Authorizable Tripartite Entanglement Routing via 3-GHZ State in Quantum Networks

Shao-Min Huang[†‡], Ching-Ting Wei[†§], Kai-Xu Zhan[†§], Juliette Chou Le Touze[†], Jian-Jhih Kuo[†*], Chih-Yu Wang[‡]
[†]Dept. of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan
[‡]Research Center for Information Technology Innovation, Academia Sinica, Taipei City, Taiwan

***Abstract*—Traditional end-to-end entanglement typically operates between two parties. However, such a setup may fall short when three parties are involved. GHZ states, a multi-qubit entangled state, provide an approach to such a problem. A fusion node is first chosen, and then the paths from all end nodes to the fusion node are fused to create an entangled state between all parties. The selection of the fusion node becomes critical since it highly affects the success probability of the whole process. Thus, in this paper, we explore the promising scenario for multiple requests of authorizable tripartite teleportation and introduce a novel optimization problem to maximize the (expected) total profit for 3-GHZ requests. Via extensive simulation results, we show that our algorithm can outperform existing approaches significantly.**

I. Introduction

Quantum networks (QNs) facilitates secure data transmission and enables innovative applications [1]. Traditional end-to-end entanglement typically operates between two parties, such as sending data qubits via quantum teleportation from Alice to Bob. However, such a setup may not suffice for scenarios involving three parties (i.e., tripartite), e.g., quantum secret sharing (QSS) and quantum information splitting (QIS) [2]. Multi-qubit entangled states [3], such as GHZ states, W states, or cluster states, are being considered to support the scenarios.

Fig. 1 shows an example, where Alice intends to send a data qubit (depicted as a green circle) to Bob, and this process requires the authorization of Jacky [2]. We first establish a 3-GHZ state, with each of the three participants holding one of the entangled qubits (depicted as red circles). Second, Alice performs a Bell state measurement on her two qubits and informs Jacky and Bob about the Bell pair basis she used. The 3-GHZ state collapses into a two-qubit entangled state between Jacky and Bob, containing information about the data qubit. The data qubit's information is split into two halves and stored in two qubits, held by Jacky and Bob, respectively. This ensures that Bob cannot access the complete information without Jacky's authorization, and vice versa. Third, Jacky measures his qubit, informs Bob about the basis he measured, and Bob adjusts his qubit accordingly to retrieve the data qubit.

The former authorizable entanglement teleportation scenario involving three parties based on the 3-GHZ entangled state has not been investigated in depth in the literature. A 3-GHZ state can be constructed in three steps [4], as illustrated in Fig.

[§]: equal contributions;*: corresponding author (lajacky@cs.ccu.edu.tw)

This work was supported by the National Science and Technology Council under Grants 111-2628-E-001-002-MY3, 111-2628-E-194-001-MY3, 112-2218-E-194-005, 113-2221-E-194-040-MY3, and the Academia Sinica under Thematic Research Grant AS-TP-110-M07-3 in Taiwan.

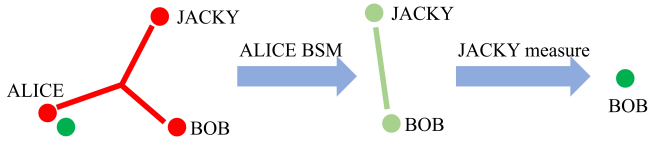
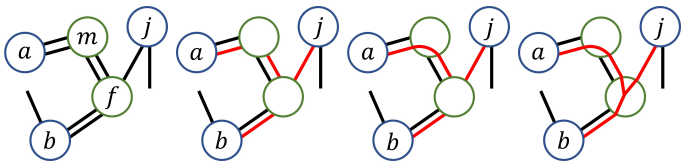


Fig. 1. Authorizable quantum teleportation process.

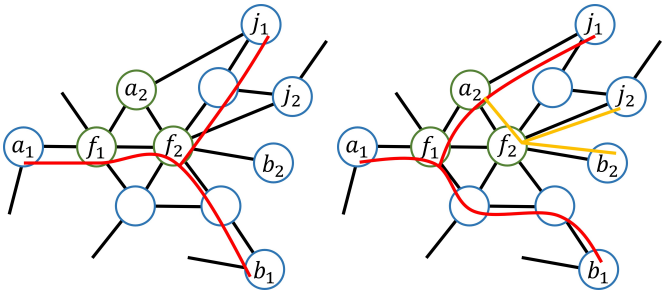


(a) QN (b) Entangling (c) swapping (d) Fusing

Fig. 2. Step-by-step generation of a three-pointed star.

2. Initially, in a QN, quantum nodes (circles) are connected by limited quantum channels (black lines) for quantum information transmission in the form of *qubits*, and each quantum node has a specific amount of *quantum memory* to store qubits and avoid rapid decoherence [5]. For ease of reading, the green (or blue) nodes have three (or two) units of memory in Figs. 2 and 3. We aim to establish a 3-GHZ state between nodes a (sender), b (receiver), and j (authorizer). First, in the entangling phase, adjacent quantum nodes can generate Bell pairs (depicted by red lines in Fig. 2(b)). Each pair consumes one unit of quantum memory on each of the two adjacent nodes and requires one quantum channel between them. Second, in the swapping phase, entanglement swapping is performed at the intermediate node (e.g., node m in Fig. 2(c)) to merge two entangled links into a longer path, connecting each end node with the fusion node f. Third, in the fusion phase, the quantum node (e.g., node f in Fig. 2(d)) fuses three paths into a *three-pointed star*. After that, a 3-GHZ state is generated. Nevertheless, the generation may fail if any of these steps fail. The success probability of creating a three-pointed star depends on the distance between nodes and the adopted technologies.

Selecting the fusion node is a crucial problem in 3-GHZ quantum networking. Each fusion node requires three units of quantum memory for a request, which can be more stringent when facing multiple requests simultaneously. Furthermore, the selection of fusion nodes impacts the subsequent selection of three-pointed stars, implying different success probabilities. Consider the scenario in Fig. 3, where we have two requests: $r_1 = (a_1, b_1, j_1)$ and $r_2 = (a_2, b_2, j_2)$, and node a_2, f_1, f_2 have three memory units, making them feasible fusion node



(a) Scenario 1 (b) Scenario 2

Fig. 3. Multi-request scenario.

candidates. Suppose nodes have equal swapping and fusing probabilities. In Fig. 3(a), r_1 chooses f_2 as the fusion node to form the red three-pointed star with the shorter distance sum, implies the higher probability. However, this choice occupies two green nodes, rendering r_2 unable to find a feasible solution. By contrast, in Fig. 3(b), r_1 selects f_1 as the fusion node, allowing r_2 to choose f_2. Then, r_1 and r_2 obtain red and orange three-pointed stars, respectively. One may observe that although Fig. 3(b) is a suboptimal choice for r_1 in terms of success probability, it allows the co-existence of r_2. Thus, Fig. 3(b) is favorable if we intend to maximize concurrent requests; however, it is worth noting that requests may have different rewards if succeeding. If r_1 has a dominant reward compared to r_2, Fig. 3(a) might actually be the better choice.

From the observations above, several challenges emerge: 1) *Fusion node selection and resource allocation*. The key of fusion node selection is its impact on resource consumption. Opting for a suboptimal fusion node may result in a three-pointed star with higher hop counts and resource consumption. For example, in Fig. 2, we designate node f as the fusion node, resulting in quantum memory consumption on nodes $(a, b, j, m, f) = (1, 1, 1, 2, 3)$ and quantum channel consumption on edges $(\overline{am}, \overline{mf}, \overline{fb}, \overline{fj}) = (1, 1, 1, 1)$. Alternatively, if we select node m as the fusion node, the paths from each party to the source become $\{(a, m), (j, f, m), (b, f, m)\}$, overlapping at (f, m). This choice leads to increases in quantum memory consumption on nodes m and f and quantum channel consumption of on edge $\overline{mf}$, i.e., $(a, b, j, m, f) = (1, 1, 1, 3, 4)$ and $(\overline{am}, \overline{mf}, \overline{fb}, \overline{fj}) = (1, 2, 1, 1)$). This is less efficient than selecting node f. Furthermore, a suboptimal three-pointed star may occupy critical resources that could be valuable for other requests, as depicted in Fig. 3(b). 2) *Fusion node selection and success probability*. The selection of the fusion node significantly impacts the success probability of the three-pointed star. Entangling and swapping play a major role in the final success probability, which corresponds to the total distance and hop count on the path. Unlike bipartite entanglement problems that involve only two parties (Alice and Bob), our scenario necessitates considering three parties (Alice, Bob, and Jacky). Biasing the fusion node might cause low success probability. For instance, in Fig. 3(b), selecting f_1 biases toward a_1 leads to a longer total entanglement distance and, consequently, lower probability. 3) *Request Prioritizing*. Different requests contain varying rewards, but resource contention frequently arises. Also, different path selections result in different probabilities and expected rewards. Thus, prioritizing each request is hard since estimating the cost-efficient ratio is non-trivial.

