Segmented Entanglement Establishment for Throughput Maximization in Quantum Networks

*Gongming Zhao, *Jingzhou Wang, *Yangming Zhao, *Hongli Xu and †Chunming Qiao
*School of Computer Science and Technology, University of Science and Technology of China
†Department of Computer Science and Engineering, University at Buffalo

***Abstract*—There are two conventional methods to establish an entanglement connection in a Quantum Data Network (QDN). One is to create single-hop entanglement links first and then connect them with quantum swapping, and the other is forwarding one of the entangled photons from one end to the other with all-optical switching at intermediate nodes to directly establish an end-to-end (E2E) connection. Since a photon is easy to be lost during a long distance transmission, all existing works are adopting the former method. However, in a room size network, the success probability of delivering a photon across multiple links with all-optical switching is not that low. In addition, with an all-optical switching technique, we can save quantum memory at the intermediate nodes. Accordingly, we can expect to establish significantly more entanglement connections with limited quantum resources by first creating entanglement segments, each spanning multiple quantum links, using all-optical switching, and then connecting them with quantum swapping.**

In this paper, we design SEE, an Segmented Entanglement Establishment approach that seamlessly integrates quantum swapping and all-optical switching to maximize quantum network throughput. SEE first creates entanglement segments over one or multiple quantum links with all-optical switching, and then connect them with quantum swapping. It is clear that an entanglement link is only a special entanglement segment. Accordingly, SEE can theoretically outperform conventional entanglement link based approaches. Large scale simulations show that SEE can achieve up to 100.00% larger throughput compared with the state-of-the-art entanglement link based approach, *i.e.*, REPS.

I. Introduction

Quantum networks have been proposed for many decades in order to support highly secure communications [1]–[6]. The main function of conventional quantum networks is to support quantum key distribution (QKD), which is used to establish a shared encryption key between two (classical) computers. In a QKD network, the information is still carried by classic bits. However, with the development of quantum computing, we need to network multiple quantum computers and build a large quantum computing system. To this end, we have to transmit the quantum states without measuring and transferring them into classic data bits. The QKD networks are not adequate for this purpose. To transmit the data quantum bits (called qubits), which carry the quantum state information to be delivered, we need to build Quantum Data Networks (QDNs).

In a QDN, a number of quantum nodes, each serving as a source (Alice), destination (Bob) or quantum repeater, are interconnected with *quantum links*, which are fibers or free-space optical links. Each quantum node has some *quantum memory* to store qubits, and each quantum link carries *quantum channels* (*e.g.*, wavelengths) that can be used to deliver qubits from one of its end to the other. Since a *data qubit* is likely to be lost if it were to be transmitted over one or more channels, and moreover, the qubit cannot be simply copied by Alice for retransmission once it is lost due to the no-cloning theory [7], the prevailing approach used in a quantum network is to establish an *entanglement connection* between Alice and Bob, and then use an approach unique to quantum communication known as *teleportation* to transfer the quantum state information carried by the data qubit from Alice to Bob. Since an entanglement connection can be used to teleport one and only one data qubit, to achieve a high-throughput QDN, we should maximize the number of entanglement connections that can be established with the limited quantum resources, such as quantum memory and quantum channels.

To establish an entanglement connection between Alice and Bob who are not directly connected with each other, the conventional way is to connect multiple entanglement links. More specifically, we will first figure out a physical path from Alice to Bob. Then, over each link between every two physically adjacent quantum nodes along this path, a Bell pair of photons are generated and distributed to the two end nodes to create an *entanglement link*. As a result, there will be a path consists of entanglement links from Alice to Bob. We refer such a path as an *entanglement path*. Along this entanglement path, Alice holds one qubit of a Bell pair and Bob holds a qubit of another Bell pair, while each repeater along the path holds two qubits, belonging to two different Bell pairs. At last, intermediate quantum nodes (*i.e.*, repeaters) along the entanglement path can perform quantum swapping to connect all these entanglement links and establish an entanglement connection. During above procedure to establish an entanglement connection between Alice and Bob, one quantum channel over each quantum link along the path will be consumed to distribute the Bell pair photons. In addition, to store Bell pair photons, one unit of quantum memory will be reserved at Alice and Bob, respectively, while two units of quantum memory will be consumed at each and every of the repeater along the entanglement path.

In this paper, we argue that though it is difficult to establish a long E2E entanglement connection by sending a photon from one end to the other, the success probability is not that low to create an *entanglement segment*, which is a partial entanglement connection over several quantum links (referred to as a *physical segment*), by directly distributing a Bell pair of photons to the two ends of a segment using all-optical switching, especially in a room size QDN. By connecting these entanglement segments with quantum swapping, we can also establish E2E entanglement connections. Though we cannot save any quantum channels, by creating an entanglement segment across several quantum links, we do not need to reserve any quantum memory, the most precious resource in QDNs [8], on the repeaters along this segment. Since entanglement segment is a more general concept (and an entanglement link is a special case of an entanglement link), such an entanglement segment based method brings a significant potential to improve the network throughput.

In this paper, we propose a Segmented Entanglement Establishment (SEE) approach to maximize the throughput of QDNs. As in previous works [9, 10], we assume a QDN works in a time slot fashion, and hence SEE maximizes the number of entanglement connections that can be established in each time slot. Given the topology, network resources (at nodes and along links), the success probabilities associated with creating entanglement segments through different physical segments, the success probability to perform swapping at each repeater, as well as a set of SD pairs, SEE will determine i). which entanglement segments will be created (over which physical segments). As in [10], there will be some redundant entanglement segments in case some of them fail to be created. ii). how to perform quantum swapping to connect the entanglement segments successfully created to establish entanglement connections. Since there are exponential combinations on how to create entanglement segments; and for each entanglement segments, it multiple choices of its physical segments, our problem is more challenging than existing entanglement link based works [9, 10].

To solve above throughput maximization problem, SEE first calculates how many and which entanglement connections we should try to establish for each SD pair, and then figures out the entanglement segments we should try to create in order to establish the desired entanglement connections. Based on the entanglement segments that are created successfully, another efficient algorithm is proposed to determine how to connect them by performing quantum swapping such that the network throughput can be maximized. Through extensive simulations, we find that SEE can increase the throughput over the state-of-the-art technique by up to 100%.

As far as we know, SEE is the first work that integrate all-optical switching and quantum swapping to maximize the throughput of QDNs. The technical contributions of this paper can be summarized as follows:

- Propose a novel approach named SEE to integrate the all-optical switching capability and quantum swapping in order to maximize the throughput of QDNs;
- Design several effective algorithms for SEE;
- Analyze the performance of the proposed algorithms and show that a near-optimal performance can be achieved with high probability.
- Extensive simulations to demonstrate the superior performance of the proposed SEE approach.

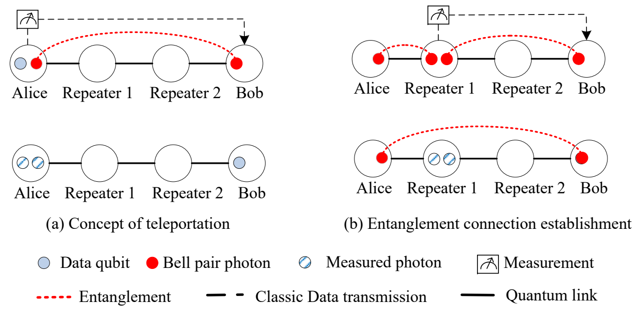


Fig. 1. Teleportation and entanglement connection establishment.

The remainder of the paper is organized as follows. In Section II, we first present related background including related work. Then, in Section III, the algorithm details in SEE are discussed. Extensive simulations are conducted in Section IV to show the superior performance of SEE. We conclude this paper in Section V.

II. Background

In this section, we first present some preliminary background of our work, including how a data qubit is teleported to its destination, how to establish an entanglement connection, and how to create entanglement segments. Based on the background, we motivate the design of SEE with an example. Then, we briefly review some recent works on the entanglement routing in quantum networks. At the end of this section, we present a high-level overview of our work.

