. In Section II, we discuss the quantum network model, the background, and our observations that motivate this work. In section III, we present our topology design algorithm TriPlace. We show simulation results to demonstrate the advantages of TriPlace in Section IV. Related work will be discussed in Section V. We will summarize this study in Section VI.

II. Network Model and Observation

In this section, we briefly review some existing architectures and communication models of quantum networks. By analyzing the characteristics of the hardware and corresponding software of existing quantum networks, we summarize the key points to pay attention to in designing a quantum repeater network.

A. Quantum Network Model

The quantum network discussed in this work includes the following hardware/software components.

Quantum node. A quantum node refers to a node with quantum computation and repeater functions. They can be categorized into end nodes, each of which can be a quantum computer, a quantum computing cluster, or a quantum sensing device deployed in a specific location (e.g., a university, a laboratory, a financial institution) and equipped with quantum functional units, and repeater nodes, which are responsible for generating the required entanglement pairs and performing BSM operations. Most previous work assumes that both types of nodes are given and then designs certain network protocols. This work considers that end nodes are given knowledge, while repeater nodes can be freely deployed to form a particular topology.

Links between two nodes. Two nodes – each side can be an end node or repeater – can be connected by a quantum link whose hardware is usually a high-quality optical fiber. Two nodes with a quantum link can generate an entanglement.

Hardware of a repeater node. A repeater node requires the necessary hardware to accomplish the repeater task. Quantum networking is a rapidly evolving field with several different hardware architectures for distributing entangled pairs between neighboring nodes that are being developed in parallel. Examples include directly exciting quantum memories to emit photons and setting up a measurement device (usually a Bell State Analyzer for 2-qubit measurement) between the two nodes to establish an entanglement [36], or setting up a separate source of entangled photons between the two nodes and making measurements inside the repeater node to realize entanglement between the quantum memories [37]. However, in general, we need to deploy necessary photon sources and measurement devices at the repeater nodes to realize the entanglement distribution among nodes. In addition, quantum frequency conversion (QFC) is needed to convert photons to a specific wavelength (e.g., 1550nm) before they enter the optical fiber, which requires a specialized quantum frequency converter chip [38]. In this paper, we do not assume a specific hardware but each repeater will incur non-trivial financial cost.

Costs required to build a quantum repeater network. Counting only the hardware overhead, building a quantum repeater network requires the following parts of expenditure: Fixed expenses for building every repeater node, and usually the most expensive unit expenses, such as the necessary buildings and supporting facilities, e.g., electrical systems, and refrigerators needed for the superconducting platform. Equipment overhead within the repeater node, e.g. quantum memories, measurement devices, and QFC devices. These are known as hardware resources and different repeater nodes may have different amounts of hardware resources units. Finally, the cost of inter-node links, e.g., the cost of purchasing and laying optical fiber, is directly related to the length of the link, but is much smaller than that of repeaters by orders of magnitude.

Transmission protocol and model For pairs of non-adjacent end nodes that need to be connected through entanglement distribution networks, the routing protocol first needs to select the path from the source to the destination and establish entanglements between each pair of nodes. After this, inside each quantum repeater, a BSM operation is performed. At this point, the corresponding qubits are consumed, but logically, an internal link is placed inside the quantum repeater to connect the external elementary links into longer links. If this process is successfully performed inside each selected quantum repeater, we end up with an end-to-end entanglement distribution from the source to the destination [10], [11], [17].

B. Observation and design preference.

As mentioned in Section I, the success rate of photon transmission decreases exponentially with the transmission distance. So, it is vital to ensure that the link between any two nodes is not too long to provide a reasonable single-hop success rate when setting up the location of the repeater nodes. A sufficient number of repeater nodes can ensure that the distance of any single hop is not too long through reasonable placement. However, too many repeaters can also hurt the performance of the network. To obtain a final end-to-end entanglement distribution from source to destination, a BSM operation is required at each repeater node to "connect" entanglement between neighboring nodes. However, this operation is probabilistic. If it fails, the entanglement links on the path will also be consumed and cause the rest of this link to fail. According to Equation 1, if a path from the source to the destination has too many hops, the final connection of the establishment of the success rate and the final entanglement of the fidelity will be poor. This can be mitigated by increasing the number of edges between nodes, i.e., the degree of the nodes, to create more possible short paths. Still, redundant links imply additional network construction costs.

Besides, there is a "performance matching" problem between the links of a quantum network [39]. Considering a routing path, since the single-hop success rate of a link is sensitive to its length, if the link lengths of different hops differ greatly, there will be "longer links are more likely to fail than shorter links", resulting in shorter links not being used at the end, and performance mismatch between links, resulting in wasted resources and loss of network performance. While no design can guarantee that all elementary links on a path will succeed or fail at the same time in every attempt, being able to guarantee in a statistical sense that the success rate of each link on a path is roughly the same can minimize the performance mismatch problem. And given that the success rate of elementary link establishment varies drastically with distance, intuitively, we need the lengths of the links in the network to be as similar as possible. Finally, based on the results of Eq. 1 and Fig. 1, we realize that there exists a very narrow range of the optimal single-hop transmission distance for a given physical parameter of the transmission medium. To maximize the network transmission rate, a good quantum network topology should always make the length of each link as long as possible in the range of optimal single-hop transmission lengths.

Therefore, we set the following preferences for the characteristics of quantum networks when designing our repeater network topology design algorithm, TriPlace:

- Due to the efficiency constraints of measurement operations, we prefer to set sufficient and proper connections between nodes to ensure that the hop count of the routing path is within a reasonable range.
- Since there is a narrow range of the optimal link length for a given hardware, we would link the links in the network to be close to the optimal link length.
- While keeping the above properties, the number of repeater nodes is minimized to save cost. We will also evaluate the total link length (fiber cost) in the network but it is less prioritized than the repeater cost.

III. Design of TriPlace

In this section, we show how TriPlace determines the number and location of repeater nodes and the links between nodes.

A. Overview

The hardness of the quantum network topology design problem. It is well-known that the Steiner tree problem, which is to find a subgraph to connect all nodes with given coordinates in a network such that the total link cost in Euclidean distance is minimized, is NP-hard [40]. The quantum network topology design is even more challenging because we do not have the coordinates of the nodes. Hence TriPlace is a heuristic that works as follows.