To this end, we present a promising network architecture for multiple authorizable tripartite teleportation requests and introduce a novel optimization problem termed the Authori**z**abl**e** concur**r**ent entanglement r**o**uting (ZERO): we aim to prioritize requests that require a connection via a 3-GHZ entangled state to maximize the total reward as each request carries its own reward. We first provide our system model's overview and formulate the ZERO in Section II. Due to the page limit, the related works are discussed in Appendix A of [6]. To tackle the ZERO, we propose a bi-criteria approximation algorithm in Section III. We devise a non-trivial separation oracle for the dual linear programming (LP) and employ a primal-dual algorithm (PDA) to obtain a fractional solution with a bounded error to address the possibly exponential number of three-pointed stars for each request. Then, a randomized LP rounding technique and a heuristic algorithm are applied to achieve the approximation ratio and further refine the solution. Section IV shows that our algorithm outperforms existing approaches by up to 103%. Finally, Section V concludes this paper.

II. System Model and Problem Formulation

A. Network Architecture and System Model

A QN comprises multiple quantum nodes, each possessing limited quantum memory and connected to neighboring nodes via a finite number of quantum channels [7]. Quantum nodes can serve as different roles to fulfill different requests concurrently: They can be senders (e.g., Alice), receivers (e.g., Bob), authorizers (e.g., Jacky), repeaters (e.g., node m in Fig. 1), and fusion nodes (e.g., node f in Fig. 1). During the entangling process, two nodes, u and v, consume one unit of their quantum memory to generate an entangled pair through the quantum channel. The success probability of entangling decreases exponentially with the distance of the quantum channel due to signal loss, i.e., $P_e(u, v) = e^{-\tau \cdot l(u,v)}$, where $l(u, v)$ is the length of the quantum channel between u and v, and τ is a constant determined by the optical fiber material [5]. Besides, the swapping process at node u can establish logical swapping links to merge two entangled links (paths) of u into a single entangled path with the success probability $P_s(u)$. Consequently, the success probability of constructing an entangled path p is $\Pr(p) = \prod_{(u,v)\in p} P_e(u, v) \cdot \prod_{v\in p\setminus\{y,z\}} P_s(v)$, where y and z denote the two end nodes of path p. Similarly, a fusion process at node u can combine three entangled paths of u into a three-pointed star configuration with the success fusion probability $P_f(u)$. Thus, the success probability of constructing a three-pointed star T with fusion node f is $\Pr(T) = P_f(f) \cdot \prod_{p\in T} \Pr(p)$, where the star T contains the three paths $p_{a,f}$, $p_{b,f}$, and $p_{j,f}$, and a, b, and j are the sender, receiver, and authorizer, respectively.

In a QN, to ensure effective coordination among all nodes, a central controller is deployed to periodically gather information regarding the outcomes of entanglement, swapping, and fusion processes on each link and node [7]. To earn the corresponding

profit, the data qubit of each request should be transmitted to the designated receiver via an authorizer. Thus, efficient resource allocation and profit maximization are vital optimization objectives in QNs, fostering a novel optimization problem termed Authori**z**ab**l**e concur**r**ent entanglement r**o**uting (ZERO).

B. Problem Formulation – ZERO

An undirected network $G = (V, E)$, where each edge $e \in E$ has a channel capacity $c(e) \in \mathbb{Z}^+$ and a success probability of entangling $P_e(e) \in (0, 1]$. Also, each node $v \in V$ has a memory limit $m(v) \in \mathbb{Z}^+$ and the success probabilities of swapping and fusion, i.e., $P_s(v) \in (0, 1]$ and $P_f(v) \in (0, 1]$ [4]. Let R be the set of requests. Each request $r \in R$ asks to construct a 3-GHZ state. Besides, each request $r \in R$ has its profit $\mathcal{O}(r) \in \mathbb{R}^+$. The ZERO aims to allocate the network resources for the requests to maximize the expected total profit for all requests, subject to the following constraints:

1) For each quantum node v, the total amount of quantum memory used on v does not exceed $m(v)$.
2) For each edge e, the total number of qubits transmitted on e does not exceed $c(e)$.
3) For each request r, at most one three-pointed star is constructed to handle the request.

Let $T(r)$ be the set of all possible three-pointed stars for each request $r \in R$, and let decision variable $x_T^r \in \{0, 1\}$ represent whether to choose the three-pointed star $T \in T(r)$ for each request $r \in R$. Then, the integer linear programming (ILP) of the ZERO is presented as follows.

$$\max \sum_{r\in R} \sum_{T\in T(r)} \Pr(T) \cdot \mathcal{O}(r) \cdot x_T^r \tag{1a}$$

$$\text{s.t.} \sum_{r\in R} \sum_{u\in V} \sum_{T\in T(r):(u,v)\in T} x_T^r \le m(v), \quad \forall v \in V \tag{1b}$$

$$\sum_{r\in R} \sum_{T\in T(r):e\in T} x_T^r \le c(e), \quad \forall e \in E \tag{1c}$$

$$\sum_{T\in T(r)} x_T^r \le 1, \quad \forall r \in R \tag{1d}$$

$$x_T^r \in \{0, 1\} \quad \forall r \in R, \forall T \in T(r) \tag{1e}$$

The objective (1a) is maximizing the expected total profit for all requests. Constraints (1b) and (1c) show the limits of node memory and link channels, respectively. Constraint (1d) guarantees that at most a three-pointed star $T \in T(r)$ is constructed for each request $r \in R$.

The NP-hardness proof of ZERO is omitted and provided in Appendix B of the technical report [6] due to the page limit.

Theorem 1. The ZERO is NP-hard.

III. Bi-criteria Approximation Algorithm

To efficiently solve the ZERO, we attempt to design an approximation algorithm based on LP relaxation and randomized rounding. However, solving the relaxed LP (i.e., constraints $x_T^r \in \{0, 1\}$ are relaxed to $x_T^r \in [0, 1]$) in polynomial time is still challenging due to the potential exponential number of decision variables. By [8], for a standard-form LP with a polynomial number of constraints (excluding non-negativity constraints on variables), its dual LP also has a polynomial number of variables. In this case, solving its dual LP in polynomial time becomes possible if there exists a separation oracle to identify the most deviated constraint among an exponential number of dual constraints. On the other hand, by [9], a near-optimal solution for the primal LP can be further acquired in polynomial time by a PDA if the following conditions are met:

1) In every constraint of the primal LP, except for the non-negativity constraints of the variables, each coefficient on the left-hand side of each inequality is less than or equal to the constant on the right-hand side.
2) In every constraint of the dual LP, except for the non-negativity constraints of the variables, all coefficients on the left-hand side and the constant on the right-hand side of each inequality are positive.
3) A *separation oracle* exists for the dual LP to identify the (most) deviated constraint efficiently.

Definition 1. The form of an LP is said to be standard if its form is $\max\{c^T x | Ax \le b, x \ge 0\}$, where x is an $n \times 1$ variable vector, and c, b, and A denote $n \times 1$, $m \times 1$, $m \times n$ constant vectors, respectively. In this case, its dual LP is $\min\{b^T y | A^T y \ge c, y \ge 0\}$, where y is an $m \times 1$ variable vector. Remark that $x \ge 0$ and $y \ge 0$ denote the non-negativity constraints of variables x and y in the primal and dual LP.