A. Teleportation and Entanglement Connection

To teleport a data qubit from Alice to Bob, an entanglement connection (*i.e.*, Alice and Bob each host one qubit from a Bell pair) has to be established between them as shown in the upper plot of Fig. 1(a). Then, Alice measures her two qubits (*i.e.*, the data qubit and the Bell pair photon), and sends the measurement results to Bob through a classic network. Based on the measurement results, Bob will perform some unitary operation on his Bell pair photon. Such an operation transfers the state of his photon to be the same as the data qubit. After above operations, as shown in the lower plot of Fig. 1(a), the entanglement between the two Bell pair photons will be destroyed by the measurement. In addition, the state of Alice's data qubit will also collapse due to the measurement. Accordingly, we can observe that i). the data qubit is not physically sent to Bob; Alice only teleports the state of the data qubit to Bob. ii). an entanglement connection can be used to teleport one and only one data qubit.

To establish an entanglement connection between Alice and Bob, Alice can generate an pair of entangled photons (called Bell pair) using *e.g.*, an Entangled Photon Source (EPS), keep one of the photons to herself and send the other one to Bob. This method is impractical when Alice and Bob are far away from each other, since the photon will be lost on its way to Bob with high probability. To solve this problem, we can first generate multiple entanglement segments (similar to entanglement connection but not directly connect Alice and Bob) to form an *entanglement path*, and then connect these entanglement segments through *swapping*. As shown in the upper plot of Fig. 1(b), there is an entanglement segment between Alice and Repeater 1, and another entanglement segment between Repeater 1 and Bob. To establish an entanglement connection between Alice and Bob, Repeater 1 will perform quantum swapping to connect these two entanglement segments. This operation is akin to teleport the photon entangled with Alice to Bob. It should be noted that through an entanglement path with more than 2 hops, all intermediate repeaters can perform quantum swapping simultaneously and we can connect all those entanglement segments together to establish an entanglement connection. It should be noted that the entanglement links in all previous works [9]–[11] are in fact special entanglement segments. The former should be created over single hop quantum links, while the later can be created over a multi-hop physical segments.

B. Failure of Entanglement Segment Creation and Swapping

An entanglement segment cannot always be created successfully. This may be due to following reasons: i). an EPS may fail to generate a pair of entangled photons; ii). when Alice sends a photon to Bob, due to the signal attenuation, the photon may be lost during the transmission; and iii). when the photon arrives at Bob's side, Bob may fail to detect its arrival. Actually, the success probability to generate an entanglement segment over a single-model fiber one attempt is about 2.18×10^{-4} [12]. Though we can have multiple tries to create such an entanglement segment, the success probability within a time slot is still low according to the state of current technology [9]–[11].

The quantum swapping operation may also fail. To perform quantum swapping, Repeater 1 (in the upper plot of Fig. 1(b)) has to read out its two photons from the quantum memory, and measure their states. Regardless of reading or measuring the photons, Repeater 1 may encounter an error which will result in a failure of establishing the entanglement connection. However, the success probability of swapping would be much higher than creating an entanglement segment. Usually, such success probability will be larger than 0.9 [9, 13].

C. Alternate Ways to Generate Entanglement Segments

To create an entanglement segment, we have following three alternate ways. The most intuitive way is that, as we discussed above, one end of the entanglement segment generates a Bell pair using an EPS, keeps one of them to itself, and sends the remaining one to the other end. In previous works [9]–[11], this operation is over single-hop quantum links. Note that, with the help of all-optical switching, we can create entanglement segments over multiple quantum links (*i.e.*, a physical segment). For example, as shown in Fig. 1, Repeater 1 can generate a Bell pair, and send one of the Bell pair photons to Bob. Repeater 2 only needs to set up an all-optical switching circuit to forward the photon to Bob, without detecting or storing the photon or performing quantum swapping. This method will benefit the QDN throughput in two folds: i). when creating an entanglement segment over multiple links, the intermediate repeaters do not need to reserve quantum memory; ii). the intermediate nodes along a segment do not need to perform quantum swapping, which promotes the success probability to establish an entanglement segment.

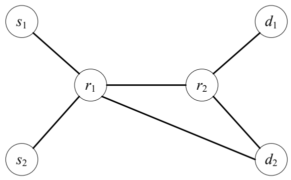
In addition to the most intuitive way discussed above, an EPS can be placed in the middle, *i.e.*, inbetween the two ends of the entanglement segment to be created, to reduce the transmission error when it sends a Bell pair of photos. EPS generates a Bell pair, and sends each of the photons to one of the ends. When both ends of the entanglement segment receive one of the Bell pair photons, they are entangled. Again, take the case in Fig. 1 as an example, an EPS at Repeater 2 can generate a Bell pair, and send each of the photons to Repeater 1 and Bob, respectively. Then, an entanglement segment between Repeater 1 and Bob is created. Combined with the all-optical switching, either Repeater 1 or Bob is not necessarily physically connected with Repeater 2. Compared with the most intuitive way discussed above, place the EPS in the middle of the entanglement segment to be created can reduce the distance a photon to be transmitted and increase the success probability.

The last alternative is more costly in that it requires both ends of an entanglement link having an EPS, and another device to be placed in the middle, which can perform Bell state measurement (BSM). Using this alternative, each end of the entanglement segment generates a Bell pair, and keeps one of the Bell pair photons and sends the other to the BSM device in the middle. Once the BSM device performs measurement successfully, an entanglement segment will be successfully created. For example, Repeater 1 and Bob can generate a Bell pair separately and each of them sends one of the Bell pair photons to Repeater 2. If Repeater 2 successfully performs a BSM, the Repeater 1 and Bob will be entangled. Again, combined with all-optical switching, this method is able to create entanglement segments over more than two physical links. The benefit to create an entanglement link in this way is that the repeater in the middle does not need to detect and store the photons, which can significantly enhance the probability to successfully create an entanglement segment.

Regardless of which method we adopt to create an entanglement segment, one unit of quantum memory is required at each end of it to store the entangled photons. In addition, since we need many attempts in order to create an entanglement segment, one dedicated quantum channel, *e.g.*, a wavelength, will be reserved on all the quantum links that are used to create such an entanglement segment. Compared with conventional entanglement link based methods, we do not need to reserve quantum memory at the intermediate nodes when creating entanglement segments. In this paper, we do not focus on the alternative adopted to create an entanglement segment. If an entanglement segment is to be created, we assume the alternative with largest success probability will be adopted.

D. A Motivation Example

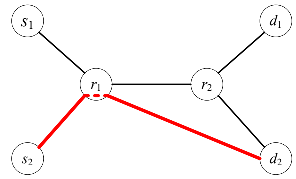
As discussed in last subsection, we have multiple alternatives to create entanglement segments. Under different environments, *e.g.*, length of physical segments and interference from the environment, *etc.*, these alternatives can achieve



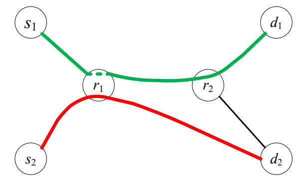
(a) Topology.

Physical segments	Success prob.
$s_1 \to r_1 \to r_2$	0.8
$r_1 \to r_2 \to d_1$	0.85
$s_1 \to r_1 \to r_2 \to d_1$	0.6
$s_2 \to r_1 \to d_2$	0.8
$s_2 \to r_1 \to r_2$	0.8
$r_1 \to r_2 \to d_2$	0.75
$s_2 \to r_1 \to r_2 \to d_2$	0.6

(b) Success probability to create entanglement links.



(c) Conventional solution.



(d) SEE solution.

Fig. 2. A motivation example (Solid black lines are the quantum links; green and red lines are the entanglement links created for SD pair (s_1, d_1) and (s_2, d_2), respectively. Dotted lines are the internal swapping operations to connect multiple entanglement links.).

different success probability by adopting different physical routing and switching schemes. By carefully choosing these alternatives, we will not only maximize the probability to establish an entanglement segment, but also optimally utilize the quantum resources, especially the quantum memory. Thus, we will increase the number of entanglement connections that can be established with limited quantum resources, *i.e.*, increase the network throughput. Fig. 2 shows an example to motivate our work on segmented entanglement establishment via integrating quantum swapping and all-optical switching to maximize the QDN throughput.

