Fragmentation-Aware Entanglement Routing for Quantum Networks

Shengyu Zhang, Shouqian Shi, Chen Qian, *Member, IEEE*, and Kwan L. Yeung

***Abstract*—New capabilities enabled by quantum networks, e.g. quantum key distribution, rely on the concurrent quantum entanglements via multiple untrusted repeaters between the source-destination pairs, which is known as quantum routing. Existing quantum routing algorithms pre-assign the free qubits to specific paths, in order to avoid resource contention in the process of entanglement swapping. Since not all pre-assigned qubits can be utilized, the resource fragmentation problem caused can limit the network throughout. In this paper, a new fragmentation-aware entanglement routing (FER) algorithm is proposed. With FER, the entangled qubits are shared by all the paths, and the resource contention problem is solved by a priority connection mechanism. Simulation results show that FER consistently outperforms existing algorithms under different network settings.**

***Index Terms*—Fragmentation, entanglement routing, quantum internet, quantum networks.**

I. Introduction

Quantum network is a distributed system that enables the transmission of quantum bits (qubits) between distant quantum devices to achieve novel capabilities that are provably impossible using classical communication [1]–[3]. It is not meant to replace the classical Internet communication. In fact, quantum networks supplement the classical Internet and enable a number of important applications such as quantum key distribution (QKD) [4]–[6], clock synchronization [7], secure remote computation [8], and distributed consensus [9]. Most of them cannot be easily achieved by the classical Internet. Many experimental studies have also demonstrated that long-distance secret sharing via quantum networks are practical. They include the DARPA quantum network [10], SECOQC Vienna QKD network [11], the Tokyo QKD network [12], the satellite quantum network in China [13], the Cambridge quantum network [14], the QKD network in Florence [15], the Madrid Quantum Network [16] and 5GUK Test Network in Bristol [17].

These applications in quantum networks are developed based on the *entanglement of qubits*. The fundamental difference between qubits and classical bits is that a qubit can occupy both zero and one values simultaneously, known as *superposition*. Before we *measure* the state of a qubit, it can represent multiple values in different proportions at the same time. When the state is measured, the superposition *collapses*, and only one single classical bit of information is presented. For some groups of qubits, a strong correlation exists between these qubits that cannot be explained by independent probabilities for individual qubits. Instead, these groups must be considered as a whole, with interdependent probabilities. This phenomenon is known as *quantum entanglement*.

The quantum entanglement brings two radically new features to the quantum network that could not be achieved by classical networks. 1) Quantum entanglement is inherently private by the laws of quantum mechanics such as the "no-cloning theorem" [18] and hence prevents a third party from eavesdropping the

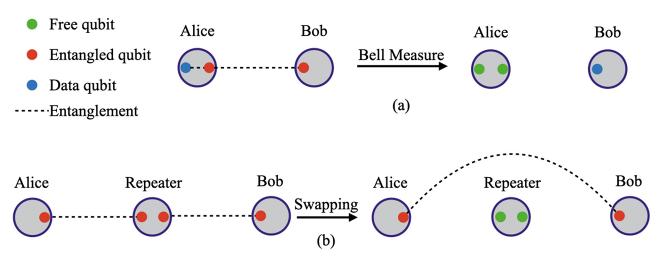

Fig. 1. (a) An example of quantum teleportation, (b) An example of quantum entanglement swapping.

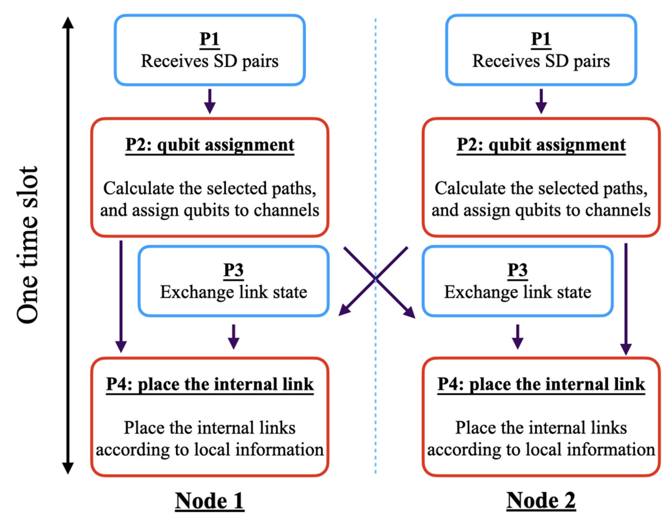

Fig. 2. Flow chart for the four phases in one time slot for two arbitrary neighbor nodes, Node 1 and Node 2.

Manuscript received November 23, 2020; revised February 18, 2021 and March 19, 2021; accepted March 31, 2021. Date of publication April 28, 2021; date of current version July 16, 2021. *(Corresponding author: Shouqian Shi.)*

Shengyu Zhang and Kwan L. Yeung are with the Department of Electrical and Electronic Engineering, University of Hong Kong, Hong Kong (e-mail: zhangsy@eee.hku.hk; kyeung@eee.hku.hk).

Shouqian Shi and Chen Qian are with the Department of Computer Science and Engineering, The University of California, Santa Cruz, California, CA USA (e-mail: sshi27@ucsc.edu; qian@ucsc.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JLT.2021.3070859.

Digital Object Identifier 10.1109/JLT.2021.3070859

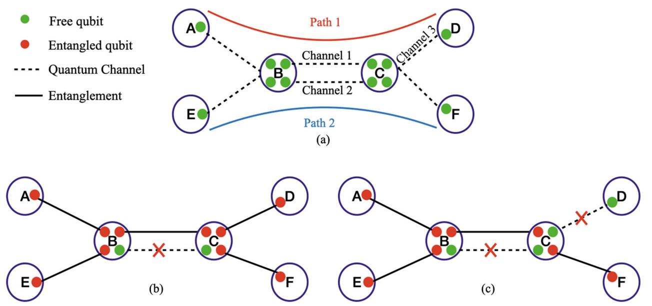

Fig. 3. Examples of resource contention and fragmentation: (a) physical network, (b) an example of resource contention, (c) an example of fragmentation.

