SFTRAP: Satisfying Fidelity Threshold Routing and Adaptive Purification for Throughput Maximum in Quantum Network

Zhi Wang, Tao Gong, Yingpu Nian, Bo Yi, *Member, IEEE*, Xingwei Wang, *Member, IEEE*, Xinhao Zhou, Jianhui Lv, *Senior Member, IEEE*, Geyong Min, *Member, IEEE*, and Keqin Li, *Fellow, IEEE*

***Abstract*—The core function of quantum networks is to establish high-fidelity quantum entanglement for long-distance communication. However, the main challenge is to efficiently allocate resources under limited conditions, maximize throughput, satisfy end-to-end (E2E) fidelity requirements, and prevent quantum decoherence caused by inefficient routing algorithms. Current research focuses on optimizing either throughput or fidelity, with a lack of approaches that optimize both simultaneously; furthermore, existing algorithms suffer from high computational complexity. To tackle these challenges, this study proposes a Satisfying Fidelity Threshold Routing and Adaptive Purification Strategy (SFTRAP). SFTRAP maximizes throughput for each request by selecting multiple paths and dynamically choosing links for entanglement purification based on the current state of link resources, thus minimizing throughput loss while satisfying fidelity threshold. The strategy also adaptively adjusts the number of purification rounds according to the fidelity threshold, thereby optimizing the time required for deep purification and enhancing algorithmic efficiency. For multi-request scenarios, SFTRAP employs a priority sorting mechanism that takes into account both path cost and path freedom, which refines request scheduling and path selection to create more efficient request combinations, thus further boosting the overall network throughput. Simulation results indicate that SFTRAP surpasses state-of-the-art methods in terms of both throughput and algorithmic efficiency, highlighting its potential for optimizing resources in quantum networks.**

***Index Terms*—Quantum networks, fidelity-guaranteed, entanglement purification, entanglement routing, resource allocation.**

I. Introduction

IN RECENT years, the rapid development of classical information theory has profoundly transformed human life but also revealed new challenges. Moore's law suggests inherent limits to the future growth of computational power [1]. With the advent of quantum computers, however, new paradigms in computation are emerging. By exploiting quantum properties such as superposition, quantum computers offer inherent parallelism and can solve mathematical problems that are difficult or inefficient for classical systems, potentially achieving exponential gains in computational capability. Notably, quantum computing shows strong potential in accelerating computationally intensive tasks, including large-scale training of artificial intelligence models [2], [3]. Despite these advantages, quantum computers are constrained by the limited memory capacity of qubits. A promising approach to mitigate this limitation is the deployment of quantum networks, which interconnect multiple quantum computers into a distributed processing system [4]. Beyond addressing memory constraints, quantum networks are crucial for enabling advanced applications such as quantum sensor networks [6], quantum secure communication [5], [7], and military information countermeasures [8], all of which depend on the reliability and security of quantum networks.

The central challenge in realizing these applications lies in establishing high-fidelity entanglement between non-adjacent nodes and ensuring robust quantum information transmission. A quantum network comprises nodes—serving as quantum computers or repeaters—interconnected by optical fibers or free space links. Each node is equipped with limited quantum memory and supports qubit generation, storage, manipulation, and swapping. Given the physical constraints of quantum memories, available entanglement resources are inherently scarce. In the basic case, adjacent nodes establish entanglement through photon-based mechanisms, typically by encoding quantum states in photon polarization or phase [9]. The entangled photons are then distributed via optical channels and stored in quantum memories to sustain entanglement over time.

Received 28 December 2024; revised 2 June 2025 and 17 October 2025; accepted 19 January 2026. Date of current version 5 February 2026. This work is supported by the National Natural Science Foundation of China under Grant No.62472078, U22A2004; The Fundamental Research Funds for the Central Universities of China under Grant No. N25LPY013; the Open Research Fund from Guangdong Laboratory of Artificial Intelligence and Digital Economy under Grant No. GML-KF-24-31; the State Grid Corporation of China Science and Technology Project Funding under Grant No.2024YF-95. The associate editor coordinating the review of this article and approving it for publication was Z. Rezki. *(Corresponding author: Bo Yi.)*

Zhi Wang, Tao Gong, Yingpu Nian, and Xingwei Wang are with the College of Computer Science and Engineering, Northeastern University, Shenyang 110169, China (e-mail: 2201740@stu.neu.edu.cn; 2401766@stu.neu.edu.cn; 2410795@stu.neu.edu.cn; 2401790@stu.neu.edu.cn).

Bo Yi is with the College of Computer Science and Engineering, Northeastern University, Shenyang 110169, China, and also with the Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Shenzhen 518060, China (e-mail: yibo@cse.neu.edu.cn).

Xinhao Zhou is with the Software College, Northeastern University, Shenyang 110169, China (e-mail: xinhaoz@alumni.cmu.edu).

Jianhui Lv is with the Multi-modal Data Fusion and Precision Medicine Laboratory, The First Affiliated Hospital of Jinzhou Medical University, Jinzhou 121012, China (e-mail: lvjh@jzmu.edu.cn).

Geyong Min is with the Department of Computer Science, University of Exeter, EX4 4QF Exeter, U.K. (e-mail: g.min@exeter.ac.uk).

Keqin Li is with the Department of Computer Science, State University of New York, New York, NY 12561 USA (e-mail: lik@newpaltz.edu).

Digital Object Identifier 10.1109/TCOMM.2026.3658364

Quantum information is transmitted to the destination through quantum teleportation, utilizing entanglement resources. However, the success rate of entanglement distribution between two quantum nodes decreases exponentially with increasing physical distance. Moreover, quantum information adheres strictly to the no-cloning theorem [10], rendering classic signal amplification and regeneration methods unsuitable for quantum communication. As a result, quantum repeaters are essential for establishing remote end-to-end (E2E) entanglement. These repeaters, situated between the source and destination nodes, facilitate entanglement swapping [11], converting link-level entanglement into E2E entanglement. The E2E entanglement connection includes three steps: First, adjacent nodes establish link-layer entanglements, which are defined as the quantum network's resources. Second, the repeaters exploit Bell State Measurements (BSM) [12] to convert link-level entanglements into E2E entanglements through a process known as entanglement swapping. Finally, the BSM results are sent to the destination node. The destination node applies Pauli X or Z gates [13] based on the BSM results to recover the state of the entanglement to the desired Bell state.

Although repeaters are used to establish remote entanglements, the success rate of quantum information transmission is not guaranteed. This is because the quantum entanglement is extremely fragile and susceptible to external environmental interference during entanglement distribution and swapping. Quantum decoherence would further exacerbate quantum information loss. Therefore, fidelity is used to quantify the quality of entanglements. Fidelity is a value between 0 and 1 that measures the similarity between a transmitted quantum state and target state. Entanglement purification is utilized to improve fidelity. This technique consumes low-fidelity entanglements between adjacent nodes to obtain high-fidelity entanglements. To construct a large-scale quantum network and enable efficient and accurate quantum information transmission, key challenges include: *1) In a resource-constrained quantum network, purifying a link with limited resources or one that is already multiplexed can significantly decrease available resources, potentially turning it into a bottleneck or even causing congestion, ultimately reducing the network's overall throughput. 2) To satisfy the fidelity requirements, purification of the entanglements is necessary. However, the purification process significantly increases the computational complexity of the algorithm. In addition, the entanglement fidelity decreases with time increasing due to quantum decoherence. As a result, fidelity loss is exacerbated when less efficient purification algorithms are used, further degrading the network's performance.*

This paper proposes the **S**atisfying **F**idelity **T**hreshold for **R**outing and **A**daptive **P**urification (SFTRAP). To maximize throughput, SFTRAP employs the K-shortest path algorithm to identify multiple paths from source to destination, and uses these paths to establish multiple E2E entanglements. Next, to meet the fidelity requirements of requests and minimize the impact of purification on throughput, SFTRAP jointly optimizes the routing and purification strategies, satisfying fidelity requirements and maximizing throughput. Finally, to reduce the computational complexity involved in finding the optimal purified link, SFTRAP incorporates an adaptive purification strategy that dynamically adjusts the number of purification rounds according to fidelity demands. Through these optimizations, our ultimate goal is to transform quantum networks from a theoretical concept into a practical, seamlessly integrated system that works alongside existing infrastructures. By effectively balancing fidelity and throughput, SFTRAP provides a solid foundation for the next generation of quantum applications.

The main contributions are as follows:

- We propose a joint optimization approach for routing and purification, which employs the k-shortest path algorithm to identify multiple paths and establishes E2E entanglements based on these paths, maximizing the throughput of the quantum network and satisfying fidelity requirements.
- We introduce an adaptive purification strategy to reduce the computational complexity of the purification process. This strategy dynamically adjusts the number of purification rounds based on the specific fidelity requirements, effectively reducing both the time and computational cost associated with finding the optimal purified links.
- We extend SFTRAP to multi-request scenarios by proposing a priority sorting mechanism that considers both path cost and path freedom. This mechanism optimizes request scheduling and path selection, improving request combinations in terms of throughput and resource efficiency, thereby further enhancing overall network performance.

The structure of this paper is as follows: Section II reviews related work, Section III presents the problem model, Section IV focuses on the design of SFTRAP, Section V provides an analysis of the experimental results, and Section VI concludes the paper.

II. Related Work

The Internet has revolutionized our world, and now the quantum Internet ushers in a new era, aiming to enable global quantum communication while simultaneously enhancing computational capabilities and the security of information transmission [14]. Furthermore, the latest achievement has opened a new phase in quantum Internet research, paving the way for large-scale quantum networks and bringing entanglement-based communication closer to practical implementation [15].