Fig. 2(a) shows the network topology of the motivation example. In this network, r_1 and r_2 have 2 units of quantum memory, while the remaining 4 node has only 1 unit of quantum memory; every link carries only 1 quantum channel. In each time slot, the success probability to create an entanglement link over any physical link is assumed to be 0.9, and the swapping success probability at any node is also 0.9. Fig. 2(b) shows the success probabilities to create different entanglement segments.

In the motivation example, we would like to establish entanglement connections for two SD pairs, *i.e.*, (s_1, d_1) and (s_2, d_2). With conventional method, namely, connecting single hop entanglement links, we can establish at most one entanglement connection due to the limitation of quantum memory. By taking into consideration the success probability, the optimal solution is shown in Fig. 2(c) which is able to establish an entanglement connection through the path $s_2 \to r_1 \to d_2$, with the success probability $0.9^3 = 0.729$. The expected number of entanglement connections that can be established is 0.729.

With segmented entanglement establishment approach, we can derive a solution as shown in Fig. 2(d). By creating an entanglement segment over the segment $s_2 \to r_1 \to d_2$, we can save the quantum memory at node r_1, which can be used to host entanglement links (s_1, r_1) and (r_1, d_1) for the SD pair (s_1, d_1). Then, r_1 will perform quantum swapping to establish an entanglement connection between r_1 and d_1. With the solution shown in Fig. 2(d), the expected number of entanglement connections that can be established is $1 \times 0.8 + 1 \times 0.689 = 1.489$, which outperforms the conventional method by 2x.

In this example, we can observe that with segmented entanglement establishment approach that integrates all-optical switching and quantum swapping, we can not only increase the probability to establish entanglement connections, but also save quantum memory which enables us to establish more entanglement connections.

E. Previous Works

For many decades, quantum networks have been proposed for Quantum Key Distribution (QKD) systems [1]–[3, 5], and several real QKD systems have been built around the world, including the US, Europe, Japan, and China [3]–[6]. QKD network is fundamentally different from QDNs since it is used only to establish a shared encryption key between two (classical) computers, and the data in a QKD network is still sent as classic bits. However, a QDN is used to deliver the accurate state of qbits. Due to the no-clone theory [7], we cannot keep a copy of any qubit for retransmission purpose in case that the data qubit is lost during transmission. Once data loss happens, we will not be able to recovery the data to be transmitted. Accordingly, reliability is a critical issue in QDN.

Teleportation can significantly improve the qubit transmission reliability and is widely adopted by QDNs. To improve the throughput of a QDN based on teleportation, we have to maximize the number of entanglement connections that can be established. Early works in this area discussed how to fully utilize the quantum resources to maximize the number of established entanglement connections on some specific types of topology, such as diamond topology [14], ring or sphere topologies [15], star topology [16], and chain topology [17]. After that, [18] and [19] were proposed to establish entanglement connections on a general topology. However, both of them assume the entanglement links have been successfully created and only focus on how to connect the existing entanglement links to form entanglement connections.

The most recent work [9] and [10] considered how to create the entanglement links with limited quantum resources and how to perform quantum swapping to establish entanglement connections. They also took in to consideration the success probability to create entanglement links and perform swapping. However, neither of them considered the alternative based on entanglement segments. [11] is another representative to establish entanglement connections. This work mainly focused on how to physically create the entanglement links and perform swapping, such that the probability to establish an entanglement connection can be maximized.

F. SEE in a Nutshell

Motivated by the superior performance of REPS [10], we assume SEE works in a time-synchronous network operating in time slots, and it also provisions redundant entanglement to deal with the entanglement link failure. Since an entanglement segment would cross multiple quantum links, different from entanglement links in [10], entanglement segments connecting the same two ends may be created over different physical segments. For example, on the topology shown in Fig. 2(a), if we want to create two entanglement segments connecting s_2 and d_2, one of them may go through $s_2 \rightarrow r_1 \rightarrow d_2$, while the other one may go through $s_2 \rightarrow r_1 \rightarrow r_2 \rightarrow d_2$.

In SEE, a central controller maintains all the basic network information, such as the network topology, quantum resources at each node and link, the success probability of swapping on each node, and especially the success probabilities of creating entanglement segments over different physical segments.

With above information, SEE will teleport a batch of data qubits in a time slot through following four steps, which is different from those in [10]:

i). The central controller collects the information about the SD pairs and determines the optimal set of entanglement segments that are to be created. For each entanglement segment, in addition to its two ends, the central controller should also figure out the physical segment to create it. Some of these entanglement segments will be used as backups.

ii). The central controller notifies the corresponding nodes to reserve quantum memory, generate Bell pairs, set up all-optical switching circuits, and send out photons, in order to create entanglement segments, though not all the entanglement segments can be created successfully.

iii). Every node reports back the successfully created entanglement segments. Based on this information, the central controller will try to figure out how to perform the swapping operation to establish entanglement connections.

iv). Corresponding nodes (*i.e.*, the destination node of every entanglement connection) report the swapping result to the source node. If all related swapping operations associated with an entanglement path succeeds, the source node can teleport data qubits to their destination.

Apparently, the key steps in SEE are the first and third steps where the central controller has to determine how to create entanglement segments and how to connect the successfully created entanglement segments to establish entanglement connections. In the next section, we will describe these two algorithms in detail.

III. SEE Design

In this section, we first formulate the problem to be solved in Section III-A, and then design efficient algorithms to solve the formulation in Section III-B. Theoretical analysis on the proposed algorithms will be presented in Section III-C. For clear presentation, the notations used in this section are summarized in Tab. I.

A. Problem Formulation

In this section, we formulate the problem to maximize the network throughput in terms of the number of entanglement connections that can be established in each time slot. The formulation is shown in (1). Compared with the formulation in [10], we directly determine if each entanglement connection should be established or not without estimating the number of entanglement connections that should be established for each SD pair. Accordingly, the variables to formulate the entanglement path $f_i^n(u,v)$ are binary variable, rather than integer variables, which will benefit our algorithm design (see details in Section III-B).

TABLE I
Notation list

Parameters	Description
$(\boldsymbol{V}, \boldsymbol{E})$	Network topology. $\boldsymbol{V}$ is the set of quantum nodes, while $\boldsymbol{E}$ is the set of quantum links.
s_i	Source node of i^{th} SD pair.
d_i	Destination node of i^{th} SD pair.
c_{uv}	Number of quantum channels over link $(u,v) \in \boldsymbol{E}$
p_{uv}^k	Success probability of creating an entanglement link (u,v) over the k^{th} segment between u and v.
m_u	Quantum memory size at node u.
q_u	Success probability of a quantum swapping operation at node u.
N_i	The number of entanglement connections we are trying to establish for SD pair i.
n_i	$0 \le n_i \le N_i$. The index of an entanglement connection we are trying to establish for SD pair i. We also use n_i to refer to the n_i^{th} entanglement connection established for SD pair i. Without ambiguity, we may ignore the subscript.
k_{uv}	The number of physical segments over which we can create entanglement link (u,v).
C_{uv}^k	The k^{th} physical segments to create entanglement link (u,v).
Variables	**Description**
$f_i^n(u,v)$	Binary variable indicating if or not an entanglement link (u,v) is used to establish the n^{th} entanglement connection for SD pair i.
t_i^n	Binary variable indicating if or not the n^{th} entanglement connection for SD pair i will be established.
x_{uv}^k	Number of entanglement link (u,v) that will be created through the k^{th} segment between u and v.