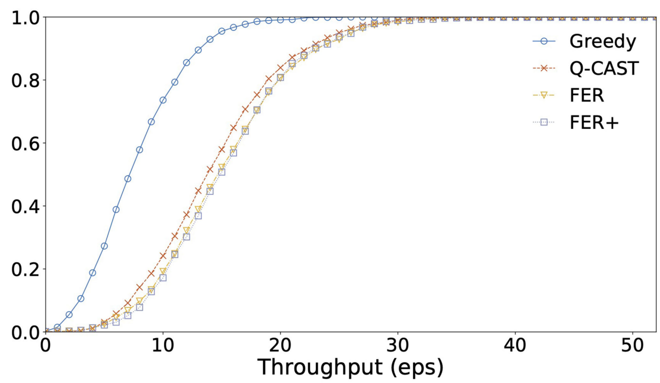

Fig. 4. CDF of throughput under the reference setting.

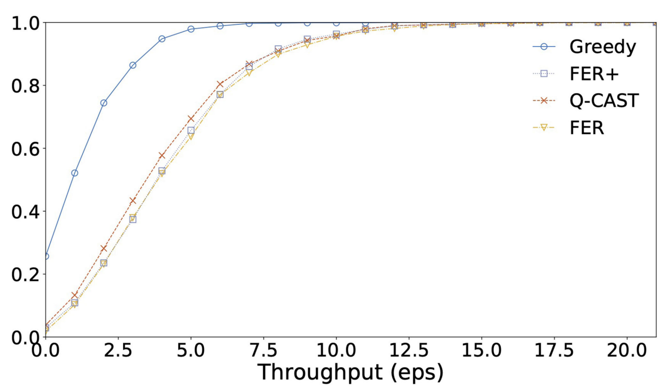

Fig. 5. CDF of throughput, $E_p = 0.3$.

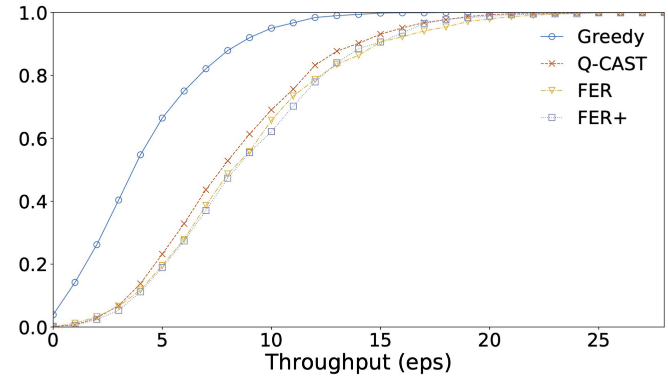

Fig. 6. CDF of throughput, n = 400.

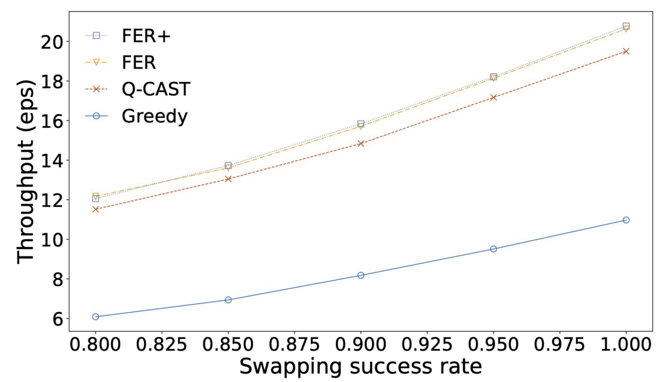

Fig. 7. Throughput vs. channel success rate.

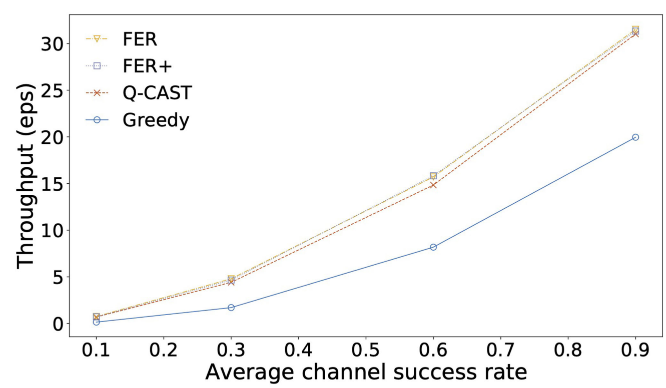

Fig. 8. Throughput vs. swapping success rates.

communication [5], [19], [20]. 2) Quantum entanglement provides strong correlation and instantaneous coordination of the communication parties. Tasks that are difficult to coordinate in a classical network can be easily achieved in the quantum network. One well-known example is *quantum teleportation*. As shown in Fig. 1(a), Alice has two qubits, one is the data qubit to be teleported, and the other one is entangled with Bob. If Alice entangles her two qubits by local operation and measures the two qubits, Alice would obtain two classical bits and destroy all the entanglements. Bob would receive the data bit from Alice directly. To confirm the correctness of the data, this data bit could be encoded with the former received qubits, and sent back to Alice with the help of classical network [21]. This is also the basic approach to achieve QKD via quantum networks.

The entanglement can be generated by sending photons through an optical channel, such as standard telecom fiber [22], [23]. However, the optical fiber is inherently lossy and the probability of successfully establishing an entanglement pair decays exponentially with the physical distance between the two qubits [24], [25]. To increase the probability of successful long-distance quantum entanglement, one solution is to deploy a set of trusted repeaters between the two remote nodes [20], [25]. The data qubit would be forwarded through the chain of

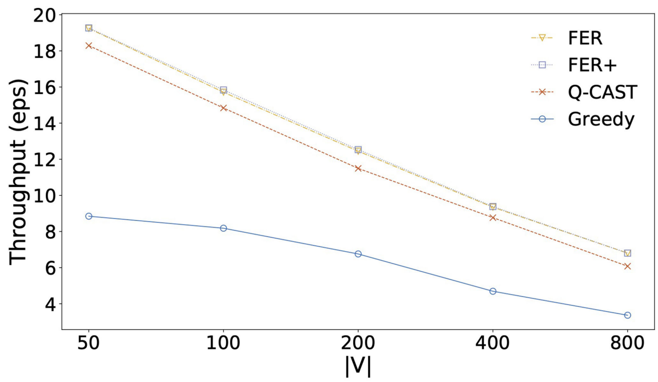

Fig. 9. Throughput vs. network size.