In a geographic region, the end nodes (users) and their locations are given as shown in Fig. 2. We want to place repeater nodes to connect the end nodes to form the network. TriPlace begins by determining the number and location of repeater nodes. It is intuitive to think that we can select some points from the plane as candidates and pick the desired and appropriate ones to build the initial topology, which can then be optimized at a finer granularity. Thus, TriPlace first discretizes the geographic plane using a grid network such that all intersections in the grid network form a repeater nodes candidates set C, and removes from it all points not contained in the concave hull formed by end nodes as shown in Fig. 3. Then, we build a 2D Delaunay Triangulation (DT) [33] on all end nodes and repeater candidates as shown in Fig. 4 to form connections between nodes other than the square lattice network. A DT maximizes the minimum angle of the triangle in this triangulation and avoids "skinny" triangles, thus avoiding our initial topology having some excessively long edges. This provides a good start for the subsequent topology optimization.

After obtaining the initial topology, we start optimizing the topology. As mentioned before, we want to reduce the construction cost of the network without hurting its performance. The cost of the network mainly includes the cost of the repeater nodes and their corresponding devices as well

Algorithm 1 TriPlace

Input: $P, E = \{e_i\}$
// P: The geographical plane.
// E: Set of all end nodes e_i on the geographical plane.
Output: G_{final}
// G_{final}: Final required topology.

Generate a grid graph G_{grid} for P;
$R \leftarrow$ all nodes in G_{grid};
$L_{grid} \leftarrow$ all edges in G_{grid};
Use Delaunay Triangulation to generate edges L_{DT} for all nodes $E \cup R$;
$L_{tri} \leftarrow L_{grid} \cup L_{DT}$;
$V_{tri} \leftarrow E \cup R$;
$G_{tri} = \langle V_{tri}, L_{tri} \rangle$;
while no illegal edges **do**
    Use k-means clustering to merge repeater nodes in G_{tri} from k to $k-m$.
    Use Genetic Algorithm to optimize the positions of repeater nodes in G_{tri} finely.
end while
$G_{final} = G_{tri}$;
return G_{final}

as the cost of the links between the repeater nodes, which are related to the length of every link. The cost of building repeater nodes is the most expensive part and is much larger than the remaining parts. Therefore, we need to reduce the number of repeater nodes to minimize the cost. Considering that geographically neighboring or close repeater nodes usually carry similar traffic, an intuitive idea is that we can merge these candidate nodes. Thus, after obtaining the initial topology, we iteratively proceed the steps as follows: 1) cluster all the repeater candidates using the k-means clustering algorithm with fewer clusters than the repeater candidates. We merge the candidates in these classes and use the clustering center of each class as a new repeater candidate node to replace those merged points. Since the clustering result of k-means is a Voronoi diagram for the plane and the Voronoi diagram is actually the dual graph of DT, the properties of DT are preserved. We set the upper limit of the optimal link length range obtained from the physical parameters of the links as the maximum legal distance of the links. And we set the link length constraint of a link to be the optimal link length computed from Eq. 1 for each hardware setting. Then, we use a genetic algorithm (GA) to adjust the positions of these repeater nodes to ensure that all links meet the link length constraint while minimizing the variance of all link lengths so that the link lengths are as close as possible. The iterative process is repeated as above until meeting the length constraint is no longer possible as shown in Fig. 5. Alg. 1 shows the pseudocode of TriPlace.

B. TriPlace algrotihm

1) Determining the initial solution by discretizing the plane using a grid network: For a given region where end nodes

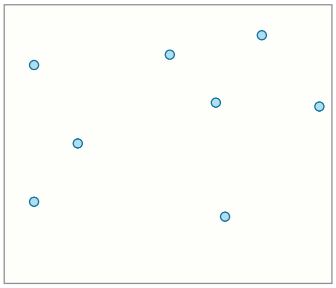

Fig. 2: Some end nodes are located on a geographical plane.

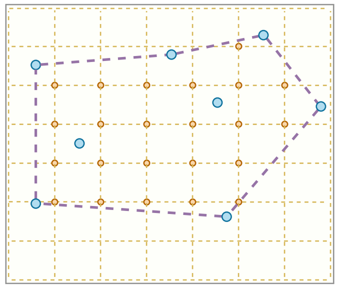

Fig. 3: Init plane using a grid network.

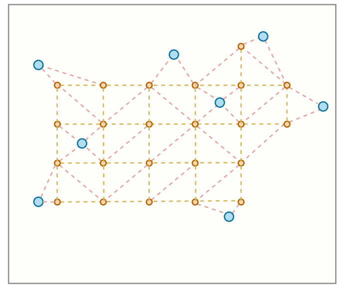

Fig. 4: Add extra edges to the grid network.

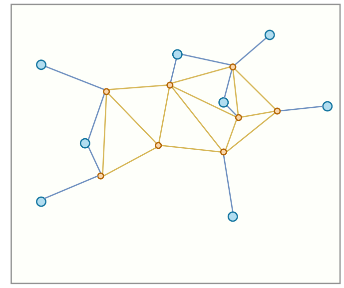

Fig. 5: Final topology after running TriPlace

exist, we first discretize the region using a grid network, and all the intersection nodes in the grid network form a repeater candidate point set C. We then build a concave hull of all end nodes and remove all repeater nodes outside the concave hull as shown in Fig. 3. But how to determine the edge length of each square lattice? Fine granularity means finer approximations, which results in more candidate nodes, increasing the algorithm's complexity, which grows by $O(n^2)$ with the size of the geographical plane. Coarse granularity reduces the size of the set of candidate nodes, making the search space smaller, but the result is more likely to be suboptimal. Here, we propose to set the square lattice's edge length according to the optimal link length that can optimize the e2e success rate as in Equation 1. More specifically, we set the edge length of the square lattice to be $\varepsilon \in (0, 1)$ of the maximum length of the fiber, and after experimentation, we choose ε from range $[\frac{1}{3}, \frac{2}{3}]$. Value in this range can provide sufficient space for the subsequent optimization of candidate node positions.

2) Add extra edges between nodes using Delaunay Triangulation: After providing basic connectivity between repeater nodes using the grid network, we need to connect the end nodes to the appropriate repeater nodes. Also, the basic connectivity of the grid network is about 4, which is not enough to ensure the robustness of the network considering that we need to merge repeater nodes later, which may also merge edges. Some additional links need to be added. Therefore, based on the grid network, we choose to use 2D Delaunay Triangulation (DT) for all nodes as shown in Fig. 4. DT is a method for forming links between discrete points [33]. More specifically, for a given set of points distributed in the plane, all points are connected into a triangular mesh such that the circumcircle of each triangle does not contain a fourth node as illustrated in Fig. 6. DT has the following nice features that is useful for quantum network topology design:

- A DT maximizes the minimum angle and avoids skinny triangles. This helps reduce the difference in the lengths of the three sides of any triangle in the network. Even when dealing with unevenly distributed data points, Delaunay triangulation can produce a relatively uniform and stable triangular mesh.
- For a 2D DT, the average degree is $6 - \frac{12}{n}$ for n nodes, which is strictly less than but close to 6. Nodes have similar degrees.
- A DT is proved to be a geometric spanner: the sum of the link lengths on the path between two arbitrary nodes on the DT is known to be close to the Euclidean distance between them and bounded by a factor of 1.998 [41].
- A DT can provide path diversity between each pair of nodes while maintaining low node degrees, hence fewer links.