To relax the primal LP, we replace constraint (1e) with:

$$x_T^r \ge 0, \qquad \forall r \in R, \forall T \in T(r). \tag{2}$$

Then, the primal LP (1a)−(1d), (2) becomes standard form with a polynomial number of constraints (excluding the non-negativity constraints for variables) and also satisfies the first condition. Accordingly, its dual LP with the dual variables $\alpha_v, \beta_e, \gamma_r$ is derived as follows, meeting the second condition.

$$\min \sum_{v\in V} m(v) \cdot \alpha_v + \sum_{e\in E} c(e) \cdot \beta_e + \sum_{r\in R} \gamma_r \tag{3a}$$

$$\text{s.t.} \sum_{(u,v)\in T} (\alpha_u + \alpha_v) + \sum_{e\in T} \beta_e + \gamma_r \ge \Pr(T) \cdot \mathcal{O}(r), \quad \forall r \in R, \forall T \in T(r) \tag{3b}$$

$$\alpha_v, \beta_e, \gamma_r \ge 0, \quad \forall v \in V, \forall e \in E, \forall r \in R \tag{3c}$$

We then design a separation oracle of the dual LP to satisfy the third condition for using the PDA in Section III-A. The PDA iteratively finds a three-pointed star T by the separation oracle and increases its decision variable $\hat{x}_T^r$. Afterward, Section III-B develops a randomized algorithm to round the fractional solution $\hat{x}_T^r$, making all requests have an opportunity to compete for the resources, effectively addressing the third challenge in the ZERO. Last, we provide a heuristic algorithm in Section III-C to refine the rounded solution.

A. The Separation Oracle

Given an arbitrary fractional solution (α, β, γ) of the dual LP, a separation oracle should be able to return the most

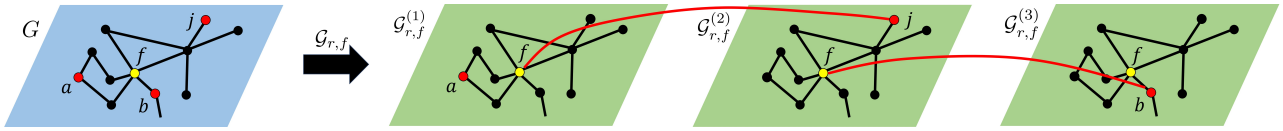


Fig. 4. An example of the construction for the auxiliary graph $\mathcal{G}_{r,f}$ with a specific fusion node f for request r.

deviated constraint. It is easy to find a deviated constraint in (3c) in polynomial time if it exists. However, identifying whether a deviated constraint exists in (3b) is difficult since there are exponential inequalities.

Note that since the number of requests is polynomial, we can focus on finding the most deviated constraint for a specific request $r \in R$. Moreover, fusion node f can be located at one of $|V|-3$ positions, while $\mathcal{O}(r)$ is a constant. Thus, the computing of separation oracle can narrow down to:

$$\min_{T_f \in T(r)} \frac{\sum_{(u,v)\in T_f}(\alpha_u+\alpha_v)+\sum_{e\in T_f}\beta_e+\gamma_r}{\Pr(T_f)}, \tag{4}$$

where T_f denotes a three-pointed star with the fusion node at f. Nevertheless, computing Eq. (4) for a given request r with a specific fusion node f is still non-trivial since the coefficient $\frac{1}{\Pr(T_f)}$ is associated with each three-pointed star in $T(r,f) = \{T \in T(r) \mid f \in T\}$. It is impractical to examine every star in $T(r,f)$ since the size of $T(r,f)$ is exponential.

To efficiently find the desired three-pointed star, we construct an auxiliary undirected edge-weighted graph $\mathcal{G}_{r,f}$ such that every feasible three-pointed star $T \in T(r,f)$ corresponds to a path from a virtual node $\tilde{s}_{r,f}$ to another one $\tilde{d}_{r,f}$ in $\mathcal{G}_{r,f}$.

Initially, we replicate every node and edge in the network G for three times, say $\mathcal{G}^{(1)}_{r,f}$, $\mathcal{G}^{(2)}_{r,f}$, and $\mathcal{G}^{(3)}_{r,f}$, then incorporate them into $\mathcal{G}_{r,f}$. Let a, b, and j denote the three end nodes of request r in G. Then, we create a set V_{end} and make it include the replica of a in $\mathcal{G}^{(1)}_{r,f}$, the replica of j in $\mathcal{G}^{(2)}_{r,f}$, and the replica of b in $\mathcal{G}^{(3)}_{r,f}$, as shown by the red nodes in Fig. 4. Similarly, a set V_{fus} is created to include the three replicas of f in $\mathcal{G}^{(1)}_{r,f}$, $\mathcal{G}^{(2)}_{r,f}$, and $\mathcal{G}^{(3)}_{r,f}$, as shown by the yellow nodes in Fig. 4. Then, we create two edges and include them into $\mathcal{G}_{r,f}$ for the connectivity, as shown by red edges in Fig. 4. One connects the replica of f in $\mathcal{G}^{(1)}_{r,f}$ and the replica of j in $\mathcal{G}^{(2)}_{r,f}$, and the other is incident with the replica of f in $\mathcal{G}^{(2)}_{r,f}$ and the replica of b in $\mathcal{G}^{(3)}_{r,f}$.

Note that each edge $\tilde{e} \in \mathcal{G}^{(1)}_{r,f} \cup \mathcal{G}^{(2)}_{r,f} \cup \mathcal{G}^{(3)}_{r,f}$ has a corresponding edge e between nodes u and v in G. Then, the dual cost and success probability of each edge $\tilde{e}$ between node $\tilde{u}$ and node $\tilde{v}$ in $\mathcal{G}_{r,f}$ can be set by the following formulas, respectively.

$$\mathcal{W}(\tilde{e}) = \begin{cases} \alpha_u+\alpha_v+\beta_e & \text{if } \tilde{e} \in \mathcal{G}^{(k)}_{r,f},\ k=\{1,2,3\} \\ \frac{\gamma_r}{2} & \text{otherwise.} \end{cases}$$

$$\mathcal{P}(\tilde{e}) = \begin{cases} P_e(e)\cdot \Pr(\tilde{u})\cdot \Pr(\tilde{v}) & \text{if } \tilde{e} \in \mathcal{G}^{(k)}_{r,f},\ k=\{1,2,3\} \\ 1 & \text{otherwise,} \end{cases}$$

$$\text{where } \Pr(\tilde{u}) = \begin{cases} 1 & \text{if } \tilde{u} \in V_{end} \\ \sqrt[3]{P_f(u)} & \text{else if } \tilde{u} \in V_{fus} \\ \sqrt{P_s(u)} & \text{otherwise.} \end{cases}$$

Then, $\tilde{s}_{r,f}$ is set to the replica of a in $\mathcal{G}^{(1)}_{r,f}$, and $\tilde{d}_{r,f}$ is set to the replica of f in $\mathcal{G}^{(3)}_{r,f}$. With this clever setting, every three-pointed star with fusion node f for request r in G can be mapped to a path from $\tilde{s}_{r,f}$ to $\tilde{d}_{r,f}$ in $\mathcal{G}_{r,f}$. Let $P_{\mathcal{G}_{r,f}}(\tilde{s}_{r,f},\tilde{d}_{r,f})$ be the set of all possible paths from $\tilde{s}_{r,f}$ to $\tilde{d}_{r,f}$ in $\mathcal{G}_{r,f}$. Computing Eq. (4) is equivalent to computing:

$$\min_{p\in P_{\mathcal{G}_{r,f}}(\tilde{s}_{r,f},\tilde{d}_{r,f})} \frac{\sum_{e\in p}\mathcal{W}(e)}{\prod_{e\in p}\mathcal{P}(e)}, \tag{5}$$

Nevertheless, using the classical Dijkstra algorithm to solve Eq. (5) is infeasible due to the absence of the monotonicity property in the criterion [10], [11]. To tackle these challenges, we adopt an bi-objective shortest path algorithm [10], [11], which takes into account the impact of both the numerator and denominator (i.e., bi-objective) in Eq. (5). Based on [10], [11], the bi-objective shortest path can be acquired in polynomial time in expectation if the following properties are satisfied:

1) The first- and second-objective cost functions are additive. That is, the cost of a path is the cost sum of its edge.
2) The bi-objective total cost function is quasiconcave.