Quantum networks play a critical role in the field of quantum information [14], [16], [17], [18], [19]. Previous research explored several specific network topologies, such as diamond, ring, sphere, and star topologies [17]. These studies aimed to maximize the utilization of network resources and throughput within specific topologies. Based on these efforts, some studies focused on entanglement routing design in universal network topologies [20], but overlooked the potential failures of entanglement establishing and swapping. To simulate the quantum network more realistically, Shi et al. [21] considered entanglement generation failure and utilized successfully generated entanglement to establish E2E entanglement with the fewest hops. Moreover, Zhao and Qiao [22] considered both entanglement generation and swapping failures, allocating additional

resources to repeaters to repeatedly generate entanglement until successful. Furthermore, Hahn et al. [23] proposed an efficient adaptive routing algorithm based on graph theory. In the event of an entanglement swapping failure, the algorithm dynamically selects the shortest alternative path, thereby ensuring the maintenance of E2E entanglement. However, these studies failed to adequately address entanglement fidelity, which directly impacts the probability of successful quantum information transmission. Pant et al. [24] highlighted that time-induced quantum decoherence has a significant impact on entanglement fidelity. To address this, their study focused on maintaining high-fidelity entanglement paths by minimizing the time for entanglement generation and swapping, thereby preserving the desired state of entanglement. However, this approach overlooked potential fidelity degradation during the entanglement generation phase.

Wang et al. [25] proposed several quantum purification schemes to enhance entanglement fidelity by allocating additional resources to each link. However, their approach mainly improved local fidelity without ensuring E2E fidelity. Li et al. [26] proposed a method where entanglements are purified first, and paths are subsequently selected based on the purified links to ensure E2E fidelity. However, this approach would waste resources by purifying all links, and the fidelity loss during entanglement swapping undermines the ability to guarantee E2E fidelity. Li et al. [27] proposed a method in which the Dijkstra algorithm is used to determine the optimal path, and the links along this path are then purified to ensure E2E fidelity. However, this approach selects the link with the greatest potential for fidelity improvement for purification, neglecting the resource limitations of the link itself and the possibility of reusing the link for multiple paths. This leads to the link becoming a communication bottleneck, thus limiting the network's maximum throughput. Moreover, the purification strategy in this method is static and lacks adaptive adjustment of purification rounds, resulting in increased resource consumption and higher processing latency.

TABLE I
KEY VARIABLES AND DEFINITIONS

Symbol	Concise Meaning
Network & Request	
$G = (V, E)$	Quantum network topology
$l(u, v)$	Quantum link between nodes u and v
$r^k = (s^k, d^k)$	Routing request from s^k to d^k
p_m^k	The m-th path for request r^k
Physical Layer & Resources	
$\alpha, \Delta t$	Fiber attenuation coefficient, Time slot
$R_{(u,v)}$	Number of available entangled pairs on link $l(u, v)$
R_{max}	Global maximum of entangled pairs across all links
$r_{(u,v)}^{k,m}$	Resources allocated to path p_m^k on link $l(u, v)$
Fidelity & Purification	
$F_{k,m}^{E2E}$	End-to-end fidelity of path p_m^k
$F_{(u,v)}$	Fidelity of link $l(u, v)$
$F_{k,m}^{th}$	Fidelity threshold for path p_m^k
N^{pur}, P_{succ}	Number of purification rounds, Success probability of one round purification
SFTRAP Core Metrics	
$ET_{k,m}^{E2E}$	Expected throughput for path p_m^k
$ET_{k,m}(u, v)$	Throughput contribution on link $l(u, v)$ for path p_m^k
$pairs_{k,m}(u, v)$	Number of purification attempts for path p_m^k on link $l(u, v)$
$D_{k,m}^{pur}$	Purification decision metric for path p_m^k
$R_{k,m}^{max}, E_{k,m}^{max}$	Path-specific max resources and link reuse for normalization
Q	Queue for storing data or path schemes
$Cost_{k,m}(u, v)$	Resource consumption on link $l(u, v)$ for path p_m^k

III. NETWORK & PROBLEM MODEL

A. Network Model

Table I presents a summary of the principal variables employed throughout the paper, along with their concise definitions.

We adopt a quantum network model based on a centralized classical-quantum collaborative architecture, a widely recognized paradigm for future quantum networks [28]. In this model, the quantum layer handles entanglement generation, distribution, and maintenance, while the classical layer provides centralized control to monitor network states and coordinate operations. The controller collects event-driven updates on entangled pair availability, fidelity, link utilization, and node resources, enabling adaptive decision-making at the network layer. This design allows the network to maintain high performance under dynamic conditions such as link degradation or node failures. For simplicity, decoherence within a time window Δt is not modeled, as the controller operates per Δt with updated states. We also assume ideal Bell-state measurements, negligible classical communication latency, and memory lifetimes sufficiently long so that storage decay does not bind within Δt. These assumptions allow us to isolate and evaluate the routing and purification strategies, while more detailed physical-layer effects are left for future work. Within this framework, the proposed SFTRAP algorithm leverages real-time resource information to jointly optimize routing and purification. Using a K-shortest path algorithm, SFTRAP selects candidate paths for each request to meet throughput demands, applies purification to satisfy fidelity constraints, and then executes entanglement swapping to establish E2E connections. Thus, the process of generating E2E entanglement is realized through coordinated entanglement generation, purification, and swapping.

1) Entanglement Generation: Quantum entanglement is generated using spontaneous parametric down-conversion (SPDC) [29], where a laser beam passes through a nonlinear crystal to produce photon pairs entangled in their polarization states. These photon pairs are transmitted through the optical fiber network to adjacent quantum repeaters. We define the entanglement generation rate on link $l(u, v)$ as $r(u, v)$, which is set to be equal across all links. Due to the imperfect efficiency of the SPDC process and optical fiber attenuation, the success probability of generating an entanglement pair on $l(u, v)$ is

$$q(u, v) = \left(1 - \gamma_{init}\right) \cdot 10^{-\alpha d(u,v)/10}, \tag{1}$$

where γ_{init} denotes the initial loss rate from imperfect SPDC efficiency, $d(u, v)$ is the physical length of link $l(u, v)$, and

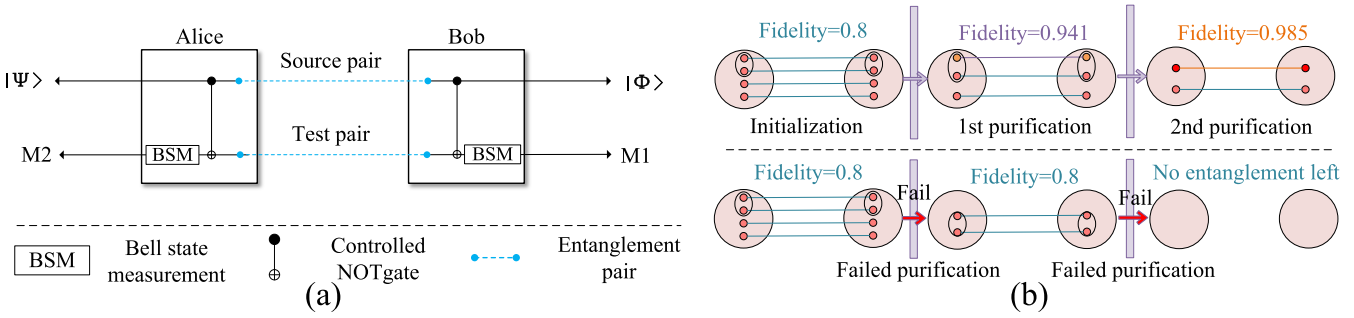


Fig. 1. Using purification to improve fidelity.

α is the fiber attenuation coefficient in dB/km (typically $\alpha \approx 0.2$ dB/km at 1550 nm). Within a time interval Δt, the number of entangled pairs generated on $l(u,v)$ follows a binomial distribution $B(r(u,v)\Delta t, q(u,v))$, with expected value $r(u,v)q(u,v)\Delta t$. Considering the memory capacities of nodes u and v, denoted by m_u and m_v, the available entanglements $R_{(u,v)}$ on link $l(u,v)$ are then given by

$$R_{(u,v)} = min(r(u,v)q(u,v)\Delta t, m_u, m_v) \tag{2}$$

where $R_{(u,v)}$ denotes the available entanglement bandwidth on link $l(u,v)$ per time slot Δt, reflecting the link's effective entanglement capacity. This value is determined by the minimum of three factors: the expected number of successfully generated and transmitted entangled pairs, $r(u,v)q(u,v)\Delta t$, and the quantum memory capacities at the endpoint nodes, m_u and m_v.

2) Entanglement Purification: Entanglement purification is achieved by sacrificing low-fidelity entanglement to obtain high-fidelity entanglements. The purification operation is depicted in Figure 1 (a). Direct measurement of the source entanglement would lead to the collapse of the quantum superposition state into a single state, thereby losing its entanglement properties. Consequently, it becomes imperative to use test entanglement to probe the state of the source entanglement. Alice and Bob each have one qubit for the source entanglement pair and the test entanglement pair. These qubits are sent to a controlled-NOT (CNOT) gate, which performs an XOR operation, preserving the parity check between the source and test qubits as they pass through the gate. We measure the state of the test pair to determine whether the source pair is in the desired state.