The first feature that maximizes the minimum angle reduces the possibility of having excessively long edges, which will significantly reduce the e2e success rate due to the exponential decrease of the entanglement generation rate. In addition, this property also ensures that the lengths of the edges in the graph will be relatively similar, which has been demonstrated as the condition to achieve the optimal e2e success rate [31].

The degrees of repeater nodes in the Delaunay Triangulation being no more than 6 and similar ensures that the routing load on most repeater nodes is balanced. No node will become a significant "hot spot".

The geometric spanner feature along with the degree feature provides a nice guarantee of the routing path on the topology: the sum of link lengths on the path connecting two end nodes a and b will not be significantly higher than the Euclidean distance between them. Thus the link lengths to connect two nodes can be smaller than those on non-spanner topologies.

3) Iterative merge repeater nodes and fine-grained optimize nodes locations.: After obtaining an appropriate grid network with all end nodes connected to their nearby repeater candidates via 2D Delaunay Triangulation, we need to further optimize the network cost by removing some candidates without significantly hurt the quality of paths among the end nodes. We propose using the k-means clustering algorithm [34] to

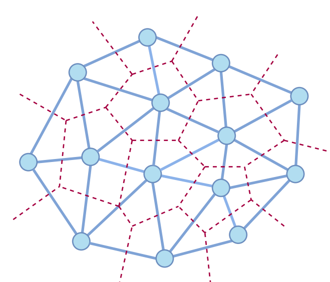

Fig. 6: Delaunay Triangulation (Blue lines) and its dual Voronoi diagram (Red lines).

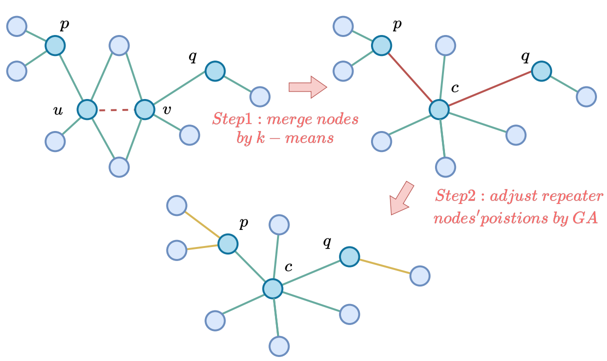

Fig. 7: Merge nodes using k-means and finely optimize nodes locations using GA

accomplish this optimization process. More specifically, to eliminate redundant repeater candidates and reduce the number of repeater nodes, we use the k-means algorithm for merging repeater node candidates. Subsequently, using the genetic algorithm (GA) [35], the position of each node is finely adjusted to minimize the variance of the overall link lengths and adjust potential links that may exceed the maximum length limit due to merging.

Why k-means? k-means is a widely used method of vector quantization, employed in various tasks such as unsupervised machine learning clustering, and it is supported by a solid theoretical foundation. k-means aims to divide n observations into k clusters, where each observation belongs to the cluster whose center (mean or centroid) is the nearest one to the observation among all cluster centers. This approach effectively partitions the data space into a Voronoi diagram [33] and each cluster is a Voronoi cell. A Voronoi diagram divides a plane into regions based on proximity to a set of defined points (also known as seeds, sites, or generators). Each point or seed has an associated region, termed a Voronoi cell, which encompasses all the points on the plane closer to that seed than any other. The Voronoi diagram related to a set of points is the dual of the Delaunay triangulation for that set [33] . Additionally, k-means clustering minimizes within-cluster variances (squared Euclidean distances). When we constructed the initial topology in the previous step, we used the DT graph to determine the connections among the end and repeater nodes. Naturally, we do not want to lose the good properties of DT when we further optimize the topology. k-means maintains the properties of DT succinctly and indirectly by always making use of the dual of the Voronoi diagram when clustering and merging repeater candidates.

A specific merging process example is shown in Fig. 7. In a certain round of iteration, there remains a set of k repeater node candidates and we want to reduce the number of repeater node candidates further without making any link length longer than the maximum length limit. Step 1 uses the k-means algorithm to cluster these k nodes by taking the geographic locations of these nodes as the input and setting the number of classes to $k - m$ for $(m \geq 1)$.

In Fig. 7, we make $m = 1$, i.e., we want to find two nodes u, v, whose geographic locations are closer than any other pairs, implying that the traffic load will also be more similar and mutually redundant. Therefore, these two nodes can be the right candidates to merge. Take the clustering center c of these two nodes as the new node, as shown in Step 1, and all other nodes are the clustering centers of themselves, whose locations remain unchanged. Since k-means guarantees that the final result divides the plane into a Voronoi diagram, and each node is a vertex of its dual graph: the DT graph. The new node c (the cluster center of u and v) obtained after merging also becomes a new node in the topology without destroying the DT properties. At this point, the properties of the DT of the whole network graph are maintained. It is easy to obtain that the description above is also an informal implementation of the proof that the DT property is preserved using induction. For the case $m > 1$, we can think of repeating the process as above for m times and each iteration reduces the size by 1. Large m may cause the best solution to be skipped during optimization, so we dynamically adjust m when running the algorithm. A large m can speed up the algorithm in the early optimization stage. Near the end of the algorithm's run, decreasing the value of m allows the algorithm to obtain the best results. **Note the result of the topology design has very long-term impacts hence the computation time of these steps is not sensitive. For this reason, one might always make $m = 1$.** We use adaptive values of m because we need to control the time for running experiments.

Simply utilizing the new node instead of the original candidates may create new problems. The links connected to the two nodes before the merging are now naturally connected to the new node c. Since c is at a different location than u, v, the lengths of these links change and may exceed the maximum length limit, and the variance of the links' length is not minimal and can be further optimized. As shown in Fig. 7, the links l_{pu}, l_{qv} are connected to the nodes u, v before merging, and after merging, these two edges are connected to the node c, at this time, the lengths of them are forced to be longer, and therefore, the maximum length limit of the links may be exceeded. To address these issues, we need to optimize the position of the current node at a finer granularity to adjust links that violate the length limit, as well as further optimize the overall length of the link to reduce the variance of links' length. We use the genetic algorithm (GA) in Step 2 to achieve this while ensuring that none of the links exceed the maximum length limit. In Fig. 7, by adjusting the positions of nodes p, q, we shortened the length of links l_{pc} and l_{qc} and also minimized the link length's variance.