$$\max \quad \sum_i \sum_n t_i^n \tag{1}$$

Subject to:

$$\sum_v f_i^n(u,v) - \sum_v f_i^n(v,u) = t_i^n, \ \forall u = s_i, \ n \le N_i \tag{1a}$$

$$\sum_v f_i^n(u,v) - \sum_v f_i^n(v,u) = -t_i^n, \ \forall u = d_i, \ n \le N_i \tag{1b}$$

$$\sum_v f_i^n(u,v) - \sum_v f_i^n(v,u) = 0, \ \forall u \neq s_i, d_i, \ n \le N_i \tag{1c}$$

$$\sum_{i,n} [f_i^n(u,v) + f_i^n(v,u)] \le \sum_k p_{uv}^k x_{uv}^k \sqrt{q_u q_v}, \ \forall u, v \tag{1d}$$

$$\sum_{u,v,k:(i,j) \in C_{uv}^k} x_{uv}^k \le c_{ij}, \ \forall (i,j) \in E \tag{1e}$$

$$\sum_{v,k} x_{uv}^k \le m_u, \ \forall u \tag{1f}$$

$$t_i^n \ge t_i^{n+1}, \ \forall i, n < n_i \tag{1g}$$

$$f_i^n(u,v), \ t_i^n \in \{0,1\}, \ x_{uv}^k \in \boldsymbol{N} \tag{1h}$$

The objective of (1) is to maximize the number of entanglement connections that will be established. The first three constraints, *i.e.*, $(1a)$–$(1c)$, are flow conservation constraints which should be held in all the routing related problems. Constraint $(1d)$ states that the number of entanglement segments from node u to node v used by all the SD pairs to establish entanglement connections cannot exceed the expected number of entanglement segments that can be created. It should be noted that in this constraints, i). u and v could be separated by multiple quantum links and each entanglement segment can be created over either of the k_{uv} physical segments we have prepared for it; ii). we apportion the success probability of quantum swapping operations to the incident entanglement segments [10]. $(1e)$ says that the number of entanglement segments that we are trying to create going through quantum link (u, v) must be less than or equal to the number of quantum channels carried by (u, v). To create an entanglement segment that incidents upon node u, one quantum memory is required. Constraint $(1f)$ states that number of entanglement segments that incident on node u cannot exceed its quantum memory size. Constraint $(1g)$ is an auxiliary constraint to limit the solution space and reduce the problem complexity. Given an entanglement path, we can assign it an arbitrary index, which significantly increases the solution space without bringing any benefit to improve the objective. This constraint can limit an entanglement path to be built unless all the entanglement paths labeled by a smaller index have been built. $(1h)$ states that the number of entanglements on every edge to be integral.

The problem (1) is difficult to solve due to the integral natural of the variables. In fact, we have following theorem.

Theorem 1. *The problem formulated in* (1) *is NP-hard.*

Proof: By setting the success probability to create an entanglement segment over multi-hop physical segments to be 0, the success probability to create an entanglement segment over single-hop quantum links to be 1, and the success probability of quantum swapping operation to be 1, the problem in (1) will be reduced to a classic integer multi-commodity flow problem, which is a well-know NP-hard problem [20]. ■

Due to the complexity of the proposed problem, we will design efficient algorithms to solve it in the next subsection.

B. Algorithm Design

In this subsection, we will propose a series of algorithms to solve the problem formulated in last subsection. At first, we will derive a set of entanglement paths to establish the entanglement connections with Entanglement Path Identification (EPI) algorithm. To establish as many entanglement connections as possible according to the entanglement paths identified by Algorithm EPI, Entanglement Segment Creation (ESC) algorithm is leveraged to determine how many entanglement segments will be created over each physical segment. Since some of the entanglement segments cannot be successfully created, Entanglement Connection Establishment (ECE) algorithm is proposed to determine how to establish entanglement connections by connecting the entanglement segments successfully created.

1) Entanglement Path Identification (EPI): The entanglement paths identified by solving (1) will maximize the network throughput. However, (1) is difficult to solve. Accordingly, we propose an Entanglement Path Identification (EPI) algorithm based on randomized rounding to derive a near-optimal solution. The basic idea of Algorithm EPI can be summarized as follows: we first relax the integral constraint of $(1h)$, and solve the derived linear programming (LP) model. Then, we derive a solution to (1) based on the solution of this LP via randomized rounding. The details of this algorithm are shown in Algorithm 1.

Algorithm 1: Entanglement Path Identification (EPI) Based on Randomized Rounding

Input: The formulation of (1)
1: **Step 1: Solving the Relaxed Formulation**
2: Construct a linear program by relaxing the integral constraints $(1h)$ as $f_i^n(u,v) \in [0,1]$, $t_i^n \in [0,1]$, and $x_{uv}^k \geq 0$
3: Solve the LP and obtain the optimal solutions $\{\widetilde{f}_i^n(u,v)\}$ and $\{\widetilde{t}_i^n\}$
4: **Step 2: Identify entanglement paths via randomized rounding**
5: Set $t_i^n = 1$ with the probability $\widetilde{t}_i^n$
6: **for** All n_i such that $t_i^n = 1$ **do**
7: Calculate the set of paths traversed by the entanglement connection n_i according to $\{\widetilde{f}_i^n(u,v)\}$
8: Say the path set is $\{P_{ni}^{(r)}\}$, $P_{ni}^{(r)}$ also denotes the fraction of flow going through the corresponding path
9: Select one path $P_{ni}^{(r)}$ with probability $P_{ni}^{(r)}/\widetilde{t}_i^n$
10: Set $f_i^n(u,v) \leftarrow 1$ for all $(u,v) \in P_{ni}^{(r)}$ and $f_i^n(u,v) \leftarrow 0$ for all $(u,v) \notin P_{ni}^{(r)}$
11: **end for**
12: **return** $f_i^n(u,v)$ and t_i^n

This algorithm contains two steps. In the first step (Lines 1–3), we relax the Problem 1 and solve it. The solution of the relaxed model is usually infeasible to Problem 1 due to two reasons: i). an entanglement path may not be fully satisfied, *i.e.*, $0 < t_i^n < 1$; and ii). an entanglement path will be split onto multiple paths, *i.e.*, $0 < f_i^n(u,v) < 1$. Algorithm EPI solves these two problems in Step 2. At first, it figures out which entanglement paths will be built up (Line 5), *i.e.*, rounds t_i^n to be a binary value, and then indicates the corresponding concrete path (Lines 6–11), *i.e.*, round $f_i^n(u,v)$ to be a binary value. The first rounding is based on how much fraction of each corresponding entanglement path is satisfied according to the LP solution, while the seconding rounding is based on the fraction of the entanglement path carried by different paths.

We will show the effectiveness of Algorithm EPI in terms of its ability to achieve optimal throughput and produce feasible entanglement paths using theorems in Section III-C.

2) Entanglement Segment Creation (ESC): Due to the randomized rounding, we cannot ensure the entanglement paths derived by Algorithm EPI are feasible solutions to (1). In addition, Algorithm EPI only specifies the entanglement segments that will be created to establish each entanglement connection, but not how many entanglement segments should be created over different physical segments and how to connect the suc-

Algorithm 2: Entanglement Segment Creation (ESC) Algorithm

Input: The set of entanglement paths identified by Algorithm ESC $\boldsymbol{T}$

1: Reorder all the entanglement paths
2: Initialize the number of entanglement segments created over each physical segment $x_{uv}^k \leftarrow 0$, and the set of all the entanglement paths for which we have allocated quantum resources $\boldsymbol{D} \leftarrow \Phi$
3: **for** Any path $p \in \boldsymbol{T}$ **do**
4: $\quad \boldsymbol{D} \leftarrow \boldsymbol{D} \cup p$
5: $\quad$ **for** Any entanglement segment $(u, v) \in p$ **do**
6: $\quad\quad$ Assign minimum quantum resources on segment $\langle u, v\rangle$ such that $\sum_{p \in \boldsymbol{D}} I_{\langle u,v\rangle \in p} \leq \sum_k p_{uv}^k x_{uv}^k$
7: $\quad\quad$ Update the quantum resource assignment x_{uv}^k
8: $\quad\quad$ **if** qunatum resources are not enough **then**
9: $\quad\quad\quad$ Release all the quantum resources assigned for p, $\boldsymbol{D} \leftarrow \boldsymbol{D}/p$
10: $\quad\quad\quad$ **break**;
11: $\quad\quad$ **end if**
12: $\quad$ **end for**
13: **end for**
14: **return** $\{x_{uv}^k\}$ and $\boldsymbol{D}$

cessfully created entanglement links to establish entanglement connections. We will propose Entanglement Segment Creation (ESC) algorithm and Entanglement Connection Establishment (ECE) algorithm to address these two issues, respectively. The goals of Algorithm ESC are that i). establish as many entanglement connections identified by Algorithm EPI (Say the set of entanglement paths identified by Algorithm EPI is $\boldsymbol{T}$) as possible; and ii). pursue the fairness among all the SD pairs. To pursue the first goal, we first reserve quantum resources to the entanglement paths with fewer hops, and use the physical segments that have higher probability to successfully create entanglement segments directly with all-optical switching of a Bell pair photon (rather than using quantum swapping). For the second goal, we reserve quantum resources to each SD pairs following round robin principle. Following the line of these thoughts, we propose the Algorithm ESC shown in Algorithm 2.