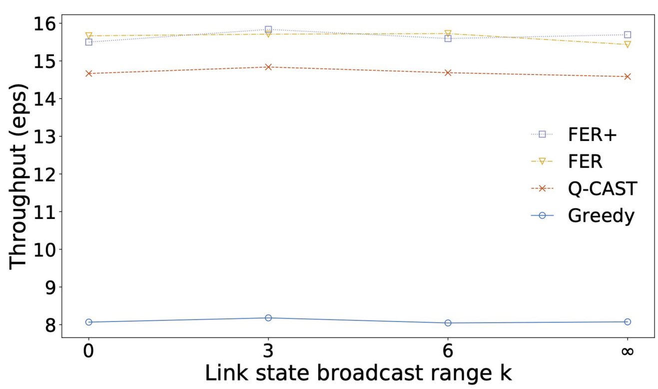

Fig. 10. Throughput vs. node degree.

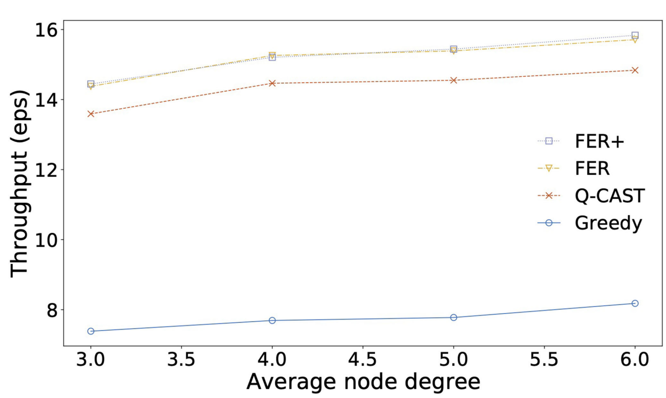

Fig. 11. Throughput vs. SD pairs.

trusted nodes, similar to the *packet switching* in the classical network. However, each trusted repeater needs to get the actual data qubit teleported from the sender. To avoid this, a more attractive approach named *quantum swapping* is proposed [1], [11]–[13], [26]. As shown in Fig. 1(b), one quantum repeater holds entanglements to both Alice and Bob. If the two entangled qubits at the repeater are swapped by local operation, the qubits at Alice and Bob would be entangled directly, and the quantum teleportation can be realized accordingly. Since the data qubit is sent after the swapping, the repeater does not know the qubit information. Therefore, the repeater does not have to be trusted.

In order to design future large quantum networks, one fundamental problem is to route quantum entanglements with high reliability in quantum repeater networks [2]. Although many studies have focused on this area [25], [27]–[33], they are limited to analyzing the traditional routing algorithms on special network topologies. The routing problem in quantum network is fundamentally different from that in classical networks due to two main reasons: 1) The entanglement persistent time is limited, e.g. 1.46s [1]; 2) the optical fiber is inherently lossy and the success of the entanglement cannot be guaranteed. This means that the local link state cannot be propagated to the whole network. Each node in the network only knows the entanglement state within a limited distance. Recently, a practical network model has been presented in [34].

Based on this model, a comprehensive study on entanglement routing in quantum networks has been conducted, where time is loosely synchronized in time slots [31], [34]. Each time slot is a device-technology-dependent constant and set to an appropriate duration such that the established entanglements do not discohere within a slot [31], [34]. During each slot, each node receives the information including network topology and the source/destination (SD) pairs. These nodes calculate the selected paths and qubits assignment according to an identical routing algorithm that ensures consistent results. Then, each node tries to entangle the qubits according to the assignment. After the entanglement, the entanglement states are exchanged among nearby nodes, or nodes within k-hops. After that, each node makes a local decision to swap its own qubits according to the path information and link states. To avoid the link contention, each qubit pair is pre-assigned to a specific path. This means that each path can only utilize the entangled qubit pairs assigned to it. Note that this approach is inefficient because if the entangled qubit pair cannot be utilized by its assigned path, other paths cannot use it. We call those entangled qubit pairs that would not be utilized, *fragments* (detailed in Section III).

In this paper, aiming at minimizing the fragments in the quantum network, we propose a fragmentation-aware entanglement routing (FER) algorithm for quantum networks. With FER, each node ranks the set of paths traversing it, and the entangled qubit pairs will be assigned to the path with the highest priority first. In doing so, entangled qubit pairs are shared and the link contentions[1] are avoided. We also extend FER to FER+. In FER+, additional recovery paths (that cannot cover the entire SD paths) are constructed to fully utilize the free qubits in the network. Simulation results show that FER consistently achieves higher network throughput than the Q-CAST algorithm in [34].

The rest of the paper is organized as follows. In Section II, the system model of the quantum network in [34] is presented. In Section III, the Q-CAST algorithm and the corresponding fragmentation problem is detailed. The two new routing algorithms, FER and FER+, are proposed in Section IV. In Section V,

[1] Each entangled qubit pair can only be utilized by one path. Because the swapping operation would break all the entanglements along the path, and remains the entanglement between source and destination node.

simulation results are presented and we conclude the paper in Section VI.