By completing "merge repeater nodes using k-means clustering algorithm" and "optimize links using GA with fine-grained adjustment of node positions" in each round of iteration until we cannot reduce the number of repeater nodes further without violating the link length constraint, we obtain the final network topology as shown in Fig. 5.

IV. Performance Evaluation

A. Implementation

To implement all design components of the quantum network topology design problem, we build a custom Python simulator but we completely re-implement all functions of a publicly available Kotlin-based quantum network simulator [10]. To measure the practical throughput of different topologies, we need to run the routing algorithm on them. We use the widely accepted Q-CAST routing algorithm as a unified runtime algorithm on different topologies to compute the paths [10]. The implementation of Q-CAST is exactly the same as the original program in Kotlin. **All code used in the following evaluations will be made publicly available.**

B. Methodology

We evaluate the network properties of the designed repeater network, the practical entanglement distribution rate, the scalability, and the network resilience to failures for different users in the network.

Evaluation metrics.

- **The number of repeater nodes required to build the network:** The cost of building a quantum network is directly related to and dominated by the number of repeater nodes.
- **Link lengths and distribution:** The success rate of single-hop entanglement in a quantum network decays exponentially with distance, so the length of the links is directly related to the performance of the network. In addition, we evaluate the variation of all links in a network because one design goal is to make link lengths close. Besides, we measure the total link length of each topology to observe the potential cost of fibers when constructing the network.
- **Hop count:** We measure the average hop count of paths in the network. The success rate and fidelity of end-to-end entanglement also decrease exponentially with the number of swapping operations, i.e., the number of repeater nodes passed through, and the hop count of a path directly reflects the rate and quality of the final end-to-end entanglement distribution. In analyzing the topology's resiliency to failure, we measure how the hop count varies with the ratio of failed links in the network.
- **Network throughput:** Following the definition of throughput in many existing studies of quantum networks [10], [11], [15], the throughput of a network is the number of entanglement pairs established for given source-destination pairs in one time slot. We measure how the average network throughput varies with the hardware parameters in the network (success rate of link and swapping operations) and the network size (number of end nodes) in the reference setting. To analyze the topology's resiliency, we evaluate how network throughput varies with the ratio of failed links.

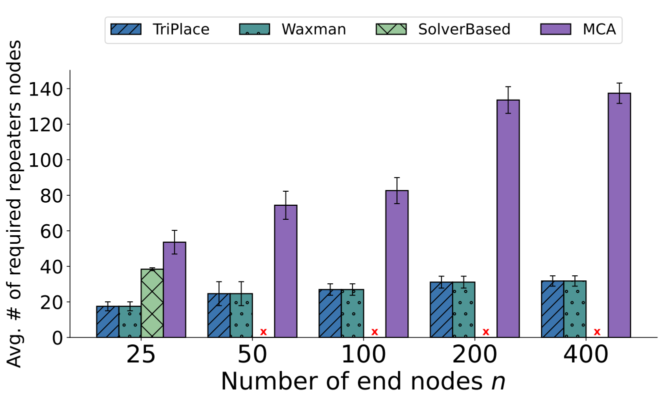

Fig. 8: # of required repeater nodes vs. # of end nodes n

Some parameters are configurable to evaluate the performance of the algorithm in different situations:

- The success probability q of the entanglement swapping (BSM) operation within a repeater node.
- The success probability p of a single-hop link in a single time slot. The success probability of the establishment of an entanglement on a single-hop link can be modeled as $p = e^{-\alpha L}$ [10], [11], [15], where α is a hardware-dependent parameter. Following previous work [10], [15], we set the success probability $p = \{0.3, 0.5, 0.7, 0.9\}$ when the link length is the optimal length calculated in Equation 1. Then we derive the corresponding α and calculate the probabilities of other links using α and their lengths. Note that when $p = 0.3$, it means the probability of generating a successful entanglement on the link in the whole time slot is 0.3, which includes multiple attempts until the success. It is not the success probability for one single attempt.
- The number of user end nodes is n, varying within $\{25, 50, 100, 200, 400\}$. These end nodes are randomly distributed on the plane.

Same with previous work [10], we assume that the number of qubits owned by each node is randomly chosen in the range $[10, 14]$. To demonstrate the impact of different variables on the algorithm, we set $n = 100, m = 5, p = 0.5, q = 0.5$ as a reference setting. For each setting of (n, m, p, q), 10 different sets of end node locations are generated, and 1000 independent time slots are simulated while running the routing algorithm, generating 10 different sets of source-destination pairs each time.

Methods to compare.

- Solver-based algorithm [32]: Rabble $et\ al.$ propose a quantum network construction method by selecting repeater nodes from a pre-existing Internet backbone topology using the integer linear programming (ILP) solver. Since it cannot be run directly on a plane without repeater candidates, we run the algorithm by first following the first step of TriPlace's discretization of the plane to obtain repeater node candidates and then apply the solver. The computational complexity of the solver-based algorithm rises when the network size increases. In our experiments, when we have more than 50 end nodes, the results cannot be computed in two days.

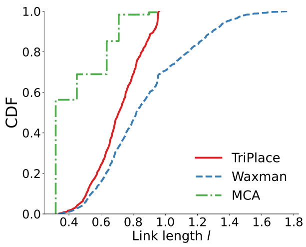

Fig. 9: Normalized edge length CDF.

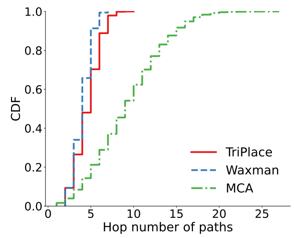

Fig. 10: Paths hop number CDF

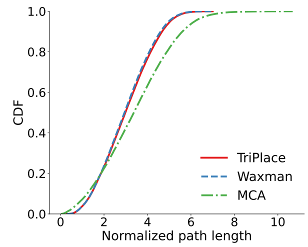

Fig. 11: Normalized path length CDF.