In Line 1, Algorithm ESC first sorts all the entanglement paths in the increasing order of path length (entanglement segment number first and then physical hop number). This is to increase the network throughput since the entanglement paths with fewer hops will require less quantum resources. Then, with the equal path length, all the SD pairs will be ordered based on round robin in order to pursue the fairness among all SD pairs. In Lines 3–13, we reserve quantum resources along each path $p \in \boldsymbol{T}$ to ensure that the expected number of entanglement segments that can be created will be enough to build all the entanglement paths for which we have already reserved resources (Line 6). To save the quantum resource, *i.e.*, minimize the number of entanglement segments that we are trying to create, the physical segments with higher probability to create an entanglement segment will be used first. It should be noted that if we cannot assign enough resources to an entanglement path, all the quantum resources reserved for this entanglement path should be released (Line 9). When all the entanglement paths have been traversed, Algorithm ESC returns the number of entanglement segments we will try to create over each physical segment and the set of entanglement paths for which we have reversed enough quantum resources.

Algorithm 3: Entanglement Connection Establishment (ECE) Algorithm

Input: The number of entanglement segments successfully created over each segment $\{e_{uv}\}$, the set of entanglement paths for which we have reserved enough quantum resources $\boldsymbol{D}$, and the set of entanglement paths identified by Algorithm ESC $\boldsymbol{T}$

1: Initialize $\boldsymbol{O} \leftarrow \Phi$
2: **for** Any entanglement path $p \in \boldsymbol{D}$ **do**
3: $\quad$ **if** $e_{uv} \geq 1$ for all $\langle u, v\rangle \in p$ **then**
4: $\quad\quad e_{uv} \leftarrow e_{uv} - 1$ for all $\langle u, v\rangle \in p$, $\boldsymbol{O} \leftarrow \boldsymbol{O} \cup p$
5: $\quad$ **end if**
6: **end for**
7: Initialize an auxiliary graph $G = \langle \boldsymbol{V}, \boldsymbol{S}\rangle$, where $\boldsymbol{S}$ is the set of all entanglement segments successfully created
8: Set the weight of each node $u \in \boldsymbol{V}$ as $-\ln q_u$
9: **while** More entanglement connections can be established **do**
10: $\quad$ **for** Any SD pair i with fewer than N_i entanglement connections in $\boldsymbol{O}$ **do**
11: $\quad\quad$ Set the weight of an edge $\langle u, v\rangle$ to be 10^{-5} if $e_{uv} \geq 1$, and 10^9 if $e_{uv} = 0$
12: $\quad\quad$ Find the shortest path from s_i to d_i, say the path is p
13: $\quad\quad e_{uv} \leftarrow e_{uv} - 1$ for all $\langle u, v\rangle \in p$, $\boldsymbol{O} \leftarrow \boldsymbol{O} \cup p$
14: $\quad$ **end for**
15: **end while**
16: **return** $\boldsymbol{O}$

3) Entanglement Connection Establishment (ECE): Algorithm ESC tells us how many entanglement segments will be created over different physical segments. However, we should note that only part of these entanglement segments will be created successfully, and we still have to determine how to perform quantum swapping to establish the entanglement connections. To this end, we propose Entanglement Connection Establishment (ECE) algorithm as shown in Algorithm 3.

The input of Algorithm ECE is the entanglement segments that are successfully created in the second step of each time slot, e_{uv}. It should be noted that when an entanglement segment is created, we do not care about the physical segment over which it was created. Based e_{uv}, Algorithm ECE first assigns the created entanglement segments to the entanglement paths in $\boldsymbol{D}$, *i.e.*, the entanglement paths which we have reserved enough quantum resources for (Lines 2–6). On the one hand, since some of the entanglement segments may fail to be created, we may not be able to built all the entanglement paths in $\boldsymbol{D}$. As a result, there may leave some entanglement segments. On the other hand, we may created more entanglement segments than we expect since we have created some redundant entanglement segments in case some of the entanglement segments may fail to be created. It will also leave some entanglement segments that are successfully created but cannot be used by the entanglement paths in $\boldsymbol{D}$. Accordingly, Algorithm ECE leverages these entanglement segments to establish more connections and improve the network throughput (Lines 9–15). To this end, Algorithm ECE first constructs an auxiliary graph on which the vertexes represent the repeaters while each edge stands for an entanglement segment (Line 7). Then, the weight of each vertex u is set to be $-\ln q_u$ (Line 8), and the weight of each edge is set to be a small number (10^{-5} in Algorithm ECE) if there are still remaining corresponding entanglement segments, while a large number (10^9 in Algorithm ECE) if all corresponding entanglement segments are assigned to some entanglement paths (Line 11). In this way, maximizing the probability to establish an entanglement connection is equivalent to minimize the length of the corresponding entanglement path from Alice to Bob (Line 12). Algorithm ECE will end when it cannot find out more entanglement paths based on the remaining entanglement segments in the network.

C. Algorithm Analysis

This section analyze the efficiency of Algorithm EPI.

Theorem 2. *Suppose O_{LP} is the optimal objective value to the relaxed version of formulation* (1)*, while O_{ALG} is the objective value achieved by Algorithm EPI, then we have* $\textbf{\textit{Pr}}[O_{ALG} \leq (1-\epsilon)O_{LP}] \leq e^{-\epsilon O_{LP}/2}$.

Proof: According to Algorithm EPI, we have $E[t_i^n] = 1 \times \widetilde{t}_i^n + 0 \times (1-\widetilde{t}_i^n) = \widetilde{t}_i^n$. Then, $E[O_{ALG}] = \sum_i \sum_n E[t_i^n] = \sum_i \sum_n \widetilde{t}_i^n = O_{LP}$. Based on Chernoff Bound, we know $\textbf{Pr}[O_{ALG} \leq (1-\epsilon)E[O_{ALG}]] \leq e^{-\epsilon E[O_{ALG}]/2}$. Combining above discussions, we conclude $\textbf{Pr}[O_{ALG} \leq (1-\epsilon)O_{LP}] \leq e^{-\epsilon O_{LP}/2}$. ■

Let y_{uv} be the number of entanglement segments (u,v) we are trying to create and $C_{uv} = \cup_k C_{uv}^k$we have

Theorem 3. $\textbf{\textit{Pr}}[\sum_{u,v:(i,j)\in C_{uv}} y_{uv} \geq (1+\epsilon)c_{ij}]] \leq e^{-\frac{\epsilon^2}{2+\epsilon}c_{ij}/2}$ *for all quantum link $(u,v) \in \boldsymbol{E}$*

This theorem shows that the solution derived by Algorithm EPI will satisfy the link capacity constraint with a high probability.

Proof: According to Algorithm EPI, we know $f_i^n(u,v)$ would be set to 1 with the probability $\widetilde{t}_i^n \times \widetilde{f}_i^n(u,v)/\widetilde{t}_i^n = \widetilde{f}_i^n(u,v)$. So we have $E[f_i^n(u,v)] = 1 \times \widetilde{f}_i^n(u,v) + 0 \times (1-\widetilde{f}_i^n(u,v)) = \widetilde{f}_i^n(u,v)$. The expected number of entanglement segment (u,v) that can be created successfully is $\sum_i \sum_n [E[f_i^n(u,v)] + E[f_i^n(v,u)] = \sum_i \sum_n [\widetilde{f}_i^n(u,v) + \widetilde{f}_i^n(v,u)] \leq \sum_k p_{uv}^k \widetilde{x}_{uv}^k \sqrt{q_u q_v}$.