II. System Model

A. Network Model

We adopt the network model in [34]. Without loss of generality, the network is modelled by an undirected graph $G(V, E, C)$, where V represents the set of nodes, E denotes the set of edges and C is the set of *quantum channels*. Each node v is a quantum processor [35], [36], equipped with a limited number (Q_v) of memory qubits (stored in quantum memories [37], [38]) and necessary hardware to perform quantum entanglement and teleportation on the qubits. It also is equipped with a classical server that enables a certain level of classical computing and storage. All quantum processors are connected via the classical Internet and are able to freely exchange classical information via the classical server. An edge existing between two nodes means that the two nodes share one or more quantum channels. C is the set of all quantum channels, each of which is identified by its two end nodes. A quantum channel connecting two nodes supports the transmission of qubits. The physical material of quantum channels may be polarization-maintaining optical fibers. A quantum channel is inherently lossy: the success rate of each attempt to create an entanglement of a quantum channel c is p_c, which decreases exponentially with the physical length of the channel:

$$p_c = e^{-\alpha L} \tag{1}$$

where L is the physical length of the channel and α is a constant depending on the physical media [20], [25], [31], [39]. The number of channels on an edge is its width, denoted by W. A node can assign/bind each of its quantum memory qubits to a quantum channel [40], [41], such that no qubit is assigned to more than one channel, and no channel is assigned more than one qubits at the same end of it. Channels that are assigned qubits at its both ends are *bound channels*. There could be more than one bound channels between two nodes. And two neighbor nodes may share multiple *quantum links*. To create a quantum entanglement, two neighbor nodes make a number of quantum entanglement attempts at the same time on the bound channels connecting them. If an attempt is successful, the two quantum processors share an entanglement pair, and a quantum link is established on this channel.

B. Quantum Swapping

For multi-hop entanglement swapping, all nodes on the path need to establish and hold quantum entanglements with its predecessor and successor at the same time. It is implemented by synchronizing the time via *time slots* [31], [34]. Within each time slot, a four-phase routing protocol is used to establish an end-to-end entanglement. Fig. 2 shows an example for the four phases in one time slot for two arbitrary neighbor nodes, Node 1 and Node 2. In Phase One (P1), via the Internet, all nodes receive the information of the current S-D pairs that need to establish long-distance entanglements and the network state. In Phase Two (P2), all the nodes calculate the paths according to an identical routing algorithm, which ensures that all the nodes produce consistent results. After that, each node binds its qubits to channels and attempts to generate quantum entanglements with neighbors on the bound channels. In Phase Three (P3), each node shares the link states (i.e., success entanglement or not) with the neighbors within k-hops. In Phase Four (P4), each node *locally* determines the swapping of successful entangled qubit pairs. Then, all the nodes perform entanglement swapping, or establishing *internal links* to connect local qubits. An internal link will be successfully with a probability q. After P4, if the end-to-end entanglement is established, the result will be confirmed by classical Internet.

III. Fragmentation in Entanglement Routing

A. Q-Cast

In this section, we first detail the Q-CAST algorithm in [34]. Notably, Q-CAST aims at maximizing throughput and therefore it does not guarantee that each source-destination pair will be served at each time slot. The related fairness problem could be addressed by providing priority to the requests based on their waiting times. The detailed operation can be found in [34].

In P1, the central controller collects and determines the S-D pairs that need to establish long-distance entanglements in this time slot. Then, each node receives the SD pairs (from central controller) and the network topology (via distributed routing algorithm, e.g. OSPF [42]). In P2, Q-CAST selects *major paths* for each SD pair, without *resource contention*. Besides, contention-free *recovery paths* are also selected in P2. Specifically, for every SD pair, it uses the extended Dijkstra's algorithm (EDA) (see [34]) to find the best path in terms of the expected throughput (EXT) E_t between this pair. Assume that, the quality of a path can be represented by (W, H). Here, H is the length of the path (measured by hops). W is the width of the path, which means that each edge of the path has at least W free channels. Suppose that the success rate of a single channel on the h-th hop is p_h, where $h \in \{1, 2, \ldots, H\}$. Let b_h be the number of successful links at the h-th hop along the path, and Q_h^i be the probability of having exactly i successful links at the h-th hop. Q_h^i can be calculated as:

$$Q_h^i = \binom{W}{i} \times p_h^i \times (1 - p_h)^{W-i} \tag{2}$$

Let P_f^i be the probability of $\min\{b_h : 0 < h \le f\} = i$, or the minimum number of successful links in the first f hops is i. We can get the following recursive formula for calculating P_f^i, for $i \in \{1, 2, \ldots, W\}$ and $f \in \{1, 2, \ldots, H\}$:

$$P_f^i = P_{f-1}^i \times \sum_{l=i}^{W} Q_f^l + Q_f^i \times \sum_{l=i+1}^{W} P_{f-1}^l \tag{3}$$

When $f = h = 1$, we have $P_1^i = Q_1^i$. Further, considering q the success probability of each internal link, we get the EXT:

$$E_t = q^H \times \sum_{i=1}^{W} i \times P_H^i \tag{4}$$

Among these best paths of each SD pair, the one with the maximum EXT is selected and the corresponding resource (qubits and channels) is reserved[2] by *marking* them with the identification of the specific path (P-ID). Then, the network topology is updated by excluding the reserved resources. These operations are repeated until no more major path can be found.

After finding all the major paths, the remaining qubits and channels can be used to construct recovery paths. For every node x on a major path, we use EDA to find the recovery path with the maximum EXT between x and y in the updated graph, where y is the 1-hop ahead node of x on the major path. When all nodes are processed, each single link (1 hop) failure is protected. Then, the algorithm iterates further for the recovery paths that cover l hops on the major path, for $l = 2, 3, \ldots, k$.

In P3, each node exchanges the link state information with the k-hop neighbors. In P4, each node knows the major paths, the recovery paths, and the k-hop link states. Based on them, for each node x, it connects the entangled channels with the same P-ID by internal links. If there are link failures (not entangled bounded channels), a recovery loop will be attempted by an exclusive-or (xor) operation. If successful, internal links will be constructed to connect the corresponding recovery path.

Note that, in order to avoid resource contention between different major paths, marking the network resource (qubits and channels) with a certain P-ID is *essential* in the EDA. In Fig. 3(a), two paths (A $\rightarrow$ D and E $\rightarrow$ F) share a common edge (BC). After P2, the entanglement of the network is shown in Fig. 3(b). In P3, nodes B and C find that only one pair of qubits are entangled between them (channel 1). If this qubit pair has not been assigned to a specific path, then nodes B and C could not make the local decision of which path should this entangled qubit pair be assigned to in P4. This would cause resource contention between different paths.