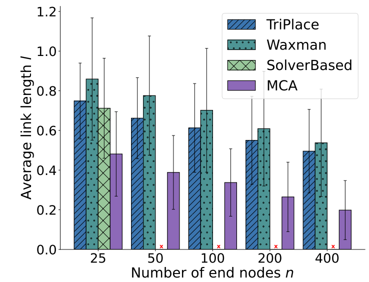

Fig. 12: Avg. edge length vs. # of end nodes n.

- MCA [23]: MCA is a heuristic that also selects repeater nodes from a pre-existing network infrastructure, which can run on a larger network than the solver-based algorithm. Since there is no publicly available code, we implemented it in Python based on the description in the paper. Similarly, we use grid networks to provide candidate repeater nodes and edges.
- Waxman model based random graph [20]: Most previous work about quantum routing algorithms use the Waxman model to generate random network topologies [10], [11], [15]. The Waxman algorithm generates edges probabilistically based on the distances between the nodes. We take as input the repeater nodes obtained from TriPlace, run the Waxman model algorithm to generate edges to get the topology, and control the degree of the generated graph to be roughly 6 to align with TriPlace.

C. Evaluation results

Topology Property. Fig. 8 shows the number of repeaters used by each topology design algorithm varied with the number of end nodes. It can be seen that compared to other topology design algorithms, TriPlace needs the least number of repeater nodes, implying the least infrastructure cost. The Waxman model uses the same number of nodes for topology generation as it uses the repeater nodes obtained from running TriPlace but we will see its performance is significantly worse than TriPlace. The solver-based algorithm gives results only when the number of end nodes is 25 and uses more repeater nodes than TriPlace. The number of candidates for the solver-based algorithm is 64. MCA, as a heuristic, can compute topologies for more end nodes but it needs significantly more repeaters (3x to 4x) compared to TriPlace.

Considering one target of our design is to make the link lengths as close as possible to avoid extremely long links. Fig. 9 shows the CDF of the normalized link lengths of the three algorithms in the reference setting. It can be seen that MCA has many short links because it includes much more repeaters than others. Waxman generates links between two points probabilistically, so the length of the links varies the most, leading to the largest variations among link lengths. And extremely long links do exist in Waxman. Besides, about 30% of links of topologies generated by Waxman exceed the maximal link length, which can easily become the "bottleneck" of paths in the network. TriPlace achieves the best link length balance, where the overall length variation between links is minimized, and the link lengths are close and do not exceed the maximum length needed. This maximizes the throughput performance in the network while maintaining small number of repeaters. More than 80% of links of TriPlace's topologies longer than $0.6 \times l_{max}$. In addition to the effect of Waxman's extra-long edges, TriPlace also has a longer average link length than MCA, as shown in Fig. 12.

In addition, both the success rate and fidelity of end-to-end entanglement distribution decrease exponentially with the number of BSM operations, i.e., the number of repeater nodes on a path. We expect the number of hops of paths in the network will not be excessive. Fig. 10 shows the CDF of the number of hops of the paths of the topology obtained by the three algorithms. Fig. 11 shows the CDF of the length of the paths. it can be seen that TriPlace similarly achieves the best balance. Waxman allows the presence of extra-long edges while minimizing the number of path hops in the network. MCA, on the other hand, has the largest number of both path hops and path lengths, and we speculate that this may be because MCA chooses most of the repeater nodes and constructs a minimum spanning tree (MST) [42]. As mentioned earlier, the network's diameter is directly related to the degree of the network, and the tree structure results in a smaller degree of the network. TriPlace guarantees the link lengths while obtaining path lengths and hop counts close to those generated by the Waxman model.

We then discuss the relationship between maximum link length l_{max} and the plane size (represented by the ratio m of the side length of the geographical plan to l_{max}). Fig. 16 shows the relationship between the number of repeaters required by TriPlace and MCA and m. It can be seen that for $m \leq 10$, the number of repeaters needed by TriPlace is much smaller than that of MCA, but it grows rapidly. This is because TriPlace iteratively reduces the number of repeaters to ensure the performance of individual links and the performance matching between links; it always needs to keep the length of all links not exceeding l_{max}. If l_{max} is too small compared to the map size, merging nodes with links shorter than l_{max} is more difficult. However, the good properties of topologies designed by TriPlace are constantly maintained, Fig. 17 shows the variation of the average hop count of paths in the network with m. As can be seen, the hop count of paths in TriPlace

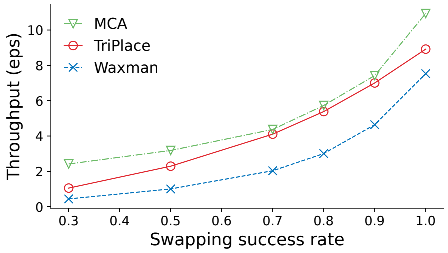

Fig. 13: Throughput vs. swapping success rate q.

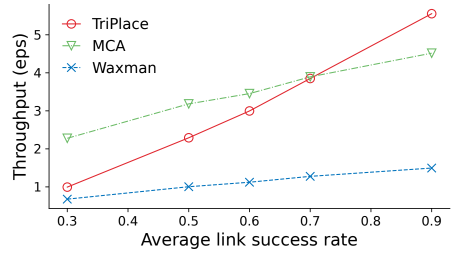

Fig. 14: Throughtput vs. link success rate E_p.

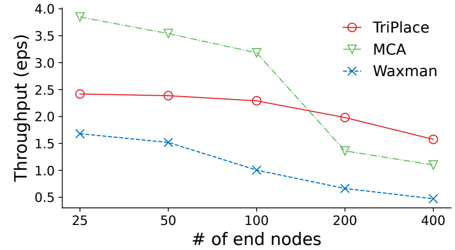

Fig. 15: Throughtput vs. # of end nodes on plane.

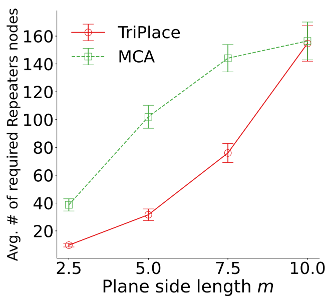

Fig. 16: # of required repeater nodes vs. plane side length m

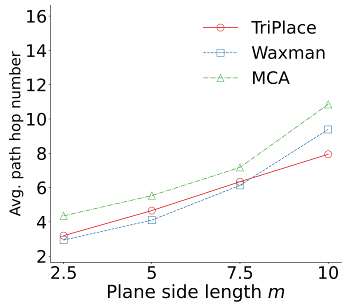

Fig. 17: Paths hop number vs. plane side length m

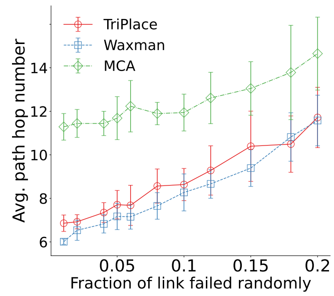

Fig. 18: Avg. path hop number vs. link failure fraction.