From the definition of y_{uv}, we have $\sum_{u,v} y_{uv} = \sum_i \sum_n [\widetilde{f}_i^n(u,v) + \widetilde{f}_i^n(v,u)] \leq \sum_k p_{uv}^k \widetilde{x}_{uv}^k \sqrt{q_u q_v}$. Since $p_{uv}^k \leq 1$, $q_u \leq 1$ and $q_v \leq 1$, we know $\sum_{u,v:(i,j)\in C_{uv}} y_{uv} < \sum_{u,v} y_{uv} < \sum_k \widetilde{x}_{ij}^k \leq c_{ij}$. Based on the Chernoff Bound, we have $\textbf{Pr}[\sum_{u,v:(i,j)\in C_{uv}} y_{uv} \geq (1+\epsilon)c_{ij}]] \leq e^{-\frac{\epsilon^2}{2+\epsilon}c_{ij}/2}$. ■

Theorem 4. $\textbf{\textit{Pr}}[\sum_v y_{uv} \geq (1+\epsilon)q_u] \leq e^{-\frac{\epsilon^2}{2+\epsilon}q_u/2}$ *for all quantum node $u \in \boldsymbol{V}$.*

Proof: The proof of Theorem 4 is similar to that of Theorem 3. Due to space limitations, we omit it here. ■

Theorem 4 shows that the quantum memory capacity can be satisfied with a high probability.

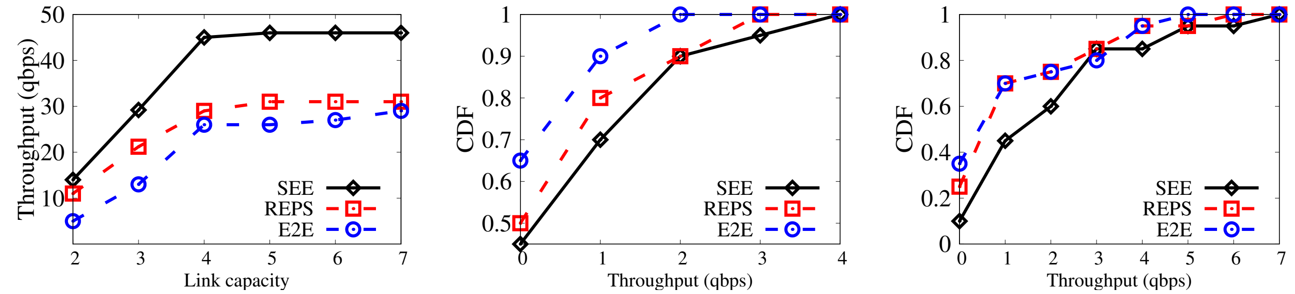
D. Discussions

Physical segments to create entanglement links: In SEE, we have to prepare several physical segments to create each specific entanglement segment. The more physical segments we prepared for each entanglement segment, the better solution, *i.e.*, the higher network throughput, we will achieve. However, it will significantly increase the problem complexity when we increase the number of physical segments prepared for entanglement segment. In SEE, we will find out K physical segments for every node pair with Yen's algorithm [21]. However, the segments consist of too many hops or with a low probability to create an entanglement segment will be removed. This is to reduce the time complexity of our algorithms.

Time complexity: Though SEE can leverage the same algorithms in REPS to calculate how many entanglement segments should be created over each physical segment, it will incur an extremely large time complexity as there are much more physical segments than physical links in a network. In addition, REPS uses progressive rounding to determine the number of entanglement segments that should be created over each physical segment. With this method, we have to solve plenty of LP models, though the scale of LP models will decrease with the process of the algorithm, it is still time consuming when there are lots of quantum links in the network. Accordingly, we design a set of new algorithms for SEE, in which the LP model will be solve only once. Since we introduce larger search space into the SEE by creating entanglement segments through different physical segments, we will see in the simulations that SEE outperforms REPS by 2x in term of the network throughput.

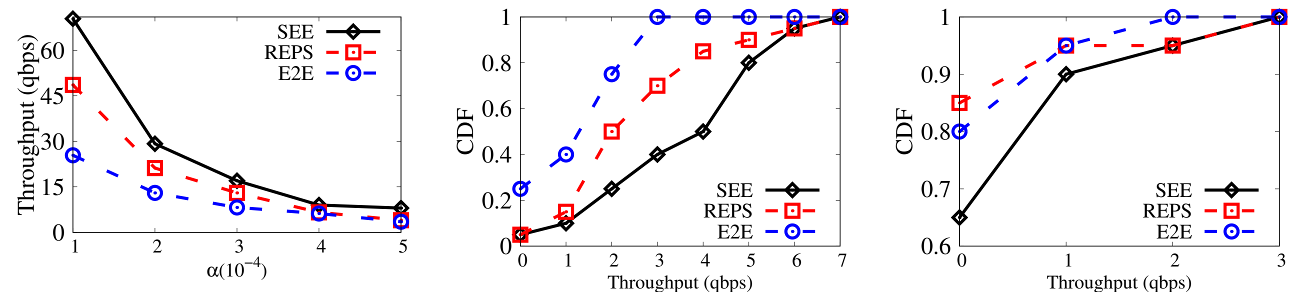
IV. Performance Evaluation

In this section, we evaluate the performance of SEE through extensive simulations using a custom in-house simulator built on Python. The LP solver used in our simulator is PuLP. Simulations involve randomly generated networks with a certain amount of quantum resources, a set of randomly chosen SD pairs and success probabilities of creating entanglement segments and quantum swapping. For the network throughput (measured in qubits per time slot, *i.e.*, qbps) shown in the simulations, each data is averaged by 100 trails, while the



(a) Link capacity vs. throughput (b) Throughput CDF when link capacity is 2. (c) Throughput CDF when link capacity is 7.

Fig. 3. How link capacity impacts network throughput.



(a) Success probability to create entanglement links vs. Throughput (b) Throughput CDF when α is 1×10^{-4} (c) Throughput CDF when α is 5×10^{-4}

Fig. 4. How the success probability to create entanglement segments impacts network throughput.

throughput CDFs, which show the throughput distributions among all SD pairs, are randomly picked up from one trail since the SD pairs and network topology in different trails are different so that we cannot show the CDF of average throughput. Accordingly, in the following figures, the sum of each SD pair's throughput (in Figs. x(b) & x(c)) is not equal to the network throughput (in Fig. x(a)).

A. Simulation Methodology

Network Topology Generation. As in [10], we randomly place a given number of nodes into a 10,000 km by 10,000 km square area. Quantum links are determined following the Waxman model [22]. On the generated topology, we prepare physical segments for every node pair with Yen's Algorithm [21]. The success probability to create an entanglement segment (u, v) over the k^{th} physical segment between node u and v is following [19]

$$p^k_{uv} = e^{-\alpha l^k_{uv}} + \delta \tag{2}$$

where l^k_{uv} is the length (measured in kilometers) of the corresponding segments and δ is a random variable uniformly distributed on $[-0.05, 0.05]$.

Default Parameters. In the default settings, there are 200 nodes and 20 SD pairs in the network. The success probability for quantum swapping is 0.9 [9, 10]; the number of quantum channels supported by each edge is 3; and the parameter that determines the success probability to create an entanglement link, *i.e.*, α in (2), is 0.0002, with which, the average external link success probability is about 0.8. By default, there are 10 units of quantum memory hosted by each quantum node.

Comparison Scheme. We compare SEE with two entanglement establishment schemes. One is REPS, which is the state-of-the-art technique. The other is to establish entanglement connections only by all-optical switching, which is labeled as E2E in all figures. In fact, REPS and E2E are the two extreme cases of SEE. The former one only uses the quantum swapping, while the later only uses all-optical switching.

B. Evaluation Results

Main observations: From our simulations, we observe that SEE outperforms REPS and E2E by up to 100% and 180%, respectively, in throughput. E2E performs the worst since it is difficult for us to establish an entanglement for a SD pair far away from each other. Compared with REPS, SEE can leverage all-optical switching to create longer entanglement segments and save the quantum memory resources. Though this will result in a smaller probability to create an entanglement segment, it enables us trying to create more entanglement segments. Even if it is not resource efficient to try to create entanglement segments over a multi-hop physical segments, it is still an option to create entanglement links and connects them via quantum swapping. Accordingly, SEE is the most feasible scheme to optimize the QDN throughput.