B. Fragmentation

Q-CAST avoids the network resource contention by marking the network resources (qubits and channels) with a certain P-ID at the cost of inefficient utilization of network resources. For the same paths (A $\rightarrow$ D and E $\rightarrow$ F) in Fig. 3(a), we assume that channel 1 is assigned to path A $\rightarrow$ D and channel 2 is assigned to E $\rightarrow$ F. After P2, the entanglement of the network is shown in Fig. 3(c). If $k = 3$, both nodes B and C notice that channel 2 and channel 3 have failed to make an entanglement. In this case, both paths A $\rightarrow$ D and E $\rightarrow$ F are broken, since channel 1 has already been assigned to path A $\rightarrow$ D, and path E $\rightarrow$ F cannot utilize it.

Note that, since $k = 3$, both nodes B and C have the information of selected paths, and the link states of channels 1, 2 and 3. Hence, B and C have the capability to assign channel 1 to path E $\rightarrow$ F. We call these entangled but not utilized qubit pairs fragments. These fragments reduce the efficiency of the resource utilization in the quantum networks and further decrease the network throughput. It is challenging to conduct defragmentation in the quantum networks due to two main reasons. 1) The entanglement of qubit pairs cannot be guaranteed or even predicted due to the inherently lossy optical fiber. Thus, a pre-calculated or offline assignment is not practical. 2) The entanglement persistence time is limited. This means that the link states cannot be propagated to the whole network. Every node makes the local decision to connect the internal links via only k-hops entanglement states.

[2]This reservation only happens on the logical layer without any physical operations. Since all the nodes run the identical routing algorithm, they will get the same result.

IV. Our Approach

A. Fragmentation-Aware Entanglement Routing

Aiming at minimizing the fragmentation in entanglement routing, we propose a fragmentation-aware entanglement routing (FER) algorithm. The key idea is to turn the "private" network resources earmarked for each specific path, to "public" resources shared by all the paths. The four phases of FER are detailed below.

In P1, each node receives all the SD pairs (Ψ) and the network state $G(V, E, C)$. In P2, FER finds the best path (p_ψ) in terms of the expected throughput (EXT), defined in (4), for each node pair (ψ). Among the best paths of all the SD pairs (P^*), the one with the maximum EXT, or a major path, is selected (p^*) and the corresponding resources (qubits and channels) are *reserved*. Reserved resources are not (yet) earmarked for a particular path (P-ID). The network is then updated to exclude the reserved resources, and the above path finding process repeats until no more major paths can be found. The major paths found are then sorted in the descending order of their EXTs.

Let the path with the maximum EXT be p'. For any edge e along p', assuming its two end nodes are s and t, let the free qubits at s and t be λ_s and λ_t, and the unbounded channels on edge e be μ_e. FER reserves η_e free qubits at both s and t, and binds them to η_e unbounded channels. η_e can be calculated as:

$$\eta_e = \min\{\lambda_s, \lambda_t, \mu_e\} \tag{5}$$

When bounded channels are reserved for each edge on p', the next best path will be considered. The operations above repeat until all the best paths are considered. Then, all the qubit pairs attempt to entangle via their bounded channels. The pseudocode of P2 is shown in Algorithm 1.

In P3, each node sends its entanglement results (or link states) to all other nodes within k-hop. At the end of P3, each node knows the link states within its k-hop and the corresponding paths. In P4, each node (x) sorts the paths passing through it ($P^\star$) in the descending order of EXT. Let the path with the maximum EXT be $p^\star$, and its width be W. In addition to W, node x also calculates the *external width* (W') and *internal width* ($W^\star$) of $p^\star$. External width W' represents that each edge along $p^\star$ (that is within k-hop of x) has at least W' *entangled* channels. Internal width $W^\star$ means that x has at least $W^\star$ free *entangled* channels with its (immediate) predecessor (y) and successor (z), respectively. "Free" means these channels are not connected to any other channels by internal links at x. Therefore, the maximum number of internal links (Ω) that can be constructed

Algorithm 1: P2 in FER Algorithm.

Input: Network status $G(V, E, C)$ and all SD pairs Ψ;
Output: Entanglement link states

$P \leftarrow \varnothing$ /$*$ initialize the set of selected major paths$*$/
while true **do**
  $P^* \leftarrow \varnothing$ /$*$ initialize the set of candidate paths$*$/
  for each SD pair, ψ in Ψ **do**
    Find the path p_ψ with the maximum EXT between SD;
    Add p_ψ into the set of candidate paths P^*;
  if P^* is not empty **then**
    Select the path p^* with largest EXT in P^*;
    Reserve the network resource in G required by p^*;
    Add p^* into P;
  else
    break;
Sort P in descending order of EXT;
while P is not empty **do**
  $p' =$ the first element in P;
  for every edge e along the p' **do**
    Reserve η_e free qubits at each end node to form η_e bounded channels;
  Remove p' from P;
Attempt to entangle all the qubit pairs using their bounded channels;
Return the entanglement link states (i.e., successful or not);

at x is:

$$\Omega = \min\{W, W,' W^\star\} \tag{6}$$

Node x then reserves Ω entangled qubit pairs with y and z (in the logical layer). With (6), the resources for $p^\star$ are not over-reserved. So more resources will be available for the subsequent paths. The operations above are repeated until all the paths traversing x are considered. The priority connection (PC) mechanism above is summarized by Algorithm 2.

B. Fragmentation-Aware Entanglement Routing Plus

FER solves both the contention and the fragmentation problems by introducing a priority mechanism, implemented using Algorithms 1 $\&$ 2. But FER only focuses on the resources along the major paths. It does not utilize the free qubits not on major paths. In FER+, these otherwise-wasted free qubits are used to construct recovery paths. Note that a major path connects a source to its destination. A recovery path connects two nodes along a major path, where the two nodes cannot be the source and destination at the same time. In other words, the recovery path alone cannot be used to establish a S-D entanglement.