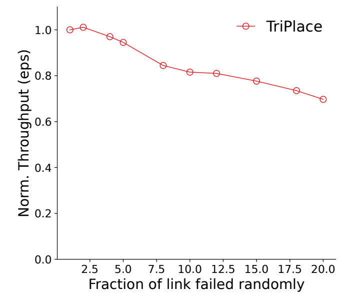

Fig. 19: Normlized Throughput vs. link failure fraction.

grows very slowly with m, and even at $m = 10$, the average hop count is only about 8, which will limit the number of swapping operations, provide better end-to-end entanglement rate and fidelity. Consider that in the real world, we set a reasonable near future $l_{max} = 300km$. (given that there has been a series of recent related hopeful physics experimental work conducted [43]–[45]). Then $m = 10$ represents a total map area of 9 million km^2, nearly the size of the continental United States. Thus, TriPlace exhibits good enough scalability. For larger geographic planes, using fiber-connected ground repeater nodes no longer seems to be a good option, considering that the fidelity and distribution rate of the final end-to-end link still decays exponentially with the number of entanglement swapping. The use of free-space satellite-based repeaters at the edge of two distant geographic planes to avoid photon losses in the optical fiber seems to be a better option, although this technology has no good enough robustness compared to the fiber-connected repeater networks, which is out of the scope of this paper. Besides, trusted repeater nodes as the border gateway of different geographical areas would also be a

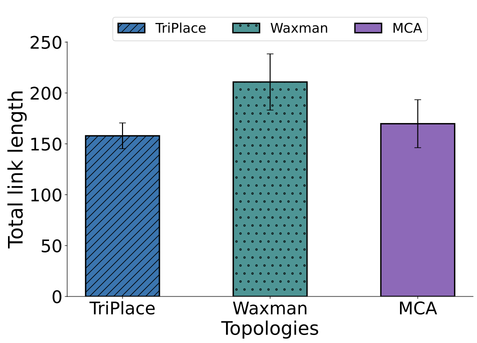

Fig. 20: Total link lengths of different topologies.

potential solution for this problem, which will be future work.

In addition, considering that the cost of links is directly related to their total length when constructing a quantum network, we compare the total link lengths of the topologies generated by the three different methods in the reference setting as shown in Fig. 20. It can be seen that compared to the other two methods, the topology generated by TriPlace requires the least total link length while maintaining good properties, implying the minimum link laying cost.

Throughput. Fig. 13-15 shows the throughput of different topologies for the same set of end nodes. TriPlace's topology achieves a 124% increase in throughput compared to the topology generated using the Waxman model with the same number of repeater nodes and similar degrees. This demonstrates the benefits of TriPlace in minimizing inter-link performance mismatching and limiting link lengths when designing topologies.

The throughput of MCA's topology is slightly better than TriPlace when the network size is small but TriPlace outperforms MCA when the network size is more than 100. Considering that MCA uses about 3x times the number of repeater nodes than TriPlace, TriPlace achieves a far better balance between cost and performance. More specifically, when observing the topologies generated by MCA, we notice that MCA tends to create dedicated links for some end nodes that are not shared with other end nodes when the number of end nodes is small $(25, 50, 100)$. When the number of end nodes is large, it will consider starting the reuse of repeater nodes. MCA uses MST for topology construction, which makes some nodes easily become "hot" nodes in the network and cause congestion when the number of end nodes is large. Fig. 15 also shows that the throughput of TriPlace exceeds that of MCA when the number of end nodes is greater than 100. This is because TriPlace ensures a certain degree of load balancing through the reasonable reuse of repeater nodes and the number of links.

Resiliency to network failures. We proceed to show the network's resilience to failure. Fig. 18 shows the trend of the average hop count of paths in the network with the link failure rate. We set the length of unreachable paths to be a larger penalty value. It can be seen that the hop counts of paths increase for all three different topologies as the fraction of failed links increases, but until the fraction of links failed randomly up to 20%, the hop numbers for both TriPlace and Waxman are smaller than the hop counts for MCA in the absence of failed links. This is guaranteed by the enough degree of nodes in the topologies generated by TriPlace and Waxman. TriPlace's hop count is slightly higher than Waxman's because the extra-long edges in Waxman become "shortcuts" bypassing the necessary repeater nodes, which will cause additional performance loss. Fig. 19 shows how the throughput in TriPlace varies with the fraction of links that failed randomly. When the number of failed links is $5\%, 10\%$ respectively, the traffic of TriPlace is still 95% and 80% of throughput without link failure. When the percentage of failed links is 20%, TriPlace is still able to maintain a throughput of about 70%, which demonstrates a good resiliency to network failure.

V. Related Work

Due to the crucial impact and influence on network performance, topology design is an important and continuous concern in conventional networks [46]–[48], including wide area networks, data center networks, and wireless networks. For developing the emerging quantum networks, although the topology design problem is under-explored, it is an important step that cannot be skipped, despite how the detailed hardware technology envolves.

Several approaches [22], [23], [32] have been proposed for choosing quantum network repeater nodes based on the topologies of existing Internet backbone nodes. Rabbie $et\ al.$ first suggest the concept of designing quantum networks by leveraging pre-existing infrastructure [32]. They present an optimization problem aiming to meet specific rate and fidelity thresholds for a collection of user pairs while minimizing the number of repeaters required in the network and solve this problem using an ILP solver. Pouryousef $et\ al.$ [22] propose a solver-based solution for this problem, but they try to maximize the network utility. Solver-based solutions can reach the optimal but have poor scalability. Islam and Arslan proposed a heuristic for this problem [23]. Since quantum networks use different fiber and hardware than the Internet backbone, limiting the locations of quantum repeaters of Internet backbone nodes may result in a sub-optimal network topology. The distance between certain cities may be far and fibers between them cannot be used to transmit single photons.