In other words, we may rewrite the objective function in Eq. (5) as a bi-objective total cost function $U(x(p),y(p))$, where $x(p)$ and $y(p)$ represent the first- and second-objective cost functions of a path p in $\mathcal{G}_{r,f}$, respectively. It then becomes the minimization problem with the bi-objective cost function:

$$\min_{p\in P_{\mathcal{G}_{r,f}}(\tilde{s}_{r,f},\tilde{d}_{r,f})} U(x(p),y(p)) \coloneqq x(p)\cdot e^{y(p)}, \tag{6}$$

where $x(p)$ and $y(p)$ are defined by summing the corresponding costs of its edges in $\mathcal{G}_{r,f}$ as follows:

$$x(p) = \sum_{e\in p}\mathcal{W}(e) \text{ and } y(p) = \sum_{e\in p} -\ln \mathcal{P}(e). \tag{7}$$

In this way, both cost functions $x(p)$ and $y(p)$ exhibit additivity, conform to the first property. Moreover, Lemma 1 guarantees that $U(x,y)$ is quasiconcave under the condition $x,y>0$, satisfying the second property. The proof of Lemma 1 is presented in Appendix C of [6] due to the page limit. Thus, the path $p \in P_{\mathcal{G}_{r,f}}(\tilde{s}_{r,f},\tilde{d}_{r,f})$ that minimizes the cost function $U(x(p),y(p))$ can be obtained and transformed to a three-pointed star $T \in T(r,f)$, representing the most deviated constraint for a specific pair of r and f in (3b). Remark that, the cost $x(p)$ reflexes the memory load, channel load, and demand satisfaction, while the cost $y(p)$ exhibits the probability $\Pr(T)$. Thus, using a three-pointed star with lower corresponding $U(x,y)$ for each request can achieve better resources allocation to address the first and second challenges of the ZERO simultaneously.

Lemma 1. $U(x,y)$ is quasiconcave when $x,y>0$.

Note that for every path $p \in P_{\mathcal{G}_{r,f}}(\tilde{s}_{r,f}, \tilde{d}_{r,f})$, its first- and second-objective costs can be mapped to a point on a 2D plane. The second property ensures that the mapped point of the bi-objective shortest path must be located on the convex hull of all mapped points. Examining the points on the convex hull (i.e., extreme paths) one by one to find the bi-objective shortest path can be accomplished by iteratively applying the Dijkstra algorithm [10], [11]. As the expected number of the extreme paths is $O(|V|^{0.5})$ [10], finding the bi-objective shortest path can be completed in polynomial time in expectation.[1]

B. Randomized Rounding Algorithm

We acquire the solution of the ZERO by rounding the near-optimal fractional solution $\hat{x}_T^r$ in a randomized manner. Let $\mathbb{T}(r)$ denote the set of three-pointed stars with $\hat{x}_T^r > 0$ for each request $r \in R$. It is worth noting that the size of $\mathbb{T}(r)$ is polynomial since the PDA terminates in polynomial time [9]. The value of $\hat{x}_T^r$ is interpreted as the probability of selecting a three-pointed star for each request r. Consider an example with $\mathbb{T}(r) = \{T_1, T_2, T_3\}$ for a request r. Assume that the given solution is $\hat{x}_{T_1}^r = 0.3$, $\hat{x}_{T_2}^r = 0.4$, and $\hat{x}_{T_3}^r = 0.1$. Thus, the probabilities of selecting three-pointed star T_1, T_2, and T_3 are 0.3, 0.4, and 0.1, respectively, while the probability of selecting no path set for request r is $1 - 0.3 - 0.4 - 0.1 = 0.2$.

The bi-criteria approximation ratio of the algorithm for the ZERO is $(O(1), O(\log |V|))$. Due to the page limit, the proof of Theorem 2 is detailed in Appendix D of [6].

Theorem 2. The proposed randomized algorithm for the ZERO is a $(O(1), O(\log |V|))$-approximation algorithm.

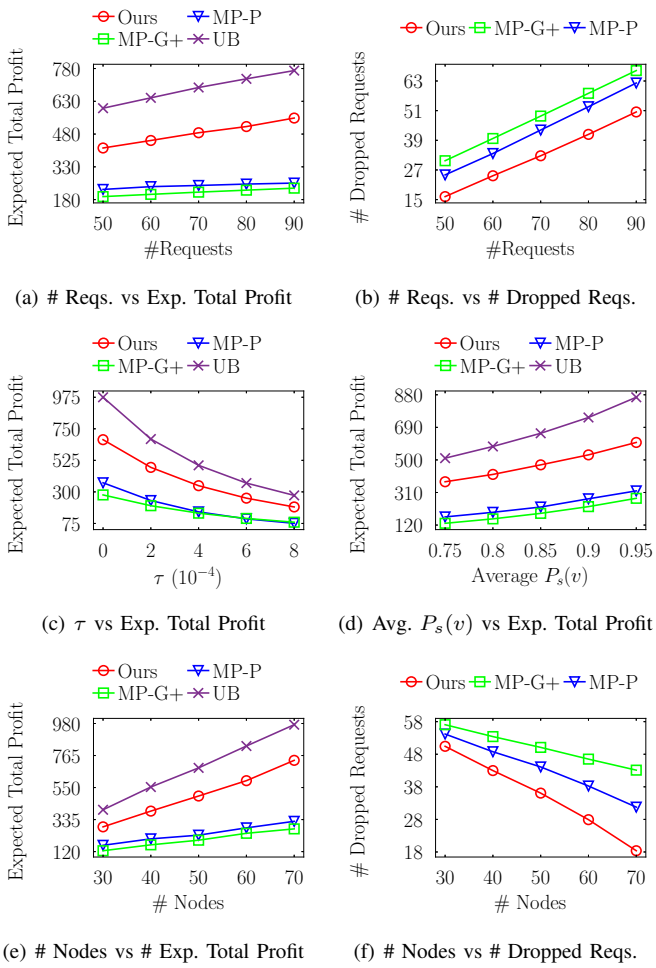
C. Heuristic Algorithm for Solution Improvement

As the rounded solution by randomized rounding may deviate the memory and channel limits, we propose a heuristic algorithm to improve the solution. First, we sort the nodes and links in non-increasing order of their load. Then, for each node (or link) in this order, we iteratively remove the three-pointed star with the lowest profit until the limit of the node (or link) is not deviated. After that, we sort the requests with no three-pointed star allocated in non-increasing order of their profit. We then iteratively examine each request whether there is a three-pointed star $T \in \mathbb{T}$ can be accommodated in the residual graph. If so, the request is served by the three-pointed star $T \in \mathbb{T}$ with the maximum $\Pr(T)$. The heuristic algorithm will terminate if no further request can be served.