FER+ is based on FER for finding major paths, and it modifies two phases of FER, P2 and P4, for finding additional recovery paths. In P2, when all major paths are found by the same procedure in FER, EDA is utilized to find recovery path with the maximum EXT between x and y in the updated graph, where y is the 1-hop ahead node of x on the major path. When all nodes

Algorithm 2: Priority Connection Algorithm.

Input: The set of all major paths (P) and the k-hop link states
Output: Internal links to be added at node x

$P^\star \leftarrow \varnothing$ /$*$ initialize the set of paths traversing node $x*$/
for each path p in P **do**
  if $x \in p$ **then**
    Add p to $P^\star$;
Sort $P^\star$ in descending order of EXT;
while $P^\star$ is not empty **do**
  $p^\star =$ the first element in $P^\star$
  Calculate the Ω according to the $p^\star$ and k-hop link states;
  Reserve Ω entangled qubit pairs between node x and its predecessor y and successor z respectively;
  Remove p' from P;
Return the internal links to be added at node x;

are processed, each single link (1 hop) failure is protected. Then, the algorithm iterates further for the recovery paths that cover l hops on the major path, for $l = 2, 3, \ldots, k$. Each recovery path found is assigned with an increased P-ID.

In P4, FER+ uses an algorithm similar to Algorithm 2 to construct internal links for major paths. The difference is that FER+ does not consider the external width (W') in determining the maximum number of internal links to be constructed (Ω').

$$\Omega' = \min\{W, W^\star\} \tag{7}$$

As compared with (6), more entangled qubit pairs are "saved" for subsequent major paths. As a result, the entangled qubit pairs are assigned to the paths with high EXT, and in case of entanglement failure, they also have a higher probability to be recovered. We call the resulting priority connection mechanism PC+. After all the major paths are considered, each node tries to enhance the major paths by their recovery paths (if any): The recovery paths are first sorted by their P-IDs. Starting with the smallest P-ID value, each recovery path attempts to connect to its major path via internal links enabled by PC+. Note that these decisions are calculated by each node in the logical layer. And, the swapping operations (i.e., connecting internal links) are done in parallel.

C. Complexity Analysis

We denote the number of SD pairs as m, and the maximum width of paths as W_m, which is determined by node capacities and edge widths. The largest hop count of selected paths (with $E_t > 1$) is the maximum hop count (h_m) of all the selected paths. The number of nodes is $|V|$. For a classical network (V, E), the computing complexity of the conventional Dijkstra's algorithm is $O(|V|log|V| + |E|)$, and the computing complexity of EDA is $O(|V|log|V| + |E|(h_m W_m))$ [34].

Let the maximum number of major paths found by FER be $|P|_m$, and the protection paths that found by FER+ and Q-CAST be $|P'|_m$. Since FER only uses the free resources along the path for protection, the computing complexity of finding protection paths with FER is $|E|(h_m W_m)$. On the other hand, the computing complexity of finding protection paths for FER and Q-CAST is $O(|P'|_m|E|(h_m W_m))$. Thus, the computing complexity of FER is $O(|P|_m|V|log|V| + (|P|_m + 1)|E|(h_m W_m))$, and the computing complexity of FER+ and Q-CAST is $O((|P|_m + |P'|_m)(|V|log|V| + |E|(h_m W_m)))$.

V. Simulation Results

We compare the performance of our fragmentation-aware entanglement routing (FER) and FER+ algorithms with existing Q-CAST algorithm [34] and Greedy algorithm (a distributed algorithm based on shortest path routing) [32] by simulations. The simulator developed in [34] is adopted, where the topology of the quantum networks is randomly generated. Assume the area holding a quantum network has a size of 100 K units by 100 K units square, and each unit may be 1 km. The network is generated based on the number of nodes $|V|$, and the average number of node degrees D. Nodes are randomly placed and the distance of any two nodes is at least $\leq 50/\sqrt{|V|}$. The edges are generated by the Waxman model [43] that was used for Internet topologies [44]. The number of free qubits stored in each node is independently uniformly picked from 10 to 14. The width of each edge is independently uniformly generated from 3 to 7. After the generation of the topology, a binary search is utilized to calculate parameter α in (1), which ensures that the average channel success rate to be $E_p \pm 0.01$.

For simplicity, we use the same set of parameters adopted in [34]. We set the number of nodes $|V|$ vary in set $\{50,\ 100,\ 200,\ 400,\ 800\}$, average channel success rate E_p vary in $\{0.6,\ 0.3,\ 0.1\}$, internal link success rate q vary in $\{0.8,\ 0.9,\ 1.0\}$, link state range k vary in $\{0,\ 3,\ 6,\ \infty\}$, average degree D vary in $\{3,\ 4,\ 6\}$, and the number of SD pairs m vary from 1 to 10. For comparison, the results under the *reference setting* $n = 100,\ E_p = 0.6,\ q = 0.9,\ k = 3,\ D = 6,\ m = 10$, is used. For each set of $(n,\ E_p,\ q,\ k,\ D,\ m)$ parameters, 10 random networks are generated, and we simulate 1000 independent time slots on each of the networks.

Fig. 4 shows the CDF of throughputs for Greedy, Q-CAST, FER and FER+, under the reference setting. The throughput results are calculated in terms of ebits (entangled bits) per time slot. As expected, the Greedy shows the lowest throughput. This is because the conventional routing algorithm only focuses on the distance cost of each link. It ignores the width and success rate of each link. The Q-CAST outperforms Greedy by about 5 eps. Benefit from the defragmentation in entanglement routing, FER outperforms Q-CAST by about 1 eps. The gap between FER and FER+ is negligible, since only limited qubits remain on the major paths that can be utilized by recovery paths.

Fig. 5 shows the CDF of throughputs under the reference setting except $E_p = 0.3$. As E_p decreases from 0.6 to 0.3, the gap between FER and Q-CAST decreases from 1 eps to around 0.3 eps. When E_p decreases from 0.6 to 0.3, it is less likely to establish the qubit pair entanglement. Thus, FER has limited gain from the defragmentation operation. Moreover, Fig. 6 shows the CDF of throughputs under the reference setting except $n = 400$. As n increases from 100 to 400, the gap between FER and Q-CAST is similar to the reference setting. This means that the FER algorithm is suitable for large-scale networks. Considering the higher computing complexity of Q-CAST and FER+ (see Section IV.C), FER is a simple and efficient entanglement routing algorithm for future large-scale quantum networks.