The top part of Figure 1 (b) illustrates a successful purification example. If the measurement results of the test pair (denoted as M1 and M2) are consistent, it indicates that the source entanglement is in the Bell state $|\beta_{00}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, where β_{00} denotes the maximally entangled state. In this state, the two qubits are perfectly correlated, meaning both are in $|0\rangle$ or both in $|1\rangle$ with equal probability. This implies that the source entanglement has not suffered from bit-flip or phase-flip errors. As a result, the entangled state is retained, and its fidelity is improved.

Conversely, the lower part of Figure 1 (b) shows a failed purification example. If the measurement results are inconsistent, the source entanglement is discarded. The test pair is always discarded, as it no longer retains entanglement after measurement. Notably, M1 and M2 produce identical results in two cases: (i) when the source entanglement is truly in the ideal Bell state $|\beta_{00}\rangle$; and (ii) when both qubits have undergone a bit-flip, falsely suggesting the state is $|\beta_{00}\rangle$, resulting in a false positive. Assuming the initial fidelities of the two entangled pairs are F_1 (the source pair) and F_{anc} (the ancillary pair), we adopt the entanglement pumping protocol based on the DEJMPS protocol [36]. Thus the probability of a positive measurement outcome [30] is

$$P_{\text{succ}}(F_1, F_{\text{anc}}) = F_1 F_{\text{anc}} + (1 - F_1)(1 - F_{\text{anc}}). \tag{3}$$

Accordingly, the fidelity of the purified source entanglement is

$$F = \frac{F_1 F_{\text{anc}}}{P_{\text{succ}}(F_1, F_{\text{anc}})} > \max(F_1, F_{\text{anc}}). \tag{4}$$

To further improve the quality of raw entanglements, we further increase the fidelity of entanglement by applying multiple rounds of purification, as shown in the top part of Figure 1 (b). After t rounds of purification, the fidelity of the entanglement, denoted by F_t, is updated as

$$F_t = \frac{F_{t-1} F_{\text{anc}}}{P_{\text{succ}}(F_{t-1}, F_{\text{anc}})}, \tag{5}$$

with $F_0 = F_1$.

The success probability of the t-th purification round is exactly $P_t = P_{\text{succ}}(F_{t-1}, F_{\text{anc}})$. For a sequence of n rounds, the overall purification success probability is the multiplicative product:

$$P_n = \prod_{t=1}^{n} P_{\text{succ}}(F_{t-1}, F_{\text{anc}}). \tag{6}$$

3) Entanglement Swapping: Entanglement swapping is a method for achieving E2E entanglements. The entangled photons are stored in repeaters. By performing the BSM operation in repeaters, two pairs of entangled photons are coupled together to form a long-distance entangled pair. Entanglement swapping enables remote communication partners to establish E2E entanglements by connecting multiple single-hop entanglements along a predetermined path. The fidelity of an

E2E entangled pair $F_{k,m}^{E2E}$ follows the standard Werner-state mapping under ideal BSMs [35]:

$$F_{k,m}^{E2E} = \frac{1}{4} + \frac{3}{4} \prod_{(u,v)\in p_m^k} \frac{4F_{(u,v)} - 1}{3}. \tag{7}$$

Here, $F_{(u,v)}$ denotes the fidelity of link $l(u,v)$, and p_m^k represents the m-th path of request r^k. As the number of hops increases, $F_{k,m}^{E2E}$ decreases accordingly, posing a challenge in designing entanglement routing with guaranteed fidelity.

B. Problem Model

The problem of satisfying fidelity-threshold entanglement routing is formulated as follows: At a given time slot, for a quantum network $G = (V, E)$, the bandwidth $R_{(u,v)}$ on the link $l(u,v)$ is determined by equation (2). We define a request set $Req = \{r^k | k \leq K\}$, where K is the total number of requests. Each individual request r^k consists of two components: $r^k = (s^k, d^k)$. Here, s^k represents the source node and d^k is the destination node of request r^k. For each request r^k, it is necessary to determine its associated path set and purification set in order to satisfy the routing and fidelity requirements. A given request r^k may have multiple paths, each with a corresponding purification strategy, denoted by P^k and D^{pur}, where $P^k = \left\{p_m^k | m \leq M^k\right\}$ and $D^{pur} = \left\{D_{k,m}^{pur} | m \leq M^k\right\}$. p_m^k represents the m-th path of request r^k with the corresponding purification strategy $D_{k,m}^{pur}$, and M^k represents the total number of paths for request r^k.

Our goal is to allocate resources to the requests and maximize throughput while satisfying the fidelity threshold. The objective function is formed as follows:

$$\begin{aligned} \max \quad & \sum_k \sum_m ET_{k,m}^{E2E} \\ \text{s.t.} \quad & ET_{k,m}(u,v) \leq R_{(u,v)}^{k,m}, \quad \forall (u,v) \in E \forall k, m, \\ & \sum_k \sum_m b_{(u,v)}^{k,m} r_{(u,v)}^{k,m} \leq R_{(u,v)}, \quad \forall (u,v) \in E, \\ & r_{(u,v)}^{k,m} \in \mathbb{Z}_+, \quad \forall (u,v) \in E, \forall k, m. \end{aligned} \tag{8}$$

Here, $ET_{k,m}^{E2E}$ denotes the E2E throughput of the m-th path for request r^k, while $ET_{k,m}(u,v)$ represents the throughput contribution of an individual link $l(u,v)$ on that path. The binary variable $b_{(u,v)}^{k,m}$ indicates whether path m of request r^k traverses link $l(u,v)$, and $R_{(u,v)}$ is the available capacity of that link. The decision variable $r_{(u,v)}^{k,m}$ specifies the amount of entanglement resources allocated to the path on link $l(u,v)$. The objective maximizes the total throughput across all requests. The first constraint bounds each path's throughput by the allocation on its bottleneck link. The second ensures that total allocations on each link do not exceed its capacity. The third enforces integrality, reflecting that entangled pairs are discrete resources.

In classic networks, the problem of routing multiple source-destination pairs is classified as a multi-commodity flow problem and has been shown to be NP-hard [31]. Furthermore, in quantum networks, the entangled routing problem is coupled with the purification scheme, which further increases its complexity. In the above problem model, to achieve E2E entangled routing that satisfies the fidelity threshold, the problem requires not only determining the optimal paths for the set of requests Req, but also devising a corresponding purification scheme that specifies where and how purification should be performed along those paths.

Algorithm 1 SFTRAP Algorithm for a Single S-D Pair

Input: Network $G = (V, E)$, Request $r = (s, d)$, Fidelity threshold $F_{k,m}^{th}$.
Output: Selected paths P_{sel}^k, Strategies D_{sel}^{pur}, Total throughput ET_{total}.

Step 1: Initialization
Prune graph G to G' by removing links unable to meet $F_{k,m}^{th}$;
Initialize an empty priority queue Q for candidate paths, ordered by cost;
Step 2: Path Selection
Find candidate paths P_{cand}^k from s^k to d^k in G' using K-Shortest Path;
Step 3: Purification Strategy Execution
foreach *path* $p_m^k \in P_{cand}^k$ **do**
  while *current* $F_{k,m}^{E2E} < F_{k,m}^{th}$ **do**
    Select link $l(u,v) \in p_m^k$ that maximizes the purification metric $D_{k,m}^{pur}$;
    Update the strategy $D_{k,m}^{pur}$ and recalculate $F_{k,m}^{E2E}$;
  Calculate total path cost and push $(p_m^k, D_{k,m}^{pur}, \text{cost})$ to Q;
Step 4: Resource Allocation & Throughput Update
while Q *is not empty* **do**
  Pop the lowest-cost path-strategy pair $(p_m^k, D_{k,m}^{pur})$ from Q;
  Calculate path throughput $ET_{k,m}^{E2E}$;
  if *path is feasible and resources are available* **then**
    Allocate resources and update the network state G;
    Update output sets P_{sel}^k, D_{sel}^{pur} and total throughput ET_{total};
return $P_{sel}^k, D_{sel}^{pur}, ET_{total}$;

IV. SFTRAP for Single S-D Pair

In this section, we present the SFTRAP algorithm, which is designed to solve the problem of establishing E2E entanglements under single request scenario.

A. Design Overview

Given a routing request r^k and a quantum network topology $G = (V, E)$, we aim to find a routing solution that maximizes the throughput, satisfies the fidelity threshold and improves the algorithm's computational efficiency. We propose the SFTRAP algorithm to achieve these objectives. The algorithm follows

the core steps outlined below: The algorithm begins by exploring all feasible paths between the source node s^k and the destination node d^k. For each path, the optimal entanglement purification scheme is then determined to ensure transmission quality. Next, the algorithm evaluates the total cost of each path, taking into account both the cost of establishing entanglement and the cost of purification. Based on these evaluations, the paths are sorted in ascending order of cost, forming a ranked list. The algorithm then iterates through this list and selects the lowest-cost paths for resource allocation. This step includes removing the used resources and updating the network topology accordingly. After allocating resources to the selected path, the algorithm removes that path from the list and proceeds to the next iteration, considering the remaining paths until all paths have been visited. The detailed, step-by-step implementation of this process is presented in Algorithm 1.