IV. Performance Evaluation

A. Simulation Settings

We compare our algorithm (Ours) with the MP-P [13] and the MP-G+ [13]. Following [1], [7], we use the Waxman model to generate networks. Each simulation result varies by only one parameter (i.e., independent variable), and the default parameters are set as follows. We randomly deploy 50 nodes within an 1000 km $\times$ 2000 km area for each network, where the parameters of the Waxman model, δ and ϵ, are set to 0.85 and 0.4, respectively. An edge would exist with the probability $\delta e^{-l(u,v)/\epsilon L}$, where L is the farthest distance between any two nodes in the network, so the average edge length is around 300 km. The memory limit of each node and the channel limit of each edge are randomly sampled in $[10, 14]$ and $[3, 7]$, respectively. Besides, τ is set to 0.0002, and thus the average success probability of entangling $P_e(u, v)$ is 0.94, while the success probability of swapping $P_s(v)$ and fusion $P_f(v)$ ranges $[0.8, 0.95]$ and $(0, 1]$, respectively. Next, we randomly generate 70 3-GHZ requests and assign each request's profit $\mathcal{O}(r)$ based on its willingness to pay, which is usually related to its resource utility [14]. Thus, we assign each request's willingness to pay based on the minimal resource consumption of constructing a three-pointed star for each request. However, resource scarcity may lead to competition among users, and auction prices of customers typically follow a normal distribution [15]. Therefore, we set the ratio of willingness to pay randomly between 1 and 3 by the right half part of the normal distribution model with the median of 1 and standard deviation of 1.5. Finally, the key metric "Expected Total Profit" is the sum of expected profit of requests with sufficient allocated resources. Each result is averaged over 60 trials. We conduct an ablation study to show the effectiveness of separation oracle in Appendix E of [6].



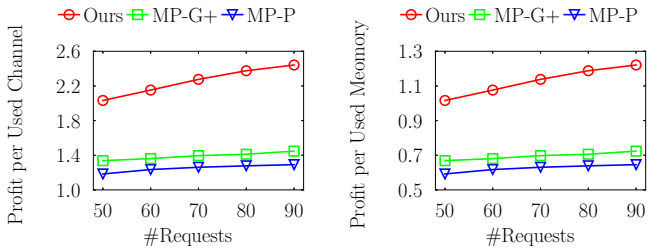
(a) # Reqs. vs Exp. Total Profit (b) # Reqs. vs # Dropped Reqs.

(c) τ vs Exp. Total Profit (d) Avg. $P_s(v)$ vs Exp. Total Profit

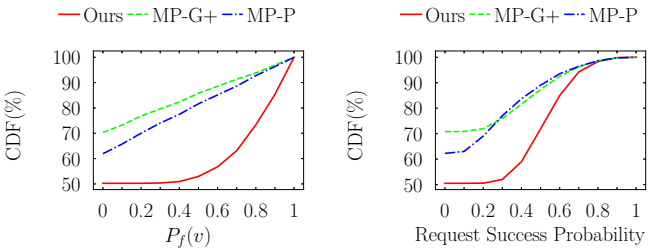
(e) # Nodes vs # Exp. Total Profit (f) # Nodes vs # Dropped Reqs.

Fig. 5. Expected total profit and # dropped requests with different parameters.

[1]To bound the worst-case complexity, we can further generate a polynomial-sized set of ϵ-approximated extreme paths by [12] to get an approximated bi-objective shortest path for the PDA at a slightly higher bounded error.



(a) # Reqs. vs Profit Used per Channel (b) # Reqs. vs Profit Used per Memory



(c) CDF of $P_f(v)$ (d) CDF of Req. Success Prob.

Fig. 6. Effect of different parameters on different metrics.

B. Numerical Results

Overall, Ours outperforms all the other algorithms, as shown in Figs. 5 and 6. Note that UB denotes the fractional solution of the primal LP of the ZERO derived by the PDA [9], which might be infeasible but can serve as a lower bound.

1) Effect of Number of requests: Fig. 5(a) shows the expected total profit with varying numbers of requests. The more requests cause the more resource competition. Ours performs better due to its ability to leverage the fractional solution provided by the LP, enabling it to select three-pointed stars with a balanced success probability and higher profitability for requests while circumventing hot points. Thus, Ours satisfies proper requests first, yielding the most profit to address the first challenge. In contrast, the others greedily allocate resources to requests without considering load balancing and hot point congesting. Thus, they tend to drop more requests, as shown in Fig. 5(b), while Ours has the most expected total profit.

2) Effect of Probability and Resource: Figs. 5(c)−5(d) show the impact of entangling and swapping success probabilities on the expected total profit. As the two probabilities grow (i.e., lower τ and higher $P_s(v)$), the expected total profit increases because all entangled paths have a higher success probability of construction. Fig. 5(e)−5(f) depict the impact of the number of nodes on the expected total profit and request satisfaction. More nodes implies more candidate paths, leading to higher expected total profit and request satisfaction for all algorithms. Ours identifies the most suitable three-pointed stars (i.e., a lower load and a higher probability) by the separation oracle, addressing the second and third challenges. Ours can outperform the MP-P and MP-G+ by up to 103% and 139%, respectively.

3) Resource Efficiency: Figs. 6(a)−6(b) display the average resource efficiency (i.e., the ratio of the expected total profit to the total number of utilized channels or memory units). More requests provide more opportunities to enhance resource efficiency. Ours performs better since it carefully balances resource consumption and expected profit. Although MP-P generally achieves a higher expected total profit than MP-G+, it tends to waste more resources. As a result, its profit per used channel or memory is lower than that of MP-G+.

4) Effect of Fusion Node Selection and Request Success Probability: Fig. 6(c) shows the cumulative distribution function (CDF) of fusion probability $P_f(f)$ of each three-pointed star with fusion node f selected for each request. Ours tends to select those fusion nodes with a higher probability and thus increases the success probability of three-pointed star construction. Fig. 6(d) further shows that Ours selects the three-pointed stars with higher success probabilities than the others. Overall, Ours can strike a better balance between resource allocation and request satisfaction, reducing wastage and enhances network efficiency in QNs.

V. Conclusion

This paper presents a novel optimization problem named ZERO to maximize the expected total profit under the memory and channel limits for 3-GHZ requests, which is NP-hardness. To deal with the challenges, we cleverly design the separation oracle for the dual problem incorporated by the primal-dual algorithm. Then, we develop a $(O(1), O(\log |V|))$-approximation algorithm via randomized rounding for the ZERO. A heuristic is further provided to rectify the deviations of constraints. Finally, simulation results show that our algorithm can outperform the existing approaches by up to 103%.

References

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

[2] M. Hillery, V. Bužek, and A. Berthiaume, "Quantum secret sharing," *Physical Review A*, vol. 59, no. 3, p. 1829, 1999.

[3] R. Van Meter, *Quantum Networking*. John Wiley & Sons, Ltd, 2014.

[4] Z. Yiming *et al.*, "Entanglement routing over quantum networks using greenberger-horne-zeilinger measurements," in *IEEE ICDCS*, 2023.

[5] N. Sangouard *et al.*, "Quantum repeaters based on atomic ensembles and linear optics," *Rev. Mod. Phys.*, vol. 83, p. 33, 2011.

[6] S.-M. Huang *et al.*, "Authorizable tripartite entanglement routing via 3-GHZ state in quantum networks (full version)," Aug. 2024. [Online]. Available: https://github.com/Ching-Ting-Wei/24-GC-3GHZ-QN

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

[8] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Third Edition*. The MIT Press, 2009.

[9] N. Garg and J. Könemann, "Faster and simpler algorithms for multicommodity flow and other fractional packing problems," *SIAM J. Comput.*, vol. 37, pp. 630–652, 2007.

[10] M. I. Henig, "The shortest path problem with two objective functions," *Eur. J. Oper. Res.*, vol. 25, pp. 281–291, 1986.

[11] A. Sedeño-Noda and A. Raith, "A dijkstra-like method computing all extreme supported non-dominated solutions of the biobjective shortest path problem," *Comput Oper Res.*, vol. 57, pp. 83–94, 2015.

[12] S. Vassilvitskii and M. Yannakakis, "Efficiently computing succinct trade-off curves," *Theor. Comput. Sci.*, vol. 348, pp. 334–356, 2005.

[13] E. Sutcliffe and A. Beghelli, "Multiuser entanglement distribution in quantum networks using multipath routing," *IEEE Trans. Quantum Eng.*, vol. 4, pp. 1–15, 2023.

[14] Y. Lee *et al.*, "Quantum network utility: A framework for benchmarking quantum networks," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 121, p. e2314103121, 2024.

[15] V. Krishna, *Auction theory*. Academic press, 2009.