Fig. 7 shows the average throughput on different quantum device abilities by varying the swapping (internal link) success rate (q). As q increases, the gap between Q-CAST and FER increases. This is because FER constructs more "continuous" paths (every link along the path has entangled qubit pairs) for swapping and takes the advantage of increased q. Fig. 8 shows the average throughput on different quantum device abilities by varying the average channel success rate (E_q). It is not surprising that, as E_q increases, the gap between Q-CAST and FER increases, since more network resources are available for defragmentation operation.

We also evaluate the scalability of routing algorithms on three dimensions: the size of the network n, the average node degree and the number of concurrent SD pairs m. Fig. 9 shows the throughput under different sizes of networks. All algorithms exhibit a throughput decrease with more nodes in the network. FER+ outperforms others on all network sizes, since it fully utilizes the network resources. Note that, when the network scale is small (e.g., $n = 50$), the gap between Q-CAST and FER is also small, because multiple paths routing is unlikely. Besides, Fig. 10 shows the throughput under different node degrees D in networks. D contributes little to the overall performance because the bottleneck resources in the quantum network are the qubits that are at each node, instead of qubit channels. Finally, as shown in Fig. 11, the throughput of all algorithms grows sub-linearly with the number of SD pairs, due to resource contentions. Nevertheless, FER+ still outperforms others on all the settings.

VI. Conclusion

In this paper, we first identified the fragmentation problem in quantum entanglement routing. Aiming at minimizing the fragments in quantum networks, we proposed two algorithms, FER and FER+. Simulation results show that FER+ always outperforms FER and existing algorithm Q-CAST due to its efficient utilization of network resources. That said, FER has lower computing complexity than FER+ yet can achieve similar performance. Therefore, FER can be a better entanglement routing algorithm for large-scale quantum networks.

References

[1] A. Dahlberg *et al.*, "A link layer protocol for quantum networks," in *Proc. ACM Special Int. Group Data Commun.*, 2019, pp. 159–173.

[2] R. V. Meter and J. Touch, "Designing quantum repeater networks," *IEEE Commun. Mag.*, vol. 51, no. 8, pp. 64–71, Aug. 2013.

[3] R. V. Meter, "Quantum networking and internetworking," *IEEE Netw.*, vol. 26, no. 4, pp. 59–64, Jul./Aug. 2012.

[4] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Theor. Comput. Sci.*, vol. 560, pp. 7–11, Dec. 2014.

[5] A. K. Ekert, "Quantum cryptography based on Bell's theorem," *Phys. Rev. Lett.*, vol. 67, pp. 661–663, Aug. 1991.

[6] S. Pirandola *et al.*, "Advances in quantum cryptography," *Adv. Opt. Photon.*, vol. 12, no. 4, pp. 1012–1236, Dec. 2020.

[7] "A quantum network of clocks," *Nat. Phys.*, vol. 10, no. 8, pp. 582–587,2014.

[8] A. Broadbent, J. Fitzsimons, and E. Kashefi, "Universal blind quantum computation," in *Proc. 50th Annu. IEEE Symp. Found. Comput. Sci.*, Oct. 2009, pp. 517–526.

[9] V. S. Denchev and G. Pandurangan, "Distributed quantum computing: A new frontier in distributed systems or science fiction?," *SIGACT News*, vol. 39, no. 3, pp. 77–95, Sep. 2008.

[10] C. Elliott, "Building the quantum network," *New J. Phys.*, vol. 4, pp. 46–46, Jul. 2002.

[11] M. Peev *et al.*, "The SECOQC quantum key distribution network in vienna," *New J. Phys.*, vol. 11, no. 7, 2009, Art. no. 075001.

[12] M. Sasaki *et al.*, "Field test of quantum key distribution in the tokyo QKD network," *Opt. Exp.*, vol. 19, no. 11, pp. 10387–10409, May 2011.

[13] J. Yin *et al.*, "Satellite-based entanglement distribution over 1200 kilometers," *Science*, vol. 356, no. 6343, pp. 1140–1144, 2017.

[14] J. F. Dynes *et al.*, "Cambridge quantum network," *NPJ Quantum Inf.*, vol. 5, no. 1, p. 101, Nov. 2019.

[15] D. Bacco *et al.*, "Field trial of a three-state quantum key distribution scheme in the florence metropolitan area," *EPJ Quantum Technol.*, vol. 6, no. 1, p. 5, Oct. 2019.

[16] V. Martin *et al.*, "The madrid quantum network: A quantum-classical integrated infrastructure," in *OSA Adv. Photon. Congr.*, 2019, p. QtW3E.5. [Online]. Available: http://www.osapublishing.org/abstract.cfm?URI=Networks-2019-QtW3E.5

[17] R. S. Tessinari *et al.*, "Field trial of dynamic DV-QKD networking in the SDN-controlled fully-meshed optical metro network of the bristol city 5GUK test network," in *Proc. 45th Eur. Conf. Opt. Commun.*, 2019, pp. 1–4.

[18] J. L. Park, "The concept of transition in quantum mechanics," *Foundations Phys.*, vol. 1, no. 1, pp. 23–33, 1970.

[19] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. 22, no. 6, pp. 644–654, Nov. 1976.

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

[21] M. Riebe *et al.*, "Deterministic quantum teleportation with atoms," *Nature*, vol. 429, no. 6993, pp. 734–737, 2004.

[22] J. F. Dynes *et al.*, "Efficient entanglement distribution over 200 kilometers," *Opt. Exp.*, vol. 17, no. 14, pp. 11440–11449, Jul. 2009.

[23] T. Inagaki, N. Matsuda, O. Tadanaga, M. Asobe, and H. Takesue, "Entanglement distribution over 300 km of fiber," *Opt. Exp.*, vol. 21, no. 20, pp. 23241–23249, Oct. 2013.