B. Satisfying Fidelity Threshold for Routing and Purification

The SFTRAP under the single S-D pair scenario consists of the following 4 steps:

1) Initialization: Before executing the routing strategy, we use a pre-generated model to establish entanglements in the network. Each node attempts to establish entanglements with its neighboring nodes. For all successfully established entanglements, we maintain a mapping table that records the relationship between fidelity improvement, purification rounds, and resource consumption for each link. According to the pumping purification protocol, a link with R_e entangled pairs can support at most $N_{max} = R_e - 1$ purification rounds, where one pair serves as the target state and the remaining $R_e - 1$ pairs act as auxiliary states. After the maximum number of purification rounds, if the fidelity is still below the threshold, the link should be removed. Finally, based on the successfully established entanglements, we construct an undirected topology graph $G^r = (V, E^r)$, representing the network structure for routing.

2) Path Selection Strategy: To identify all routing paths from the source node s^k to the destination node d^k, we adopt the following path selection strategy. First, the Dijkstra algorithm is used to find the shortest path with a length of l^{min}. Then, an iterative algorithm is designed using the path length as the iterative metric. The iterative metric l starts from the shortest path length l^{min} and progressively increases until it reaches the longest path (i.e., $|V|-1$,$|V|$ represents a total number of nodes). At each iteration step, the K-shortest path algorithm [33] is used to find all paths with length equal to l. We gradually relax the path length constraint to identify all routing paths from the source node s^k to the destination node d^k. Finally, purification operations are performed on the obtained paths in order to satisfy the fidelity requirements.

3) Design of Purification Strategy: The E2E entanglement fidelity is derived from the post-purification fidelities of all entangled links in the path, following the standard Werner-state mapping under ideal BSMs. It is calculated using equation (7):

$$F_{k,m}^{E2E} = \frac{1}{4} + \frac{3}{4} \prod_{(u,v)\in p_m^k} \left(\frac{4\, F_{k,m}^{pur}(u,v,N^{pur}) - 1}{3} \right). \quad (9)$$

Here, $F_{k,m}^{pur}(u,v,N^{pur})$ denotes the fidelity on link $l(u,v)$ after N^{pur} purification rounds along p_m^k.

Entanglement purification is viewed as a processing unit with inputs and outputs, where the fidelity of the input entanglements directly affects the fidelity of the output. Consequently, selecting different links for purification along the path p_m^k can have varying effects on E2E fidelity. Previous studies [26], [27] adopted alternative purification strategies aimed at satisfying the fidelity requirements of communication requests. However, these approaches fail to guarantee an optimal purification scheme. We illustrate this limitation with the following example:

Example: To address the challenge of fidelity guarantee in entanglement routing, several purification strategies are compared in Fig. 2. qpath (Fig. 2a) employs an iterative greedy method, selecting links that maximize E2E fidelity improvement at each step. While effective for individual paths, it is resource-agnostic and may overload critical links, creating bottlenecks. qleap (Fig. 2b) sets an average per-hop fidelity target $F_{\text{avg}} = (F_{k,m}^{th})^{1/l}$ based on the E2E threshold $F_{k,m}^{th}$ and path length l. This simplifies the protocol and improves efficiency but sacrifices optimality, often causing over-purification and reduced throughput. Since NSPS lacks a dedicated purification mechanism [32], qleap's approach is applied for its evaluation. PU (Fig. 2c) instead performs static, network-wide pre-purification below a global threshold $F_{k,m}^{th}$, which frequently wastes resources and limits support for demanding requests due to the difficulty of choosing an appropriate universal threshold. In contrast, our proposed SFTRAP (Fig. 2d) integrates routing with a dynamic, resource-aware decision mechanism. By jointly considering fidelity gain, resource availability, and link reuse, and by adapting purification rounds, SFTRAP avoids bottlenecks and maximizes throughput while ensuring fidelity requirements, offering a more practical and holistic solution for quantum networks.

To determine which link on a path should be prioritized for purification at the N^{pur}-th potential round, we introduce the purification decision metric $D_{k,m}^{pur}(u,v,N^{pur})$. This metric is calculated based on three key factors: the potential fidelity improvement from this round, the available resources on the link, and the degree of link reuse. A higher value of $D_{k,m}^{pur}$ indicates a higher priority for performing the N^{pur}-th round of purification on the link $l(u,v)$ for path p_m^k. The normalization and scoring are computed in a per-path, per-round manner to reflect the current state. The calculation is as follows:

$$\begin{aligned} D_{k,m}^{pur}(u,v,N^{pur}) &= a \cdot \delta_{k,m}(u,v,N^{pur}) \\ &\quad + b \cdot R_{(u,v)} - c \cdot E(u,v) \end{aligned} \quad (10)$$

where:

$$\begin{aligned} &\delta_{k,m}(u,v,N^{pur}) \\ &= F_{k,m}^{pur}(u,v,N^{pur}) - F_{k,m}^{pur}(u,v,N^{pur}-1) \end{aligned} \quad (11)$$

$F_{k,m}^{pur}(u,v,N^{pur})$ and $F_{k,m}^{pur}(u,v,N^{pur}-1)$ denote the fidelities of link $l(u,v)$ after N^{pur} and $N^{pur}-1$ rounds of purification, respectively. Their difference, $\delta_{k,m}(u,v,N^{pur})$, represents the fidelity improvement in the N^{pur}-th round.

When $N^{pur} = 1$, it reflects the increase from the initial fidelity after one round of purification. $R_{(u,v)}$ denotes the number of available entangled pairs on link $l(u, v)$, while the reuse degree $E(u, v)$ represents the number of paths p_m^k that traverse this link, as defined below:

$$E(u, v) = \sum_{k,m} p_m^k(u, v) \tag{12}$$

The aforementioned parameters consider aspects of fidelity improvement, resource availability, and link reuse. a, b, and c are the weight parameters, satisfying $a + b + c = 1$. Weights a and b act positively, favoring fidelity gain and available resources, while c acts negatively, penalizing link reuse to reduce purification priority on bottlenecks. Since the magnitudes of $\delta_{k,m}(u, v, N^{pur})$, $R_{(u,v)}$, and $E(u, v)$ differ, with $\delta_{k,m}(u, v, N^{pur}) < 1$ and $R_{(u,v)}, E(u, v) > 1$, it is necessary to normalize these parameters to ensure a uniform scale across the model. We adopt path-round max-normalization, recomputed for each candidate path p_m^k and each purification round N^{pur}, as follows:

$$D_{k,m}^{pur}(u, v, N^{pur}) = a \cdot \frac{\delta_{k,m}(u, v, N^{pur})}{\delta_{k,m}^{\max}(N^{pur})} + b \cdot \frac{R_{(u,v)}}{R_{k,m}^{\max}} - c \cdot \frac{E(u, v)}{E_{k,m}^{\max}} \tag{13}$$

where $\delta_{k,m}^{\max}(N^{pur})$ denotes the maximum fidelity improvement achievable after N^{pur} rounds of purification on the current path p_m^k. $R_{k,m}^{\max}$ represents the maximum available resources on p_m^k in the current round, while $E_{k,m}^{\max}$ indicates the highest level of link reuse on p_m^k in the current round. In the multi-request scheduler, $E(u, v)$ is computed over the currently active candidate path pool across all pending requests in the present scheduling iteration and is recomputed after each resource allocation or resource update; in the single-request case, it reduces to the request's own candidate set. For each link on path p_m^k, we compute $D_{k,m}^{pur}(u, v, N^{pur})$ and prioritize links with the highest values; this selection proceeds iteratively until the path satisfies the required fidelity threshold.

a) Adaptive purification strategy design: In the previous section, we devised a purification strategy, where each link is equipped with multiple entanglement pairs, enabling multiple rounds of purification. Therefore, the strategy must identify the optimal link for purification, considering the number of rounds required. As a result, the purification algorithm has a worst-case time complexity of $O(|E|R_{max})$, where R_{max} denotes the maximum number of resources on the links. Moreover, the fidelity of entanglement pairs decays over time. This section aims to address the challenge of inefficiency in purification algorithms caused by the decay of fidelity and the number of purification rounds.

Theorem 1: For the link $l(u, v)$, the improvement in fidelity gradually diminishes as the number of purification rounds increases.

$$F_{n+1} - F_n < F_n - F_{n-1}$$

Theorem 1, the proof of which is provided in Appendix A, states that for the link $l(u, v)$ the improvement in fidelity gradually diminishes as the number of purification rounds increases. ■

Theorem 1 indicates that purification yields significant fidelity gains in the initial rounds, but improvements diminish with further iterations. Motivated by this, we design an adaptive purification strategy that adjusts operations to real-time fidelity requirements and network conditions. The key idea is to iteratively purify all links in a path p_m^k until the required fidelity is met or resources are exhausted, thereby optimizing resource usage and improving efficiency. Specifically, for a given path p_m^k, if its fidelity $F_{k,m}$ is below the threshold $F_{k,m}^{th}$, purification is triggered. An initial round is performed based on $D_{k,m}^{pur}(u, v, 1)$, and if $F_{k,m}$ remains insufficient, additional rounds are executed iteratively under $D_{k,m}^{pur}(u, v, n)$. Since overly strict fidelity requirements can result in an excessive number of purification rounds and wasted resources, it is essential to estimate an upper bound on the number of rounds required. This upper bound is provided by Theorem 2.