[16] C. Elliott, "Building the quantum network," *New J. Phys.*, vol. 4, p. 46, 2002.
[17] R. V. Meter and J. Touch, "Designing quantum repeater networks," *IEEE Commun. Mag.*, vol. 51, pp. 64–71, 2013.
[18] S. Muralidharan *et al.*, "Optimal architectures for long distance quantum communication," *Sci. Rep.*, vol. 6, pp. 1–10, 2016.
[19] S. Pirandola *et al.*, "Fundamental limits of repeaterless quantum communications," *Nat. Commun.*, vol. 8, p. 15043, 2017.
[20] M. Caleffi, "Optimal routing for quantum networks," *IEEE Access*, vol. 5, pp. 22 299–22 312, 2017.
[21] Y. Zhao and C. Qiao, "Distributed transport protocols for quantum data networks," *IEEE/ACM Trans. Netw.*, vol. 31, pp. 2777–2792, 2023.
[22] M. Pant *et al.*, "Routing entanglement in the quantum internet," *NPJ Quantum Inf.*, vol. 5, pp. 1–9, 2019.
[23] Y. Zhao *et al.*, "E2E fidelity aware routing and purification for throughput maximization in quantum networks," in *IEEE INFOCOM*, 2022.
[24] L. Chen *et al.*, "A heuristic remote entanglement distribution algorithm on memory-limited quantum paths," *IEEE Trans. Commun.*, vol. 70, no. 11, pp. 7491–7504, 2022.
[25] A. Farahbakhsh and C. Feng, "Opportunistic routing in quantum networks," in *IEEE INFOCOM*, 2022.
[26] Y. Zeng *et al.*, "Entanglement routing design over quantum networks," *IEEE/ACM Trans. Netw.*, 2023.
[27] J. Li *et al.*, "Fidelity-guaranteed entanglement routing in quantum networks," *IEEE Trans. Commun.*, vol. 70, pp. 6748–6763, 2022.
[28] M. Ghaderibaneh *et al.*, "Efficient quantum network communication using optimized entanglement swapping trees," *IEEE Trans. Quantum Eng.*, vol. 3, pp. 1–20, 2022.
[29] K. Chakraborty *et al.*, "Entanglement distribution in a quantum network: A multicommodity flow-based approach," *IEEE Trans. Quantum Eng.*, vol. 1, pp. 1–21, 2020.
[30] S. Pouryousef *et al.*, "A quantum overlay network for efficient entanglement distribution," *arXiv:2212.01694*, 2022.
[31] G. Zhao *et al.*, "Segmented entanglement establishment with all-optical switching in quantum networks," *IEEE/ACM Trans. Netw.*, vol. 32, pp. 268–282, 2023.
[32] L. Yang *et al.*, "Asynchronous entanglement provisioning and routing for distributed quantum computing," in *IEEE INFOCOM*, 2023.
[33] A. Patil, M. Pant, D. Englund, D. Towsley, and S. Guha, "Entanglement generation in a quantum network at distance-independent rate," *NPJ Quantum Inf.*, vol. 8, p. 51, 2022.
[34] D. P. Williamson and D. B. Shmoys, *The Design of Approximation Algorithms*. Cambridge University Press, 2011.

APPENDIX A
RELATED WORK

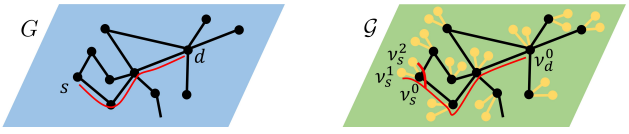
Elliott *et al.* introduced QNs for secure communications [16]. Meter *et al.* proposed a large QN architecture featuring layered recursive repeaters, designed to support quantum sessions and ensure robustness and interoperable communication among non-trusted repeaters [17]. Muralidharan *et al.* presented new quantum nodes that can execute the QEC processes and classified the theoretically feasible technologies of quantum nodes into three generations [18]. Pirandola *et al.* explored the limitations of repeater-less quantum communications and provided general benchmarks for repeaters [19]. Caleffi *et al.* designed a routing protocol aimed at maximizing the end-to-end entanglement rate between any two nodes [20]. Zhao *et al.* proposed two transport layer protocols for quantum data networks, aiming to achieve high throughput and fairness [21].

Pant *et al.* presented a greedy algorithm for determining paths for each request [22]. Shi *et al.* devised the Q-CAST routing method, based on the Dijkstra algorithm, to find routing paths and recovery paths to mitigate the impact of entanglement failures [1]. Zhao *et al.* introduced the LP-based algorithm REPS to maximize the throughput in software-defined networking (SDN)-based QNs [7]. Zhao *et al.* considered the fidelity of entangled links and utilized quantum purification to enhance link fidelity [23]. Chen *et al.* proposed two heuristics for the entangling and swapping phases separately [24]. Farahbakhsh *et al.* developed an add-on scheme to efficiently store and forward data qubits [25]. Zeng *et al.* simultaneously maximized the number of quantum-user pairs and their expected throughput [26]. Li *et al.* leveraged purification to meet fidelity requirements for multiple source-destination (SD) pairs as efficiently as possible [27]. Ghaderibaneh *et al.* addressed time decoherence in determining the swapping order for each SD pair to maximize entanglement rates [28]. Chakraborty *et al.* formulated an LP to compute the maximum total entanglement distribution rate [29]. Pouryousef *et al.* explored stockpiling entangled pairs in advance during low traffic demand periods, utilizing them when needed [30]. Zhao *et al.* investigated room-size network scenarios, where longer entangled pairs could be established using all-optical switching without intermediate node swapping [31]. Yang *et al.* proposed an asynchronous model to freely generate entangled pairs or conduct swapping processes without limit of time slots [32]. However, all these works primarily focus on Bell pairs connecting from source to destination.

Patil *et al.* developed a protocol for entanglement generation in the quantum internet, enabling a repeater node to employ n-qubit GHZ projective measurements capable of fusing n successfully entangled links [33]. Zeng *et al.* leveraged the properties of n-fusion to enhance the success probability of constructing long entangled pairs for QNs [4]. However, these works only consider two users (i.e., Alice and Bob) and do not account for the presence of a third party (i.e., Jacky).

Sutcliffe *et al.* proposed three protocols for constructing n-GHZ states among nodes. In MP-G+, a fusion node is fixed by estimating the total success rate, while MP-C finds a Steiner tree to minimize the total number of hops. Additionally, MP-P incorporates multiple Steiner trees to generate a better solution and achieve higher performance among the three protocols [13]. However, their approaches are limited to grid graphs and does not consider for the constraints imposed by quantum channels.

APPENDIX B
PROOF OF THEOREM 1



(a) Instance of IMCMFP (b) Instance of ZERO

Fig. 7. Example of reduction from an instance of the IMCMFP

We prove the theorem by reducing the decision version of the integer minimum-capacity multicommodity flow problem (IMCMFP) [34] to the ZERO. In the decision version of the IMCMFP, the input data includes an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a set of SD pairs $\mathcal{I}$, and the link load limit $\mathcal{L}$. It aims to decide whether it is possible to find a simple path for each

SD pair $i \in \mathcal{I}$, while ensuring the max link load is no greater than $\mathcal{L}$.