Effect of physical link capacity. We keep the default parameter settings, expect the capacity of each link varying from 2 to 7. The simulation results are shown in Fig. 3. From Fig. 3(a), we can observe that the SEE outperforms REPS and E2E by 27.27% – 55.17% and 58.62% – 180.00%, respectively. The network throughput increases with the link capacity regardless of which algorithm is adopted. This is intuitive since larger link capacity provides more resources to establish entanglement connections. However, when the capacity of each physical link exceeds 4, the network throughput will only slightly increase with the link capacity since the system bottleneck becomes the amount of quantum memory.

Figs. 3(b)&3(c) show the throughput CDF of all SD pairs when the link capacity is 2 and 7, respectively. From these figures, we can see that with SEE, more SD pairs will achieve higher throughput, and the largest throughput that can be achieved with SEE is also larger than other two algorithms. This coincides with the observation that SEE will achieve higher throughput than REPS and E2E.

Effect of entanglement segment success probability. To investigate how the success probability affects the performance of SEE, we vary the α in (2) from 1×10^{-4} to 5×10^{-4} and show the simulation results in Fig. 4. Generally, the larger the α is, the smaller the success probability it will be to create an entanglement segment, and so will the network throughput be. In Fig. 4(a), with the varying of the success probability to create an entanglement segment, SEE will achieve a network throughput 30.77% – 100.00% and 45.16% – 177.17% higher than that with REPS and E2E, respectively. Besides, with the decrease of the success probability to create an entanglement segment, the network throughput achieved by SEE decreases much faster than that achieved by other two algorithms, and finally, the performance of SEE will degrade to be the same as REPS. This is because that with the increase of α in (2), it will be more difficult to create an entanglement segment over a long segment. Therefore, fewer entanglement segments will be created via multi-hop physical segments. Thus, SEE will converge to solution similar to REPS. This is also verified in Figs. 4(b)&4(c). In Fig. 4(c), the throughput CDF curves of SEE and REPS is closer to each other than that in Fig. 4(b).

Effect of quantum swapping success probability. Fig. 5 shows how the quantum swapping success probability affects the performance of SEE. In Fig. 5(a), we can see that though the network throughput will increase with the quantum swapping success probability with SEE and REPS, the increase rate will be slower and slower. This is because when the swapping success probability is large enough, the main ingredient determines the network throughput is the number of entanglement segments (and the entanglement paths accordingly) that can be created. In addition, the quantum swapping success probability almost does not impact the network throughput with E2E since it does not use the quantum swapping to connect multiple entanglement segments. The most interesting observation is that when the success probability of quantum swapping is smaller than 0.6 in our simulations, the E2E outperforms REPS as it is difficult for REPS to connect entanglement segments with quantum swapping. In this case, all-optical switching would be the better option to establish long entanglements.

Effect of network scale. We evaluate the scalability of SEE by varying the number of nodes from 100 to 500. Fig. 6 shows how the throughput changes with the network scale. Generally, SEE outperforms RESP and E2E by 35.90% – 80.00% and 124.62% – 280.00%, respectively. The network throughput will become larger with the increase of the network scale. This is because that there will be more resources to support generating more entanglements and also we will be able to prepare more physical segments to create entanglement segments. Compare Fig. 6(b) with Fig. 6(c), we can see that with more resources and available physical segments to create entanglement segments, the throughput of each SD pair will also significantly increase. In a network with 100 nodes, an SD pair can establish at most 5 entanglement connections in each time slot, while some SD pairs can establish up to 10 entanglement connections in each time slot in a network with 500 nodes.

Effect of number of SD pairs. When the number of SD pairs in the network varies from 10 to 50, the throughput under different schemes are shown in Fig. 7. In Fig. 7, we can see that the network throughput first significantly increases with the number of SD pairs, and then increases in a slower pace. This is because that there are resource contentions among different SD pairs when the network suffers a heavy workload. However, from Figs. 7(b)& 7(c), we can see that the largest throughput that can be achieved by an SD pair will not significantly affected by the workload, since this is mainly determined by the maximum amount of resources that can be allocated to an SD pair, which is mainly determined by the network topology and has only slight relationship with the number of SD pairs in the network.

V. Conclusions

SEE is a framework which optimizes the throughput of Quantum Data Networks (QDNs) by deploying segmented entanglement establishment which integrates all-optical switching and quantum swapping. To the best of our knowledge, SEE is the first work that introduces segmented entanglement establishment into QDNs. We have formulated the throughput maximization problem and proposed efficient algorithms to slove it. Through extensive simulations, we demonstrate that SEE works well in networks with different features, *i.e.*, the success probability to create entanglement segments, the success probability to perform quantum swapping, the network scale, *etc.*., and preserves remarkable performance advantages over the quantum-swapping-only or all-optical-switching-only solutions.

References

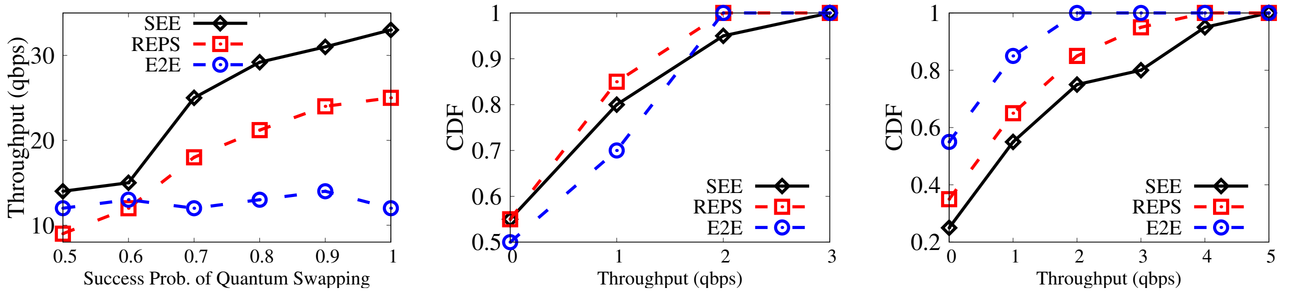
[1] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," in *Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing*, 1984.

[2] A. K. Ekert, "Quantum cryptography based on bell's theorem," *Physical review letters*, vol. 67, no. 6, pp. 661–663, 1991.

[3] C. Elliott, "Building the quantum network," *New Journal of Physics*, vol. 4, pp. 46.1–46.12, jul 2002.

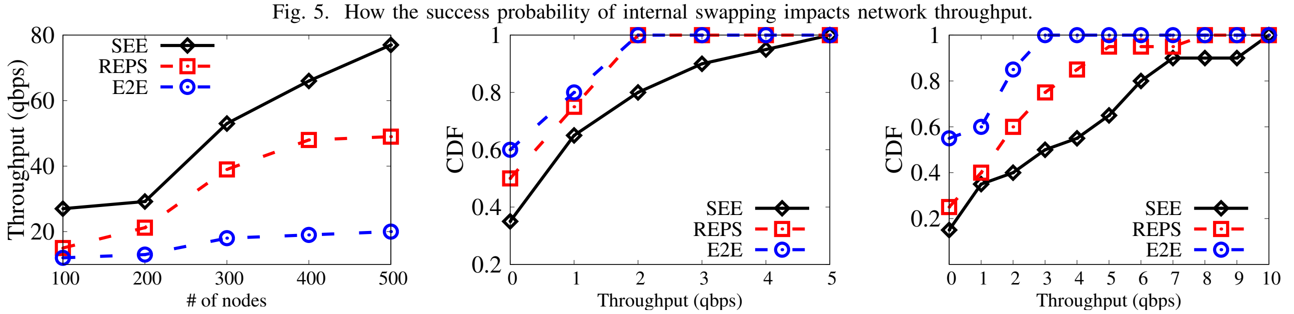
[4] M. Sasaki, M. Fujiwara, H. Ishizuka, W. Klaus, K. Wakui, M. Takeoka, S. Miki, T. Yamashita, Z. Wang, A. Tanaka, K. Yoshino, Y. Nambu, S. Takahashi, A. Tajima, A. Tomita, T. Domeki, T. Hasegawa, Y. Sakai, H. Kobayashi, T. Asai, K. Shimizu, T. Tokura, T. Tsurumaru, M. Matsui, T. Honjo, K. Tamaki, H. Takesue, Y. Tokura, J. F. Dynes, A. R. Dixon, A. W. Sharpe, Z. L. Yuan, A. J. Shields, S. Uchikoga, M. Legré, S. Robyr, P. Trinkler, L. Monat, J.-B. Page, G. Ribordy, A. Poppe, A. Allacher, O. Maurhart, T. Länger, M. Peev, and A. Zeilinger, "Field test of quantum key distribution in the tokyo qkd network," *Opt. Express*, vol. 19, no. 11, pp. 10 387–10 409, May 2011.