In addition to the placement and selection of repeater nodes, the connectivity between nodes is of equal interest. Yu $et\ al.$ [49] discusses the effect of different connectivity design methods between nodes of a wide-area quantum repeater network equivalent to the size of a United State region on the performance of the network. However, they assume strong quantum links that can directly connect any two locations in the USA without the need to set up additional repeater nodes (the equivalent attenuation rate of links is about 0.0002 db/km). In reality, the attenuation rate of high-quality ultralow-loss optical fiber, which is the most effective for building quantum networks for now, is about 0.14 db/km [30], [50], and the success rate of a single entanglement distribution is about 0.1% at the physical layer when the link distance is $150\ km$. In a time slot, multiple entanglement distributions are usually attempted to achieve temporal multiplexing, thus manifesting in the link layer of the quantum network, where the success rate of the link is an overall probabilistic representation of multiple entanglement distributions. Considering recent quantum networks, we still need to set up additional repeater nodes to build wide-area quantum networks.

Correspondence in Classical Networks: In classical networks, topology design is also a matter of concern and has been carefully studied in various types of networks. In wireless sensor networks, different topology control algorithms [51] are proposed to determine the connectivity between sensor nodes to minimize the network's energy consumption. In data center networks, link lengths are usually not sensitive, but careful consideration needs to be given to how to connect a large number of servers using switches to ensure the efficiency of data exchange in computing clusters. Hence topology design is also a problem that has been carefully studied in data center networks with a large number of solutions [47], [52].

VI. Conclusion and future work

This paper presents the problem of quantum network topology design that is not based on existing Internet infrastructure, which has not been studied in the literature. We present a methodology to achieve a desired topology for a given set of end users, including finding optimal link length, placing virtual repeater candidates, merging them, and adjusting their locations. We propose an algorithm, TriPlace, to generate a topology that maintains high network throughput while limiting network cost (mainly the number of repeaters). Simulation results show that TriPlace uses the least cost among existing solutions while achieving similar throughput to the best of them.

As a new research problem, quantum network topology design suggests many possible future directions. For example, quantum memory is also an expensive resource, and the amount on each repeater determines how many connections it can support. Hence, allocating limited quantum memory to different repeaters can be an optimization problem, as different repeaters play different roles. In addition, although TriPlace proposes a method to build a state-level quantum repeater network, for longer distances, end-to-end entanglement rate and fidelity are still limited by exponential decrease with the number of swapping. Thus, a hierarchical solution similar to Internet autonomous systems can be introduced. Moreover, we may consider a mixed topology with trusted and untrusted repeaters, or one with both fiber and satellite links.

VII. Acknowledgment

The authors were partially supported by NSF Grants 2322919, 2420632, 2426031, 2426940.

References

[1] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Theoretical computer science*, vol. 560, pp. 7–11, 2014.

[2] A. K. Ekert, "Quantum cryptography based on bell's theorem," *Physical review letters*, vol. 67, no. 6, p. 661, 1991.

[3] T. van Leent, M. Bock, F. Fertig, R. Garthoff, S. Eppelt, Y. Zhou, P. Malik, M. Seubert, T. Bauer, W. Rosenfeld, *et al.*, "Entangling single atoms over 33 km telecom fibre," *Nature*, vol. 607, no. 7917, pp. 69–73, 2022.

[4] A. S. Cacciapuoti, M. Caleffi, F. Tafuri, F. S. Cataliotti, S. Gherardini, and G. Bianchi, "Quantum internet: Networking challenges in distributed quantum computing," *IEEE Network*, vol. 34, no. 1, pp. 137–143, 2019.

[5] P. Drmota, D. Nadlinger, D. Main, B. Nichol, E. Ainley, D. Leichtle, A. Mantri, E. Kashefi, R. Srinivas, G. Araneda, *et al.*, "Verifiable blind quantum computing with trapped ions and single photons," *Physical Review Letters*, vol. 132, no. 15, p. 150604, 2024.

[6] A. Broadbent, J. Fitzsimons, and E. Kashefi, "Universal blind quantum computation," in *2009 50th annual IEEE symposium on foundations of computer science*, pp. 517–526, IEEE, 2009.

[7] L.-M. Duan, M. D. Lukin, J. I. Cirac, and P. Zoller, "Long-distance quantum communication with atomic ensembles and linear optics," *Nature*, 2001.

[8] E. Schoute, L. Mancinska, T. Islam, I. Kerenidis, and S. Wehner, "Shortcuts to quantum network routing," *arXiv preprint arXiv:1610.05238*, 2016.

[9] 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 inf. 5," 2019.

[10] S. Shi and C. Qian, "Concurrent entanglement routing for quantum networks: Model and designs," in *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*, pp. 62–75, 2020.

[11] Y. Zhao and C. Qiao, "Redundant entanglement provisioning and selection for throughput maximization in quantum networks," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pp. 1–10, IEEE, 2021.

[12] K. Chakraborty, F. Rozpedek, A. Dahlberg, and S. Wehner, "Distributed routing in a quantum internet," *arXiv preprint arXiv:1907.11630*, 2019.

[13] C. Li, T. Li, Y.-X. Liu, and P. Cappellaro, "Effective routing design for remote entanglement generation on quantum networks," *npj Quantum Information*, vol. 7, no. 1, p. 10, 2021.

[14] A. Patil, J. I. Jacobson, E. Van Milligen, D. Towsley, and S. Guha, "Distance-independent entanglement generation in a quantum network using space-time multiplexed greenberger–horne–zeilinger (ghz) measurements," in *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 334–345, IEEE, 2021.

[15] Y. Zeng, J. Zhang, J. Liu, Z. Liu, and Y. Yang, "Multi-entanglement routing design over quantum networks," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*, pp. 510–519, IEEE, 2022.

[16] Y. Zhao, G. Zhao, and C. Qiao, "E2e fidelity aware routing and purification for throughput maximization in quantum networks," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*, pp. 480–489, IEEE, 2022.

[17] Y. Zeng, J. Zhang, J. Liu, Z. Liu, and Y. Yang, "Entanglement routing over quantum networks using greenberger-horne-zeilinger measurements," in *2023 IEEE 43rd International Conference on Distributed Computing Systems (ICDCS)*, pp. 350–360, IEEE, 2023.

[18] Y. Gan, X. Zhang, R. Zhou, Y. Liu, and C. Qian, "A routing framework for quantum entanglements with heterogeneous duration," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 1, pp. 1132–1142, IEEE, 2023.

[19] M. Ghaderibaneh, C. Zhan, H. Gupta, and C. Ramakrishnan, "Efficient quantum network communication using optimized entanglement swapping trees," *IEEE Transactions on Quantum Engineering*, vol. 3, pp. 1–20, 2022.

[20] B. M. Waxman, "Routing of multipoint connections," *IEEE journal on selected areas in communications*, 1988.

[21] F. F. da Silva, G. Avis, J. A. Slater, and S. Wehner, "Requirements for upgrading trusted nodes to a repeater chain over 900 km of optical fiber," *arXiv preprint arXiv:2303.03234*, 2023.