We show how to construct a corresponding instance $S = \{G = (V, E), R, \mathcal{O}(r), m(v), c(e), P_s(v), P_f(v), P_e(e)\}$ of the ZERO in polynomial time for any given instance $\mathcal{S} = \{\mathcal{G} = (\mathcal{V}, \mathcal{E}), \mathcal{I}, \mathcal{L}\}$ of the IMCMFP. First, for each node u in $\mathcal{V}$, we create three quantum nodes v_u^0, v_u^1, and v_u^2, and then add them to V, and set their memory limits to a sufficiently large value, i.e., $m(v_u^0) = m(v_u^1) = m(v_u^2) = 2|\mathcal{I}|$. Then, we create edge e between v_u^0 and the other two nodes, (i.e., link $\overline{v_u^0 v_u^1}$ and $\overline{v_u^0 v_u^2}$) for every v_u^0 in V and set their channel capacity to a sufficient large value, i.e., $c(e) = 2|\mathcal{I}|$. For example, in Fig. 7, for a node s in Fig. 7(a), we create one black node and two yellow nodes, denoted by v_s^0, v_s^1, v_s^2, and establish yellow links $\overline{v_s^0 v_s^1}$ and $\overline{v_s^0 v_s^2}$ with sufficient channel capacity in Fig.7(b). Next, we create an edge e to connect the corresponding quantum nodes v_u^0 and v_w^0, add it to E, and set its channel limit $c(e)$ to $\mathcal{L}$ for each edge in $\mathcal{E}$ that connects nodes u and w in $\mathcal{G}$. For instance, in Fig. 7, each black link $\overline{uw}$ in Fig. 7(a) corresponds to a black link $\overline{v_u^0 v_w^0}$ in Fig. 7(b) with the same capacity size of $\mathcal{L}$. We set the entangle probability $P_e(e)$ of every edge $e \in E$, the swapping probability $P_s(v)$, and fusion probability $P_f(v)$ of every node $v \in V$ are set to one. Finally, for each SD pair $i \in \mathcal{I}$ which has source s and destination d in $\mathcal{V}$, we create a request with three end nodes v_s^1, v_s^2, and v_d^0 in V, set its profit $\mathcal{O}(r) = 1$, and add it to R. Such a request will have fusion node at v_s^0 if it is served, as shown in Fig. 7. Clearly, the above reduction can be done in polynomial time.

We proceed to demonstrate that the maximum load in the instance $\mathcal{S}$ is not greater than $\mathcal{L}$ if and only if the maximum expected profit in S is equal to the number of requests (i.e., $|R|$). Suppose there is a solution with the maximum load no greater than $\mathcal{L}$ for the instance $\mathcal{S}$, illustrated by the red line in Fig. 7(a). In this scenario, each request in S can have a corresponding three-pointed star (red line in Fig. 7(b)), which includes a corresponding subpath (e.g., the path from v_s^0 to v_d^0), to transmit one qubit with a probability of 1. Consequently, a total profit of $|R|$ can be acquired through the corresponding paths as in $\mathcal{S}$ of the IMCMFP. Conversely, assume that the maximum load in the instance $\mathcal{S}$ is greater than $\mathcal{L}$. Since the number of channels on each edge $\overline{v_u^0 v_w^0}$ is set to $\mathcal{L}$, it is impossible to satisfy the demands of all requests in S without deviating the channel limit. Therefore, the total expected profit must be less than $|R|$. It completes the reduction, and the theorem follows.

Appendix C
Proof of Lemma 1

It suffices to show that given any $x_1, y_1, x_2, y_2 > 0$ such that $U(x_1, y_1) \geq U(x_2, y_2)$, the inequality $U'_x(x_2, y_2)(x_1 - x_2) + U'_y(x_2, y_2)(y_1 - y_2) \geq 0$ always holds, where $U'_x(x, y) = \frac{\partial U(x,y)}{\partial x} = e^y$ and $U'_y(x, y) = \frac{\partial U(x,y)}{\partial y} = x \cdot e^y$. Then,

$$\begin{aligned} & U'_x(x_2, y_2)(x_1 - x_2) + U'_y(x_2, y_2)(y_1 - y_2) \\ &= e^{y_2}(x_1 - x_2) + x_2 e^{y_2}(y_1 - y_2) \\ &= x_2 e^{y_2}\left(\frac{x_1}{x_2} - 1 + y_1 - y_2\right) \geq x_2 e^{y_2}\left(\frac{x_1}{x_2} - 1 + \ln \frac{x_2}{x_1}\right) \geq 0. \end{aligned}$$

The second to last inequality is correct because $U(x_1, y_1) \geq U(x_2, y_2)$ implies $x_1 e^{y_1} \geq x_2 e^{y_2}$, leading to $y_1 - y_2 \geq \ln \frac{x_2}{x_1}$. Plus, the last one holds since $x + \ln \frac{1}{x} - 1 \geq 0$ as $x > 0$.

Appendix D
Proof of Theorem 2

It is clear that our algorithm chooses at most one three-pointed star for each request by randomized rounding, meeting constraint (1d). To prove Theorem 2, we have to guarantee the probability of deviating the memory or channel limit (i.e., constraints (1b) and (1c)) within a bounded factor is sufficiently large. Thus, we employ the Chernoff bound (i.e., Theorem 3) to bound the deviation. Subsequently, we analyze the time complexity and approximation ratio of our algorithm to complete the proof.

Theorem 3 (Chernoff bound)**.** There is a set of n independent random variables $x_1, ..., x_n$, where $x_i \in [0, 1]$ for each $i \in [1, n]$. Let $\mu = \mathbb{E}[\sum_{i=1}^{n} x_i]$. Then,

$$\Pr\left[\sum_{i=1}^{n} x_i \geq (1 + \epsilon)\mu\right] \leq e^{\frac{-\epsilon^2 \mu}{2+\epsilon}}. \tag{8}$$

1) Channel limit constraint: We first bound the extent to which the channel limits are deviated. For each edge $e \in E$, we define a random variable z_e^r to denote the number of paths visiting e in the three-pointed star set $T \in T(r)$ selected for request r after randomized rounding. Therefore, it can be written as:

$$z_e^r = \begin{cases} \sum_{p \in T: e \in p} 1 & \text{with } \Pr = \hat{x}_T^r, \ \forall T \in T(r); \\ 0 & \text{otherwise.} \end{cases}$$

Clearly, the random variables z_e^r are independent for any specific edge $e \in E$. Note that their sum $Z_e = \sum_{r \in R} z_e^r$ represents the exact number of paths planned to be constructed through edge e after randomized rounding. Then, we prove the claim that the probability of deviating any edge's channel limit by more than a factor of $(1 + 6 \ln |V|)$ is at most $\frac{1}{|V|^2}$, which is negligible.

We first derive the upper bound of the expectation of the constructed paths over an edge e for randomized rounding.

$$\begin{aligned} \mathbb{E}[Z_e] &= \mathbb{E}\left[\sum_{r \in R} z_e^r\right] = \sum_{r \in R} \mathbb{E}[z_e^r] \\ &= \sum_{r \in R} \sum_{T \in T(r)} \sum_{p \in T: e \in p} \hat{x}_T^r \leq c(e). \end{aligned} \tag{9}$$

Note that the last inequality directly follows the channel limit constraint (1c). Subsequently, we find the upper bound of the probability of deviating $c(e)$ of any edge e after rounding by the Chernoff bound as follows.

$$\begin{aligned} & \Pr(\exists e \in E : Z_e \geq (1 + 6 \ln |V|) \cdot c(e)) \\ \leq & \sum_{e \in E} \Pr(Z_e \geq (1 + 6 \ln |V|) \cdot c(e)) \\ = & \sum_{e \in E} \Pr\left(\sum_{r \in R} \frac{z_e^r}{c(e)} \geq 1 + 6 \ln |V|\right) \end{aligned}$$

$$\leq |V|^2 \cdot e^{\frac{-(6\ln|V|)^2}{2+6\ln|V|}} \leq |V|^2 \cdot e^{-4\ln|V|} = |V|^{-2}.$$

Note that the inequality $\frac{z_e^r}{c(e)} \leq 1$ holds to ensure every random variable is within the range of $[0,1]$, meeting the Chernoff bound's requirement since it is reasonable to assume that the channel usage of link e from a single request r is much lower than the total number of channels. In addition, the third inequality holds when $|V| \geq 3$. Thus, the claim holds.

2) Memory limit constraint: We then consider the memory limit constraint and bound the probability that the amount of quantum memory planned to be used in node v exceeds $m(v)$. Similarly, for each node v, a random variable z_v^r is defined to denote the amount of quantum memory of v used by the paths in the three-pointed star $T \in T(r)$ selected for request r after randomized rounding. In this way, it can be written as:

$$z_v^r = \begin{cases} \sum_{u\in V} \sum_{p\in T:(u,v)\in p} 1 & \text{with } \Pr = \hat{x}_T^r, \ \forall T \in T(r); \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, the random variables z_v^r are independent for each node $v \in V$. Note that their sum $Z_v = \sum_{r\in R} z_v^r$ is exactly the amount of memory planned to transmit data qubits after randomized rounding. Next, we prove the claim that the probability of deviating any node's memory limit by more than a factor of $(1 + 5\ln|V|)$ is at most $\frac{1}{|V|^2}$.