[5] M. Peev, C. Pacher, R. Alléaume, C. Barreiro, J. Bouda, W. Boxleitner, T. Debuisschert, E. Diamanti, M. Dianati, J. F. Dynes, S. Fasel, S. Fossier, M. Fürst, J.-D. Gautier, O. Gay, N. Gisin, P. Grangier, A. Happe, Y. Hasani, M. Hentschel, H. Hübel, G. Humer, T. Länger, M. Legré, R. Lieger, J. Lodewyck, T. Lorünser, N. Lütkenhaus, A. Marhold, T. Matyus, O. Maurhart, L. Monat, S. Nauerth, J.-B. Page, A. Poppe, E. Querasser, G. Ribordy, S. Robyr, L. Salvail, A. W. Sharpe, A. J.



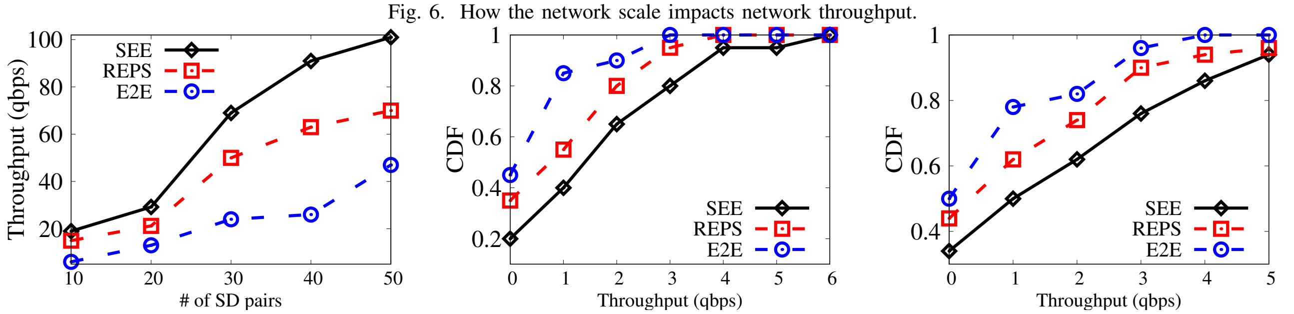
(a) Success probability of quantum swapping vs. throughput.
(b) Throughput CDF when quantum swapping success probability is 0.5.
(c) Throughput CDF when quantum swapping success probability is 1.

Fig. 5. How the success probability of internal swapping impacts network throughput.



(a) Network scale vs. Throughput
(b) Throughput CDF when there are 100 nodes.
(c) Throughput CDF when there are 500 nodes.

Fig. 6. How the network scale impacts network throughput.



(a) Workload vs. Throughput
(b) Throughput CDF with 20 SD pairs.
(c) Throughput CDF with 50 SD pairs.

Fig. 7. How the workload impacts network throughput.

Shields, D. Stucki, M. Suda, C. Tamas, T. Themel, R. T. Thew, Y. Thoma, A. Treiber, P. Trinkler, R. Tualle-Brouri, F. Vannel, N. Walenta, H. Weier, H. Weinfurter, I. Wimberger, Z. L. Yuan, H. Zbinden, and A. Zeilinger, "The SECOQC quantum key distribution network in vienna," *New Journal of Physics*, vol. 11, no. 7, pp. 1–37, jul 2009.

[6] J. Yin, Y. Cao, Y.-H. Li, S.-K. Liao, L. Zhang, J.-G. Ren, W.-Q. Cai, W.-Y. Liu, B. Li, H. Dai, G.-B. Li, Q.-M. Lu, Y.-H. Gong, Y. Xu, S.-L. Li, F.-Z. Li, Y.-Y. Yin, Z.-Q. Jiang, M. Li, J.-J. Jia, G. Ren, D. He, Y.-L. Zhou, X.-X. Zhang, N. Wang, X. Chang, Z.-C. Zhu, N.-L. Liu, Y.-A. Chen, C.-Y. Lu, R. Shu, C.-Z. Peng, J.-Y. Wang, and J.-W. Pan, "Satellite-based entanglement distribution over 1200 kilometers," *Science*, vol. 356, no. 6343, pp. 1140–1144, 2017.

[7] J. L. Park, "The concept of transition in quantum mechanics," *Foundations of Physics*, vol. 1, p. 23–33, 1970.

[8] N. Jiang, Y.-F. Pu, W. Chang, C. Li, S. Zhang, and L.-M. Duan, "Experimental realization of 105-qubit random access quantum memory," *npj Quantum Information*, vol. 5, no. 28, pp. 1–6, 2019.

[9] S. Shi and C. Qian, "Concurrent entanglement routing for quantum networks: Model and designs," in *ACM SIGCOMM*, 2020.

[10] Y. Zhao and C. Qiao, "Redundant entanglement provisioning and selection for throughput maximization in quantum networks," in *Proceedings of the IEEE INFOCOM*, 2021.

[11] W. Kozlowski, A. Dahlberg, and S. Wehner, "Designing a quantum network protocol," in *ACM CoNEXT*, 2020, pp. 1 – 16.

[12] L. J. Stephenson, D. P. Nadlinger, B. C. Nichol, S. An, P. Drmota, T. G. Ballance, K. Thirumalai, J. Goodwin, D. M. Lucas, and C. J. Ballance, "High-rate, high-fidelity entanglement of qubits across an elementary quantum network," *Physics Review Letters*, 2020.

[13] A. Dahlberg, M. Skrzypczyk, T. Coopmans, L. Wubben, F. Rozpundefineddek, M. Pompili, A. Stolk, P. Pawelczak, R. Knegjens, J. de Oliveira Filho, R. Hanson, and S. Wehner, "A link layer protocol for quantum networks," in *ACM SIGCOMM*, 2019, pp. 159—-173.

[14] S. Pirandola, "End-to-end capacities of a quantum communication network," *Communications Physics*, vol. 3, pp. 1—-10, 2019.

[15] E. Schoute, L. Mancinska, T. Islam, I. Kerenidis, and S. Wehner, "Shortcuts to quantum network routing," *CoRR*, vol. abs/1610.05238, 2016. [Online]. Available: http://arxiv.org/abs/1610.05238

[16] 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.

[17] M. Caleffi, "Optimal routing for quantum networks," *IEEE Access*, vol. 5, pp. 22 299–22 312, 2017.

[18] R. V. Meter, T. Satoh, T. D. Ladd, W. J. Munro, and K. Nemoto, "Path selection for quantum repeater networks," *Networking Science*, vol. 3, pp. 82–95, 2013.

[19] M. Pant, H. Krovi, D. Towsley, L. Tassiulas, L. Jiang, P. Basu, D. Englund, and S. Guha, "Routing entanglement in the quantum internet," *npj Quantum Information*, vol. 5, Dec. 2019.

[20] L. Brunetta, M. Conforti, and M. Fischetti, "A polyhedral approach to an integer multicommodity flow problem," *Discrete Applied Mathematics*, vol. 101, no. 1, pp. 13 – 36, 2000.

[21] J. Y. Yen, "An algorithm for finding shortest routes from all source nodes to a given destination in general networks," *Quarterly of Applied Mathematics*, vol. 27, pp. 526–530, 1970.

[22] B. M. Waxman, "Routing of multipoint connections," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 9, pp. 1617–1622, 1988.