Theorem 2: For a path p_m^k with length n and fidelity threshold $F_{k,m}^{th}$. To simplify notation, let the link $l(u_n, v_n)$ be referred to as link l_n, thus the initial fidelities of the links $l_1, l_2, \ldots, l_n$ satisfy $F_1 < F_2 < \ldots < F_n$. The upper bound for the number of purification rounds N is given by:

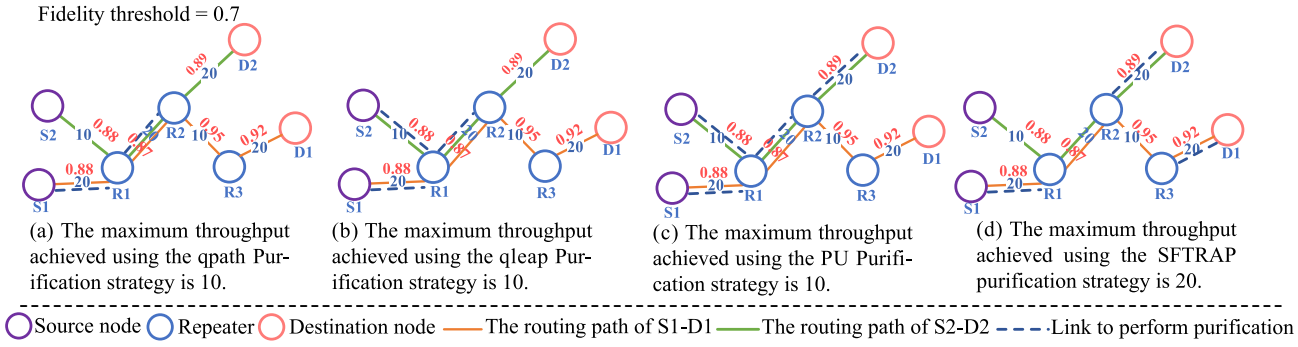
$$N = \left\lceil \frac{\log\left(\frac{3(1-\alpha_{target})}{1+3\alpha_{target}}\right)}{\log\left(\frac{1}{F_1} - 1\right)} - 1 \right\rceil$$

where the per-link target fidelity parameter, α_{target}, is defined as:

$$\alpha_{target} = \left(\frac{4F_{k,m}^{th} - 1}{3}\right)^{1/n}$$

Theorem 2, proved in Appendix A, establishes a relationship between the path length n, the fidelity threshold $F_{k,m}^{th}$, and the upper bound N on the number of purification rounds required for a given path p_m^k. ■

b) Implementation of the purification algorithm: To meet the E2E fidelity requirement, the purification procedure is performed iteratively and adaptively. The algorithm first selects the shortest path p_m^k from the candidate path set. It then identifies the optimal link for purification based on equation (13), using an adaptive strategy to accelerate convergence. During each iteration, the expected fidelity and corresponding resource consumption are recorded. This process continues until either $F_{k,m}^{E2E} \geq F_{k,m}^{th}$ is satisfied or the number of purification rounds reaches the theoretical limit N. The relevant purification data is stored in queue Q for use in later scheduling. SFTRAP supports heterogeneous fidelity requirements by assigning each request a specific fidelity threshold $F_{k,m}^{th}$. The purification process is triggered dynamically and adjusted iteratively according to this threshold. This ensures that each entanglement path satisfies the individual fidelity requirement. It is important to note that SFTRAP does not perform actual resource allocation during this phase. Instead, it conducts a theoretical evaluation based on current resource availability and the estimated fidelity of path p_m^k. This assessment informs the resource allocation decisions carried out in the subsequent phase.



(a) The maximum throughput achieved using the qpath Purification strategy is 10.

(b) The maximum throughput achieved using the qleap Purification strategy is 10.

(c) The maximum throughput achieved using the PU Purification strategy is 10.

(d) The maximum throughput achieved using the SFTRAP purification strategy is 20.

Fig. 2. Comparing with Different Purification Strategy.

4) Path Cost Update and Throughput Update: The main goal of the path cost update scheme is to select the path p_m^k with the lowest expected resource consumption from the queue Q and allocate entanglement resources to it. At the same time, the expected throughput of the path is recorded and the allocated resources are removed from the network.

Due to potential failures in the purification process, the actual number of high-fidelity entanglements will be lower than in the ideal case where all purification operations are successful. Therefore, for path p_m^k, we use the expected throughput $ET_{k,m}(u,v)$ to evaluate the number of entanglements on link $l(u,v)$, which is given by:

$$pairs_{k,m}(u,v) = \left\lfloor \frac{r_{(u,v)}^{k,m}}{N^{pur}+1} \right\rfloor,$$
$$ET_{k,m}(u,v) = P_{k,m}^{N^{pur}}(u,v) \times pairs_{k,m}(u,v), \quad (14)$$

where $pairs_{k,m}(u,v)$ is the number of purification attempts available on $l(u,v)$ given the allocated resources $r_{(u,v)}^{k,m}$, and $N^{pur}+1$ reflects the pumping protocol that consumes one working pair and N^{pur} ancillary pairs per output. $P_{k,m}^{N^{pur}}(u,v)$ denotes the multi-round success probability on $l(u,v)$ and is computed by equation (3). Additionally, the throughput of the path p_m^k is shown in equation (15):

$$ET_{k,m}^{E2E} = \min\{ET_{k,m}(u,v) \mid (u,v) \in E\} \quad (15)$$

Equation (15) shows that the expected throughput of a path is determined by the expected throughput of the smallest link on the path. Finally, the path throughput is updated, and the consumed resources at link $l(u,v)$ on path p_m^k are removed, as shown below:

$$\mathrm{Cost}_{k,m}(u,v) = \left\lceil \frac{N^{pur}+1}{P_{k,m}^{N^{pur}}(u,v)} \times ET_{k,m}^{E2E} \right\rceil \quad (16)$$

Here, $\mathrm{Cost}_{k,m}(u,v)$ is the number of resources to be deleted from each link $l(u,v)$ in the path. Additionally, $\frac{N^{pur}+1}{P_{k,m}^{N^{pur}}(u,v)}$ represents the number of resources that need to be consumed to obtain high-fidelity entanglements. Resources consumed in the network are deleted according to equation (16). Finally, we output the set of paths, the purification scheme, the fidelity of the paths, and the throughput.

C. Time Complexity Analysis

Let $|E|$ denote the number of edges; $|V|$, the number of nodes; $R_{\max}$, the maximum number of entangled pairs per edge; and $|l^{min}|$, the shortest path length. The computational complexity of SFTRAP can then be analyzed as follows. In the first stage, constructing the purification–cost mapping requires $O(|E|R_{\max})$. In the second stage, path search is performed by computing up to K candidate paths using a K-shortest-paths procedure [33]. Under our implementation, the worst-case cost of a single run is $O(K|V|(|E|+|V|\log|V|))$, where the logarithmic factor originates from Dijkstra's algorithm with a priority queue [37]. Since this procedure may iterate up to $|E|-|l^{min}|$ times, the total worst-case complexity of this stage becomes $O\big(|V|^2 + K|V|(|E|-|l^{min}|)(|E|+|V|\log|V|)\big)$. In the third stage, fidelity checking and purification decisions can be completed in $O(|E|)$. Finally, in the fourth stage, throughput update and resource deletion over K paths take $O(K|E|)$. Combining all stages, the overall worst-case computational complexity T_{single} of SFTRAP is $O\big(|E|R_{\max}+|V|^2+K|V|(|E|-|l^{min}|)(|E|+|V|\log|V|)+K|E|\big)$, which can be approximated asymptotically as:

$$O\big(K|V||E|(|E|+|V|\log|V|)\big).$$

V. SFTRAP for Multiple S-D Pairs

A. The Design of SFTRAP Under Multiple S-D Pairs Scenario

Given a quantum network topology $G = (V,E)$ and multiple S-D pairs of routing requests, our objective is to identify a routing solution that maximizes network throughput and meets fidelity requirements. This solution involves both a routing scheme and a purification scheme. For routing problems with multiple S-D pairs, we approach them as combined problems, with each S-D pair treated individually. If network resources are sufficient, the final solution will be a combination of these individual S-D pair solutions. The choice of combination strategy, such as the order in which resources are allocated to each S-D pair, significantly impacts the algorithm's performance. We use two crucial allocation metrics as decision variables for resource allocation based on

Algorithm 2 SFTRAP for Multiple S-D Pairs

Input: Network $G = (V, E)$, Request set $Req = \{r^1, ..., r^n\}$.
Output: Path set $\sum_k P^k$, Purification schemes $\sum_k D^{pur}_{k,m}$, Fidelity set $\sum_k F^{E2E}_{k,m}$, Throughput $\sum_{k,m} ET^{E2E}_{k,m}$.