[24] S. Pirandola, R. García-Patrón, S. L. Braunstein, and S. Lloyd, "Direct and reverse secret-key capacities of a quantum channel," *Phys. Rev. Lett.*, vol. 102, Feb. 2009, Art. no. 050503.

[25] S. Pirandola, R. Laurenza, C. Ottaviani, and L. Banchi, "Fundamental limits of repeaterless quantum communications," *Nat. Commun.*, vol. 8, no. 1, 2017, Art. no. 15043.

[26] H. Bernien *et al.*, "Heralded entanglement between solid-state qubits separated by three metres," *Nature*, vol. 497, no. 7447, pp. 86–90, 2013.

[27] S. Pirandola, "End-to-end capacities of a quantum communication network," *Commun. Phys.*, vol. 2, no. 1, p. 51, 2019.

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

[29] S. Das, S. Khatri, and J. P. Dowling, "Robust quantum network architectures and topologies for entanglement distribution," *Phys. Rev. A*, vol. 97, no. 1, Jan. 2018, Art. no. 012335.

[30] M. Caleffi, "Optimal routing for quantum networks," *IEEE Access*, vol. 5, pp. 22299–22312, 2017.

[31] "Routing entanglement in the quantum internet," *NPJ Quantum Inf.*, vol. 5, no. 1, p. 25, 2019.

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

[33] G. Vardoyan, S. Guha, P. Nain, and D. Towsley, "On the stochastic analysis of a quantum entanglement switch,"*SIGMETRICS Perform, Eval. Rev.*, vol. 47, no. 2, pp. 27–29, Dec. 2019.

[34] S. Shi and C. Qian, "Concurrent entanglement routing for quantum networks: Model and designs," in *Proc. Annu. Conf. ACM Special Int. Group Data Commun. Appl., Technol., Architectures, Protoc. Comput. Commun.*, 2020, pp. 62–75.

[35] C. Monroe and J. Kim, "Scaling the ion trap quantum processor," *Science*, vol. 339, no. 6124, pp. 1164–1169, 2013.

[36] P. Pathumsoot, T. Matsuo, T. Satoh, M. Hajdušek, S. Suwanna, and R. V. Meter, "Modeling of measurement-based quantum network coding on a superconducting quantum processor," *Phys. Rev. A*, vol. 101, May 2020, Art. no. 052301.

[37] B. Julsgaard, J. Sherson, J. I. Cirac, J. Fiurášek, and E. S. Polzik, "Experimental demonstration of quantum memory for light," *Nature*, vol. 432, no. 7016, pp. 482–486, Nov. 2004.

[38] M. P. Hedges, J. J. Longdell, Y. Li, and M. J. Sellars, "Efficient quantum memory for light," *Nature*, vol. 465, no. 7301, pp. 1052–1056, Jun. 2010.

[39] M. Takeoka, S. Guha, and M. M. Wilde, "Fundamental rate-loss tradeoff for optical quantum key distribution," *Nat. Commun.*, vol. 5, no. 1, p. 5235, 2014.

[40] Y. Lee, E. Bersin, A. Dahlberg, S. Wehner, and D. Englund, "A quantum router architecture for high-fidelity entanglement flows in multi-user quantum networks," May 2020, *arXiv:2005.01852*.

[41] D. Luong, L. Jiang, J. Kim, and N. Lütkenhaus, "Overcoming lossy channel bounds using a single quantum repeater node," *Appl. Phys. B*, vol. 122, no. 4, Apr. 2016.

[42] J. Moy, "OSPF Version 2," Internet Draft, RFC 2328, Apr. 1998.

[43] B. M. Waxman, "Routing of multipoint connections," *IEEE J. Sel. Areas Commun.*, vol. 6, no. 9, pp. 1617–1622, Dec. 1988.

[44] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An approach to universal topology generation," in *Proc. MASCOTS, 9th Int. Symp. Model., Anal. Simul. Comput. Telecommun. Syst.*, 2001, pp. 346–353.

Shengyu Zhang received the B.Eng. and M.Eng. degrees in communication engineering from Southeast University, Nanjing, China, in 2016 and 2019, respectively. He is currently working toward the Ph.D. degree with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong. His research interests include optical networks, satellite networks, and quantum networks.

Shouqian Shi received the B.Sc. degree in physics and the B.E. degree in computer science and engineering from the University of Science and Technology of China, Hefei, China. He is currently working toward the Ph.D. degree with the Department of Computer Engineering, University of California Santa Cruz, Santa Cruz, CA, USA. His research interests include computer networks, distributed systems, and other emerging computer systems, including quantum networks, cloud and edge computing, software defined networks, network security and privacy, and network verification. He was the recipient of the Regents' Fellowship from University of California Santa Cruz.

Chen Qian (Member, IEEE) received the B.Sc. degree in computer science from Nanjing University, Nanjing, China, in 2006, the M.Phil. degree in computer science from the Hong Kong University of Science and Technology, Hong Kong, in 2008, and the Ph.D. degree in computer science from The University of Texas at Austin, Austin, TX, USA, in 2013. He is currently an Associate Professor with the Department of Computer Science and Engineering, University of California Santa Cruz, Santa Cruz, CA, USA. He has authored or coauthored more than 60 research papers in a number of top conferences and journals including ACM SIGMETRICS, IEEE ICNP, IEEE ICDCS, IEEE INFOCOM, IEEE PerCom, ACM UBICOMP, ACM CCS, IEEE/ACM TRANSACTIONS ON NETWORKING, and IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS. His research interests include computer networking, data-center networks and cloud computing, Internet of Things, and software defined networks. He is a Member of ACM.

Kwan L. Yeung was born in 1969. He received the B.Eng. and Ph.D. degrees in information engineering from The Chinese University of Hong Kong, Hong Kong, in 1992 and 1995, respectively. In 2000, he joined the Department of Electrical and Electronic Engineering, The University of Hong Kong, where he is currently a Professor. His research interests include next-generation Internet, packet switch or router design, all-optical networks, and wireless data networks.