[22] S. Pouryousef, H. Shapourian, A. Shabani, and D. Towsley, "Quantum network planning for utility maximization," in *Proceedings of the 1st Workshop on Quantum Networks and Distributed Quantum Computing*, pp. 13–18, 2023.

[23] T. Islam and E. Arslan, "A heuristic approach for scalable quantum repeater deployment modeling," in *2023 IEEE 48th Conference on Local Computer Networks (LCN)*, pp. 1–9, IEEE, 2023.

[24] G. Vardoyan, S. Guha, P. Nain, and D. Towsley, "On the stochastic analysis of a quantum entanglement switch," *ACM SIGMETRICS Performance Evaluation Review*, vol. 47, no. 2, pp. 27–29, 2019.

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

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

[27] S. Das, S. Khatri, and J. P. Dowling, "Robust quantum network architectures and topologies for entanglement distribution," *Physical Review A*, vol. 97, no. 1, p. 012335, 2018.

[28] S. L. Braunstein and A. Mann, "Measurement of the bell operator and quantum teleportation," *Physical Review A*, vol. 51, no. 3, p. R1727, 1995.

[29] A. A. Hajomer, I. Derkach, N. Jain, H.-M. Chin, U. L. Andersen, and T. Gehring, "Long-distance continuous-variable quantum key distribution over 100-km fiber with local local oscillator," *Science Advances*, vol. 10, no. 1, p. eadi9474, 2024.

[30] Y. Zhang, Z. Chen, S. Pirandola, X. Wang, C. Zhou, B. Chu, Y. Zhao, B. Xu, S. Yu, and H. Guo, "Long-distance continuous-variable quantum key distribution over 202.81 km of fiber," *Physical review letters*, vol. 125, no. 1, p. 010502, 2020.

[31] G. Avis, R. Knegjens, A. S. Sørensen, and S. Wehner, "Asymmetric node placement in fiber-based quantum networks," *arXiv preprint arXiv:2305.09635*, 2023.

[32] J. Rabbie, K. Chakraborty, G. Avis, and S. Wehner, "Designing quantum networks using preexisting infrastructure," *npj Quantum Information*, vol. 8, no. 1, p. 5, 2022.

[33] S. Fortune, "Voronoi diagrams and delaunay triangulations," in *Handbook of discrete and computational geometry*, pp. 705–721, Chapman and Hall/CRC, 2004.

[34] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the royal statistical society. series c (applied statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[35] M. Mitchell, *An introduction to genetic algorithms*. MIT press, 1998.

[36] N. Sangouard, C. Simon, H. De Riedmatten, and N. Gisin, "Quantum repeaters based on atomic ensembles and linear optics," *Reviews of Modern Physics*, vol. 83, no. 1, p. 33, 2011.

[37] C. Jones, D. Kim, M. T. Rakher, P. G. Kwiat, and T. D. Ladd, "Design and analysis of communication protocols for quantum repeater networks," *New Journal of Physics*, vol. 18, no. 8, p. 083015, 2016.

[38] X. Wang, X. Jiao, B. Wang, Y. Liu, X.-P. Xie, M.-Y. Zheng, Q. Zhang, and J.-W. Pan, "Quantum frequency conversion and single-photon detection with lithium niobate nanophotonic chips," *npj Quantum Information*, vol. 9, no. 1, p. 38, 2023.

[39] P. Pathumsoot, N. Benchasattabuse, R. Satoh, M. Hajdušek, R. Van Meter, and S. Suwanna, "Optimizing link-level entanglement generation in quantum networks with unequal link lengths," in *2021 25th International Computer Science and Engineering Conference (ICSEC)*, pp. 179–184, IEEE, 2021.

[40] M. Brazil, R. L. Graham, D. A. Thomas, and M. Zachariasen, "On the history of the euclidean steiner tree problem," *Archive for history of exact sciences*, vol. 68, pp. 327–354, 2014.

[41] G. Xia, "The stretch factor of the delaunay triangulation is less than 1.998," *SIAM Journal on Computing*, 2013.

[42] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*. MIT press, 2022.

[43] Y. Liu, W.-J. Zhang, C. Jiang, J.-P. Chen, C. Zhang, W.-X. Pan, D. Ma, H. Dong, J.-M. Xiong, C.-J. Zhang, *et al.*, "Experimental twin-field quantum key distribution over 1000 km fiber distance," *Physical Review Letters*, vol. 130, no. 21, p. 210801, 2023.

[44] Q.-C. Sun, Y.-F. Jiang, Y.-L. Mao, L.-X. You, W. Zhang, W.-J. Zhang, X. Jiang, T.-Y. Chen, H. Li, Y.-D. Huang, *et al.*, "Entanglement swapping over 100 km optical fiber with independent entangled photon-pair sources," *Optica*, vol. 4, no. 10, pp. 1214–1218, 2017.

[45] X.-M. Hu, Y. Guo, B.-H. Liu, C.-F. Li, and G.-C. Guo, "Progress in quantum teleportation," *Nature Reviews Physics*, vol. 5, no. 6, pp. 339–353, 2023.

[46] D. Ferrari and D. C. Verma, "A scheme for real-time channel establishment in wide-area networks," *IEEE journal on Selected Areas in communications*, vol. 8, no. 3, pp. 368–379, 1990.

[47] A. Singla, C.-Y. Hong, L. Popa, and P. B. Godfrey, "Jellyfish: Networking data centers randomly," in *9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*, pp. 225–238, 2012.

[48] A. Chandra, W. Gong, and P. Shenoy, "Dynamic resource allocation for shared data centers using online measurements," in *Proceedings of the 2003 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pp. 300–301, 2003.

[49] R. Yu, R. Dutta, and J. Liu, "On topology design for the quantum internet," *IEEE Network*, vol. 36, no. 5, pp. 64–70, 2022.

[50] T. Hasegawa, Y. Tamura, H. Sakuma, Y. Kawaguchi, Y. Yamamoto, and Y. Koyano, "The first 0.14-db/km ultra-low loss optical fiber," *SEI Tech. Rev*, vol. 86, pp. 18–22, 2018.

[51] P. Santi, "Topology control in wireless ad hoc and sensor networks," *ACM computing surveys (CSUR)*, vol. 37, no. 2, pp. 164–194, 2005.

[52] S. A. Jyothi, A. Singla, P. B. Godfrey, and A. Kolla, "Measuring and understanding throughput of network topologies," in *SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 761–772, IEEE, 2016.