Step 1: Initialization
For each $r^k \in Req$, initialize a priority queue Q^k;
Initialize a global priority queue Q_{global};
Step 2: Routing and Purification Predetermination
foreach $r^k \in Req$ **do**
  Find path-strategy pairs $(p^k_m, D^{pur}_{k,m})$ via single-pair SFTRAP;
  Push pairs to Q^k, ordered by scheduling metric $U_{k,m}$;
Push the top element from each non-empty Q^k to Q_{global};
while Q_{global} *is not empty* **do**
  Step 3: Resource Allocation and Throughput Update
  Pop highest-priority pair $(p^k_m, D^{pur}_{k,m})$ from Q_{global};
  Calculate throughput $ET^{E2E}_{k,m}$ for path p^k_m;
  if $ET^{E2E}_{k,m} \geq 1$ **then**
    Allocate resources for p^k_m on graph G;
    Update output sets $P^k, D^{pur}_{k,m}, F^{E2E}_{k,m}, ET^{E2E}_{k,m}$;
    Remove $(p^k_m, D^{pur}_{k,m})$ from its local queue Q^k;
    if *request* r^k *is satisfied* **then**
      Clear all entries for r^k from all queues;
  Step 4: Re-routing Process
  Update Q_{global} with the new top element from the modified queue Q^k;
return $\sum_k P^k$, $\sum_k D^{pur}_{k,m}$, $\sum_k F^{E2E}_{k,m}$, $\sum_{k,m} ET^{E2E}_{k,m}$;

[27], namely degrees of freedom and resource consumption. Which is illustrated as follows:

$$\begin{aligned} U_{k,m} &= \alpha \cdot G\left(p^k_m\right) + \beta \cdot S\left(p^k_m, D^{pur}_{k,m}\right) \\ G\left(p^k_m\right) &= \sum_{\text{nodes on } p^k_m} \mathcal{N}(u) \\ S\left(p^k_m, D^{pur}_{k,m}\right) &= \sum_{l(u,v)\in p^k_m} \text{Cost}_{k,m}(u,v). \end{aligned} \tag{17}$$

The degrees of freedom $G\left(p^k_m\right)$ represent the total freedom of the path p^k_m, calculated as the cumulative sum of the adjacent node counts for each node along the path. A higher degree of freedom indicates a greater potential for rerouting. Meanwhile, the resource consumption $S\left(p^k_m, D^{pur}_{k,m}\right)$ represents the entanglement resources consumed by the specific path p^k_m. Less resource consumption means that more resources are allocated to other paths. The weighting coefficients α and β act as factors to balance the weight of degrees of freedom and resource consumption. When both α and β are set to 1, the network achieves optimal throughput [27]. The overall workflow for handling multiple requests, including the priority queue management and iterative resource allocation, is formally outlined in Algorithm 2.

SFTRAP under the multi-request scenario contains the following steps. First, we use the SFTRAP algorithm to search for routing paths and purification schemes for multiple requests $r^1, r^2 \dots r^n$. These paths and schemes are then stored in sets $Q^1, Q^2 \dots Q^n$. The problem model is abstracted as the merging of multiple ordered subsets into a single ordered subset. Furthermore, we adopt a priority queue approach to optimize the time complexity.

The steps to achieve this are as follows:

1) Initializing the network: Firstly, we create multiple priority queues $Q^1, Q^2 \dots Q^n$ and Q^{min} to store paths and their corresponding purification strategies.

2) Predetermine routing and purification: The SFTRAP algorithm is used to find the routing paths and purification schemes as solutions for multiple requests $r^1, r^2 \dots r^n$. Then the $U_{k,m}$ of each solution is computed according to equation (18), and the above solutions are ordered by $U_{k,m}$ in $Q^1, Q^2 \dots Q^n$. Finally, the solution with the smallest $U_{k,m}$ is then moved to Q^{min}.

3) Resource allocation and throughput update: First, we select the path p^k_m with minimum $U_{k,m}$ from Q^{min} and the purification strategy $D^{pur}_{k,m}$. Next, the link with the least resources is identified and recorded as $pairs^{min}_{k,m}$ on the path p^k_m, where $pairs_{k,m}(u,v)$ is computed based on equation (14). Subsequently, the maximum throughput of the path p^k_m is computed based on equation (15). Then, the consumed resources are removed along the path p^k_m if the path p^k_m is able to establish at least one pair of long-distance quantum entanglement. Here, $\min\left\{ET^{E2E}_{k,m}, tp_k\right\}$ denotes the minimum of the path throughput $ET^{E2E}_{k,m}$ and the required throughput tp_k for request r^k. Meanwhile, $\frac{N^{pur}+1}{P^{N^{pur}}_{k,m}(u,v)}$ denotes the consumed resources to establish a pair of high-fidelity entanglement on link $l(u,v)$. If the throughput of path p^k_m already satisfies the request r^k, the path and the purification strategy in Q^k are removed. So the resource allocation and throughput are finally updated.

4) Re-routing process: After updating the resources in the network, the SFTRAP algorithm is again used to find out if there are still resources for routing schemes and purification schemes that satisfy the request, and add it to the relevant request queue and wait for resources to be allocated to it.

B. Time Complexity Analysis

The computational complexity of Algorithm 2 is analyzed as follows. The algorithm iterates over n requests $r^1, \dots, r^n$. For each request, Step 2 (routing and adaptive purification) invokes the single-request procedure with worst-case cost T_{single} (as derived previously). Step 3 (resource allocation over the paths found in Step 2) costs $O(nK|E|)$. Step 4 (re-routing) is executed at most n times; each re-routing may trigger at most one additional single-request solve, hence its worst-case cost is $O(n\,T_{\text{single}})$. Therefore, the overall complexity is $O\big(n\,T_{\text{single}} + nK|E| + n\,T_{\text{single}}\big) = O\big(n\,T_{\text{single}} + nK|E|\big)$.

TABLE II
SIMULATION CONFIGURATION

Configuration	Details
CPU	Intel i5-13400 2.50 GHz
RAM	16 GB
Operating System	Windows 10 64-bit
Simulation Basis	US backbone network [34]
Link Bandwidth	Equal bandwidth for all links
Fidelity (Same Link)	Uniform, denoted as $F(u, v)$
Fidelity (Across Links)	$N[0.8, 0.1]$, clipped to [0.7, 1.0]
Synchronization time step	500 ms

Using the previously established bound for a single request, this yields the final asymptotic bound

$$O\big(nK|V||E|(|E| + |V|\log|V|)\big).$$

VI. PERFORMANCE EVALUATION

A. Experimental Environment

The simulations are based on the U.S. backbone network topology [34], comprising 39 nodes and 121 links. To adapt it to a quantum network context, we introduce inter-link heterogeneity by sampling the initial fidelity of each link from a Gaussian distribution $N(0.8, 0.1)$, assigning different fidelities to different links to reflect their heterogeneity, while entangled pairs on the same link share the same fidelity value due to device consistency. To ensure validity, any sampled fidelity value outside the range [0.7, 1.0] is discarded and resampled until it falls within this range. The baseline link capacity, defined as the number of entangled pairs generated per time window Δt, is 50, and each node has 200 quantum memory units. For all algorithms, the K value in the K-shortest path search is fixed at 20, and identical random seeds are used to generate the same source–destination request sets and link fidelities across 1000 runs. Simulations proceed in discrete time slots, where each slot corresponds to a 500 ms synchronized step coordinated by a centralized controller, and terminate when no further entanglement resources can be allocated. The selection rule for equal-cost paths or requests is deterministic: the lower-indexed source is selected first. For fairness, NSPS, which lacks a purification mechanism, is augmented with qleap's per-link target strategy: the E2E fidelity threshold is mapped to uniform per-link targets, and each link is purified until its target is met. Additional configuration details are summarized in Table II, and all results are averaged over 1000 independent runs to ensure statistical reliability.

B. Benchmarks

To comprehensively evaluate the performance of the proposed SFTRAP algorithm, we compare it against four benchmark methods: qpath, qleap [27], PU [26], and NSPS [32]. Table III provides a comparison of these algorithms, focusing on purification strategies, resource awareness, and time complexity. It also summarizes their respective advantages, limitations, and design trade-offs. The relevant variable definitions are as follows: N denotes the number of requests, K denotes the number of paths considered for each request, and ϵ denotes the approximation accuracy factor. In terms of time complexity, qpath exhibits the highest complexity, falling into the high-order polynomial category. qleap demonstrates the lowest complexity, achieving near-linear scalability. PU shows polynomial complexity. SFTRAP has high-order polynomial complexity as well, but with adaptive characteristics. NSPS also operates within polynomial time complexity as an approximation scheme. Notably, SFTRAP's dynamic purification mechanism introduces some variability in connection establishment time. However, it enables flexible adaptation to heterogeneous link qualities and diverse application demands. In contrast, fixed-round approaches often struggle to accommodate these variations efficiently. Additionally, SFTRAP's resource awareness prevents overburdening congested links, which improves load balancing.

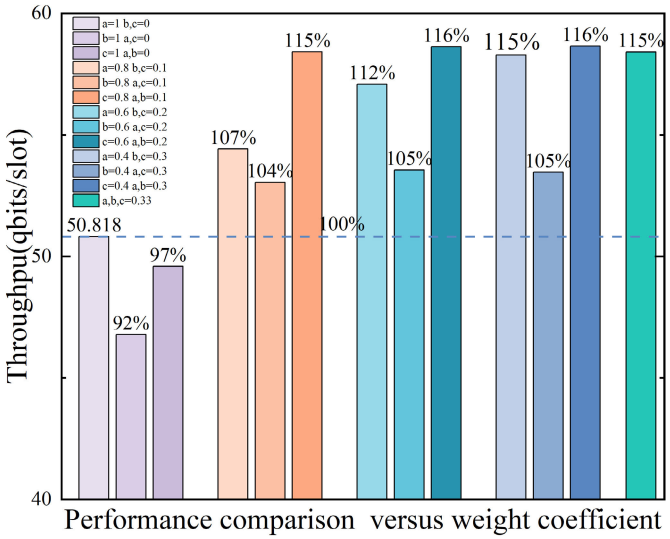


Fig. 3. Impact of different weight coefficients on throughput (fidelity threshold = 0.7, channel capacity = 50, node capacity = 200, the performance of qpath [27] is used as the 100% normalization baseline).

C. Results Under Different Weight Parameters

We conducted 1000 simulation experiments to evaluate the influence of different weight coefficient on the throughput performance. The results are shown in Figure 3.