We first acquire the upper bound of the expectation of the occupied memory at node v after rounding is:

$$\mathbb{E}[Z_v] = \mathbb{E}\left[\sum_{r\in R} z_v^r\right] = \sum_{r\in R} \mathbb{E}[z_v^r] \leq m(v).$$

Then, similarly, the probability of deviating the memory limit of any node v by more than a factor of $(1 + 5\ln|V|)$ after rounding by the Chernoff bound can be expressed as:

$$\begin{aligned} &\Pr(\exists v \in V : Z_v \geq (1+5\ln|V|)\cdot m(v)) \\ \leq\ &\Pr(Z_v \geq (1+5\ln|V|)\cdot m(v)) \\ =\ &\sum_{v\in V} \Pr\left(\sum_{r\in R} \frac{z_v^r}{m(v)} \geq 1 + 5\ln|V|\right) \\ \leq\ &|V| \cdot e^{\frac{-(5\ln|V|)^2}{2+5\ln|V|}} \leq |V| \cdot e^{-3\ln|V|} = |V|^{-2}. \end{aligned}$$

Similarly, $\frac{z_v^r}{m(v)} \leq 1$ since in most cases the demand of each request r is much lower than the amount of memory of each node v. Also, the third inequality holds as $|V| \geq 3$. Therefore, the claim also holds.

3) Time complexity: We analyze the time complexity of the proposed algorithm. According to [9], the PDA executes $O(\omega^{-2} m_1 \log m_1)$ times of separation oracle and thus outputs $O(\omega^{-2} m_1 \log m_1)$ three-pointed stars for all request. Note that ω is the user-defined error bound of primal LP solution and m_1 is the number of constraints (except for the non-negativity constraints of variables) in the primal LP, i.e., $m_1 = (|V| + |E| + |R|)$. In addition, each time of separation oracle takes $m_2 = O(|R| \cdot |V|^{1.5}(|E| + |V|\log|V|))$ in expectation based on [10], [11]. Then, in our algorithm, the PDA takes $O(\omega^{-2} m_1 m_2 \log m_1)$, and the randomized rounding spends $O(\omega^{-2} m_1 \log m_1)$. The overall time complexity is $O(\omega^{-2} m_1 m_2 \log m_1)$. Note that the worst-case time complexity can be further bounded if ϵ-approximated extreme paths [12] are used.

By now, it suffices to focus on the value of the objective function of the primal LP.

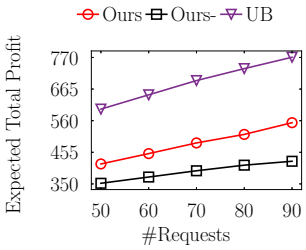
4) Approximation ratio on the value of objective: Let OPT and OPT_{PL} be the optimum values of the ZERO and our primal LP, respectively. It is evident that $OPT \leq OPT_{PL}$ since our primal LP is a relaxation of our ILP. In addition, let $\hat{x}_T^R$ denote the (fractional) solution output by the primal-dual algorithm, and let $\hat{X} = \sum_{r\in R} \sum_{T\in T(r)} \Pr(T) \cdot \mathcal{O}(r) \cdot \hat{x}_T^r$. Then, let $x_T'^r$ be the solution after randomized rounding, and let $X' = \sum_{r\in R} \sum_{T\in T(r)} \Pr(T) \cdot \mathcal{O}(r) \cdot x_T'^r$ be the value of rounded solution. The expected value of the solution after randomized rounding is equal to the solution of the primal LP derived by the PDA. In other word,

$$\begin{aligned} \mathbb{E}[X'] &= \mathbb{E}\left[\sum_{r\in R}\sum_{T\in T(r)} \Pr(T)\cdot\mathcal{O}(r)\cdot\hat{x}_T^r\right] \\ &= \sum_{r\in R}\sum_{T\in T(r)} \Pr(T)\cdot\mathcal{O}(r)\cdot\mathbb{E}\left[x_T'^r\right] \\ &= \sum_{r\in R}\sum_{T\in T(r)} \Pr(T)\cdot\mathcal{O}(r)\cdot\hat{x}_T^r \\ &= APP_{PL} \geq \frac{OPT_{PL}}{1+\omega} \geq \frac{OPT}{1+\omega} = O(1)\cdot OPT, \end{aligned}$$

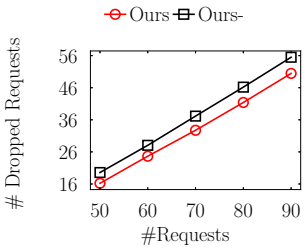
where APP_{PL} is the solution value of primal LP acquired by the PDA and $\omega > 0$ is a user-defined constant. Note that the first inequalities hold according to [9].

APPENDIX E
ABLATION STUDY

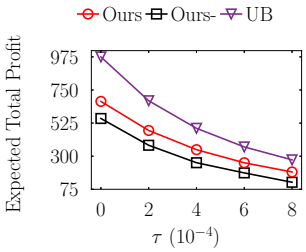
We show that considering the dual cost and probability of a three-pointed star concurrently is crucial for performing outstandingly in the ZERO. We compare Ours with its variant version Ours- in Figs. 8 and 9. Ours- ignores the probability and only considers the dual cost. It solely calculates the numerator part of Eq. (4) and Eq. (5) in Section III for the desired three pointed star with the Dijkstra algorithm. As shown in Figs. 8 and 9, Ours always has higher expected total profit, better resource efficiency, and lower dropped requests than Ours-. In conclusion, Ours outperforms Ours- by up to 29% and thus prove that the separation oracle that we cleverly come up with is necessary for the ZERO.



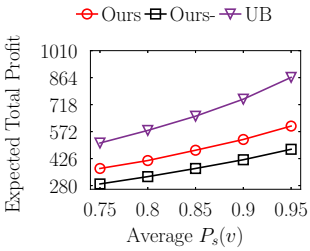
(a) # Reqs. vs Exp. Total Profit



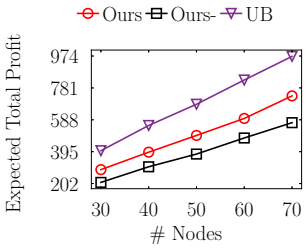
(b) # Reqs. vs # Dropped Reqs.



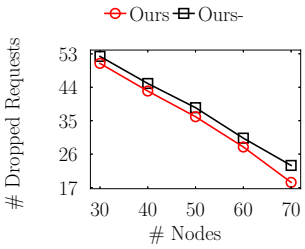
(c) τ vs Exp. Total Profit



(d) Avg. $P_s(v)$ vs Exp. Total Profit

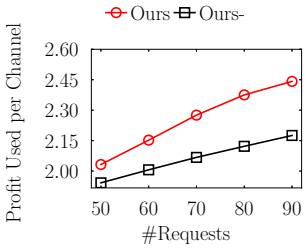


(e) # Nodes vs # Exp. Total Profit



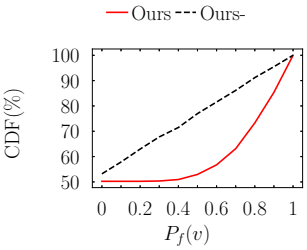
(f) # Nodes vs # Dropped Reqs.

Fig. 8. Expected total profit and # dropped requests with different parameters.

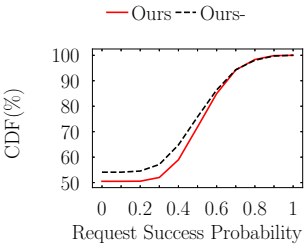


(a) # Reqs. vs Profit Used per Channel

(b) # Reqs. vs Profit Used per Memory



(c) CDF of $P_f(v)$



(d) CDF of Req. Success Prob.

Fig. 9. Effect of different parameters on different metrics.