When a = 1, b = c = 0, the purification strategy described above is consistent with the qpath [27], (i.e., selecting the link with the largest improvement in E2E fidelity to purify.) In this configuration, the achieved throughput is 50.818. To further analyze the impact of weight parameters on throughput, we conducted experiments and compared the results with the baseline of qpath. The normalized throughput values, as shown in this figure, demonstrate the influence of varying parameter settings. When considering link reuse in isolation, the throughput achieves 92%. Similarly, considering only the available resources on links results in a throughput of 97%. These results indicate that neither factor alone will lead to superior performance. However, when a = b = 0.2 and c = 0.6, or when a = b = 0.3 and c = 0.4, the normalized throughput

TABLE III
COMPARATIVE ANALYSIS OF ENTANGLEMENT ROUTING ALGORITHMS

Feature	**qpath** [27]	**qleap**[27]	**PU**[26]	**SFTRAP**	**NSPS**[32]
Core Purification Strategy	Iterative greedy; max E2E fidelity gain.	Sets average per-link fidelity target; purifies links individually to meet target.	Static global pre-purify based on a fixed fidelity threshold.	Dynamic adaptive; balances fidelity gain, available resources, and link reuse.	Combining qleap's purification strategy with an all-optical switching node bypass mechanism.
Resources Awareness	No (Ignores resources and congestion).	No (Ignores resources and congestion).	No (Static pre-purify; inefficient).	Yes (Explicitly considers resources and congestion).	No (Ignores resources and congestion).
Time Complexity	$O(\|E\|R_{max} \cdot (K\|V\|\|E\| + \|V\|^2 \log_2 \|V\| + K\|E\|R_{max}))$	$O(N(\|V\| \log_2 \|V\| + \|E\|) + \|E\|R_{max})$	$O(\|E\| \cdot N \cdot K \cdot R_{max})$	$O(K\|V\|\|E\|(\|E\| + \|V\| \log_2 \|V\|))$	$O(\|V\|^2 R_{max}^2 + \|V\|R_{max}/\epsilon)$
Advantages	Optimal single-path cost; potential high throughput.	Low complexity; simple logic.	uses global info.	Balances fidelity with throughput; avoids bottlenecks; efficient.	Maximizes success probability.
Disadvantages	High complexity; resource-blind bottlenecks.	Suboptimal; lower throughput.	Inefficient pre-purify; high complexity.	High theoretical complexity; needs parameters tuning.	Approximate solution

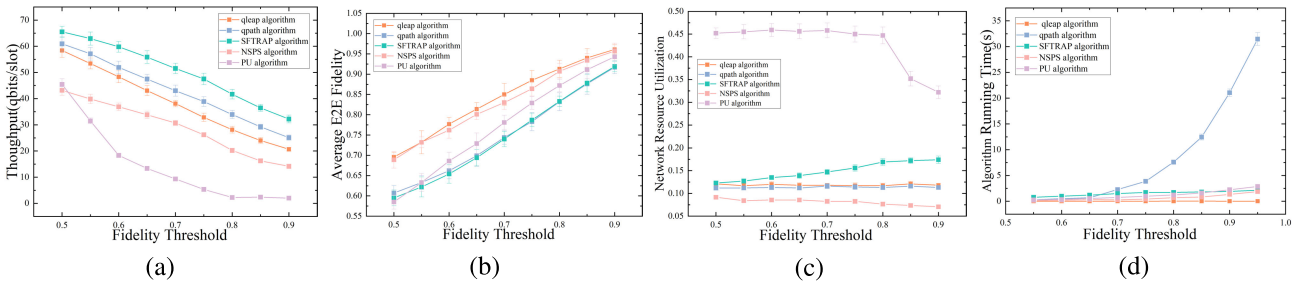


Fig. 4. Impact of fidelity threshold on key performance metrics. Compares algorithms based on (a) throughput, (b) average E2E fidelity, (c) network resource utilization, and (d) running time. Increasing fidelity demands generally reduces throughput due to higher purification costs, but SFTRAP consistently demonstrates a superior balance across all metrics, maintaining efficiency and higher throughput compared to benchmarks.

reaches 116%, highlighting the superior performance of our purification strategy. Therefore, for all other experiments, we used the weight values $a = 0.2$, $b = 0.2$, and $c = 0.6$, as they showed optimal performance in our sensitivity analysis.

D. Results Under Single S-D Pair Scenarios

1) Performance Comparison vs. Fidelity Threshold: As reported in [27], the channel capacity is 50, each node supports up to 200 quantum memories and the time slot is 500 ms. The "Network Resource Utilization" metric is defined as the ratio of the total entanglement resources consumed by all successful paths to the total resources generated across the entire network over the time slot. Figure 4 shows that throughput decreases for all algorithms as the fidelity threshold rises, since fewer entanglements meet the requirement and purification consumes more resources. Among all algorithms, SFTRAP consistently achieves the highest throughput by dynamically selecting purification links based on fidelity gain, resource availability, and link multiplexing, ensuring efficient resource allocation. In contrast, PU's static pre-purification wastes resources and yields the lowest throughput. NSPS, though adopting qleap's purification method, prioritizes success probability via optical switching and optimized paths, resulting in throughput only slightly higher than PU. In terms of fidelity, qleap performs best by converting the E2E threshold into per-link targets, enabling precise purification and surplus fidelity. NSPS achieves comparable performance due to its similar strategy. When the threshold is below 0.7, SFTRAP shows slightly lower fidelity than qpath, since qpath prioritizes links with the greatest fidelity improvement. Nevertheless, all algorithms consistently satisfy the required threshold.

Regarding resource utilization, PU maintains 33–45% utilization but drops sharply at a threshold of 0.8 due to path selection difficulties. SFTRAP achieves higher utilization than qpath and qleap, as its adaptive purification supports more E2E paths, improving throughput and efficiency. NSPS shows the lowest utilization, constrained by its success-probability-driven design and limited node memory. Execution time further distinguishes the algorithms: qpath requires multiple rounds as thresholds increase, leading to sharp growth in delay, while SFTRAP adapts rounds dynamically with an upper bound, ensuring efficiency. qleap, though efficient per link, sacrifices throughput due to excessive purification.

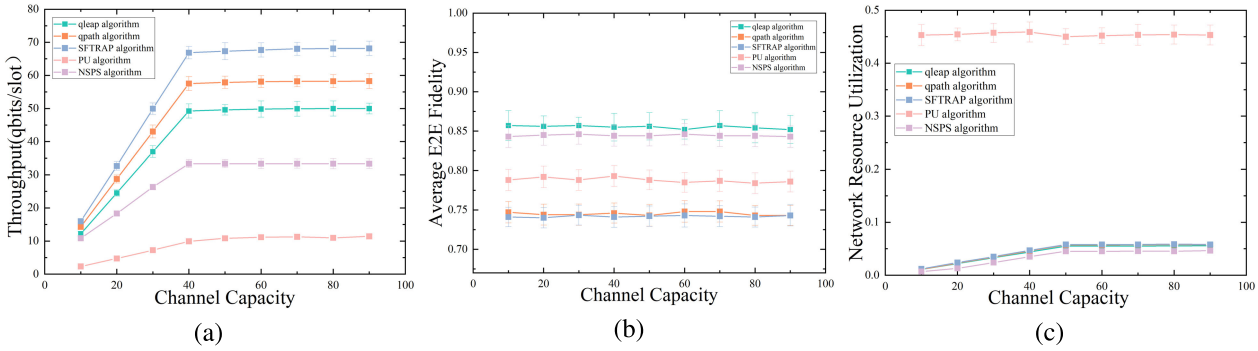


Fig. 5. Impact of channel capacity on key performance metrics. Compares algorithms based on (a) throughput, (b) average E2E fidelity, and (c) network resource utilization. While increased capacity boosts throughput for all algorithms until limited by node memory, SFTRAP demonstrates significantly better scaling and achieves the highest throughput, showcasing its ability to leverage available link resources effectively.

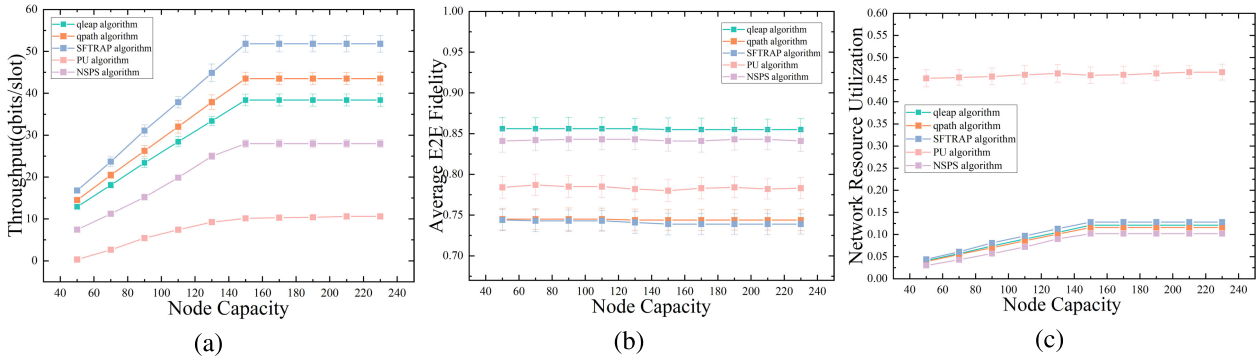


Fig. 6. Impact of node capacity on key performance metrics. Compares algorithms based on (a) throughput, (b) E2E fidelity, and (c) network resource utilization. Higher node capacity improves throughput until link capacity becomes the bottleneck; SFTRAP consistently achieves the highest throughput, indicating its efficiency in utilizing available quantum memory.

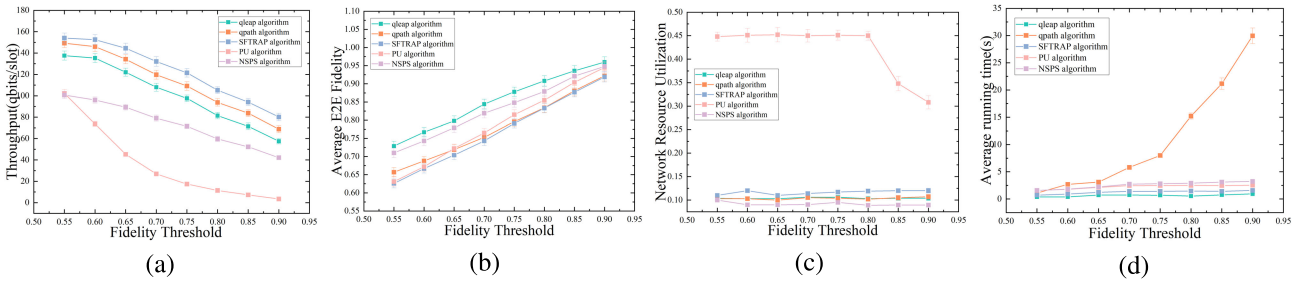


Fig. 7. Impact of fidelity threshold on key performance metrics for multiple S-D pairs. Compares algorithms based on (a) throughput, (b) average E2E fidelity, (c) network resource utilization, and (d) running time. As in the single-pair case, increasing fidelity demands reduces overall throughput (a), but SFTRAP maintains the highest throughput and stable efficiency (d), highlighting its robustness and effective resource allocation in concurrent request scenarios.

2) Performance Comparison vs. Channel Capacity: The link capacity, defined as the number of entanglements generated per time step Δt, is calculated by $v(u,v)q(u,v)\Delta t$. The fidelity threshold is set to 0.7, each node has 200 quantum memories, and the time slot is 500 ms. In Figure 5, throughput increases linearly with link capacity initially. At a link capacity of 10, the throughput for SFTRAP, qpath, qleap, PU and NSPS is 16.0, 14.3, 12.0, 2.6 and 10.2, respectively. As link capacity grows, the disparity among algorithms widens. At a link capacity of 40, throughput for SFTRAP, qpath, qleap, PU and NSPS reaches 68.2, 58.3, 50.0, 11.9 and 31.5, respectively, demonstrating SFTRAP's superior performance in resource-rich environments. Beyond this point, throughput plateaus due to quantum memory limitations. With a fidelity threshold of 0.7, the achieved fidelity for qleap, PU, qpath, SFTRAP and NSPS is approximately 0.85, 0.784, 0.745, 0.742 and 0.842, respectively. All algorithms meet the specified fidelity requirement. PU exhibits high resource utilization (0.452) due to significant pre-purification costs. For other algorithms, resource utilization rises from 0.03 at a link capacity of 10 to 0.1 at 40, as increased capacity allows for successful purification. Utilization plateaus beyond a capacity of 40, constrained by quantum memory bottlenecks.

3) Performance Comparison vs. Node Capacity: The node capacity refers to the size of the quantum memories in each node, with the capacity of each link set to 50. In Figure 6 (a), when the node capacity is 50 and the fidelity threshold is set to 0.7, the throughput for SFTRAP, qpath, qleap, PU and NSPS is

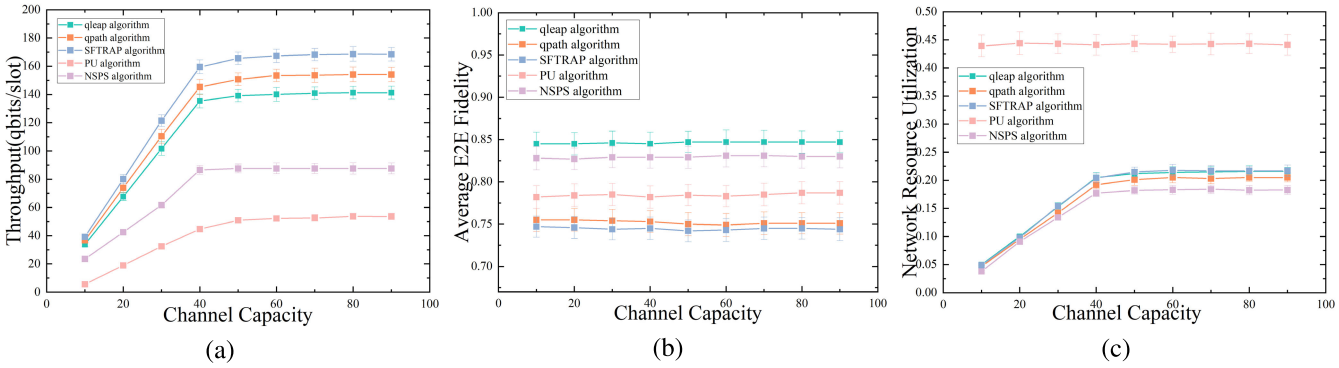


Fig. 8. Performance comparison versus channel capacity for multiple S-D pairs. Evaluates (a) throughput, (b) fidelity, and (c) resource utilization. SFTRAP achieves the highest overall throughput, scaling more effectively with increased link capacity compared to benchmarks.

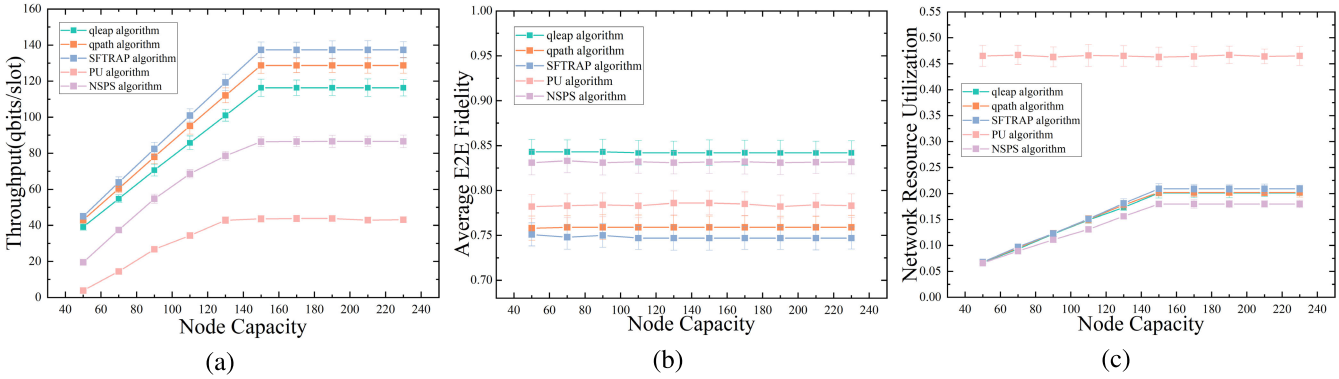


Fig. 9. Performance comparison versus node capacity for multiple S-D pairs. Evaluates (a) throughput, (b) fidelity, and (c) resource utilization. SFTRAP delivers the highest total throughput, demonstrating effective scaling with increased node memory.

16.8, 14.5, 12.9, 1.9 and 7.3, respectively. As the node capacity increases, the throughput of all five algorithms rises, with the differences becoming more pronounced. For instance, when the node capacity reaches 140, the throughput for SFTRAP, qpath, qleap, PU and NSPS is 51.8, 43.5, 38.4, 10.7 and 33.1, respectively. Notably, when the node capacity exceeds 140, link capacity becomes the bottleneck affecting overall network performance. In Figure 6 (b), the qleap algorithm maintains the highest fidelity, with the fidelity of PU, qpath, SFTRAP and NSPS at approximately 0.783, 0.744, 0.740 and 0.841, respectively. Importantly, all five algorithms meet the fidelity requirement. In Figure 6 (c), the resource utilization of PU reaches 0.452 and remains relatively constant, suggesting its performance is not primarily limited by node memory in this range. In contrast, the other four algorithms (SFTRAP, qpath, qleap and NSPS) show gradual improvement in resource utilization as node capacity increases. For example, at a node capacity of 50, the resource utilization of the four algorithms is approximately 4%. As the node capacity increases to 140, their resource utilization approaches 12%. This trend reflects their ability to establish more concurrent E2E entanglements by utilizing the larger available node memories, until the fixed link capacity limits further growth in utilization.

E. Results Under Multiple S-D Pairs Scenario

In this section, we have conducted simulation experiments to compare the performance of the SFTRAP, qpath, and qleap algorithms in handling multi-request scenarios. According to [27], the weighting coefficients α and β must be normalized, where $\alpha = \frac{\alpha^*}{2|E|}$ and $\beta = \frac{\beta^*}{|E|C_{\text{channel}}}$, where $C_{channel}$ represents the link capacity and the utility measures of α^* and β^* are both 0.5.

1) Performance Comparison vs. Fidelity Threshold: As shown in Figure 9, with a link capacity of 50, node capacity of 200, time slot of 500 ms and 4 requests. The throughput of SFTRAP, qpath, qleap, PU, and NSPS decreases as the fidelity threshold increases due to the reduced availability of high-quality entanglement paths and increased resource consumption for purification. Despite this, SFTRAP achieves the highest throughput. In Figure 7 (b), qleap, PU, and NSPS achieve higher fidelity compared to SFTRAP and qpath, though all algorithms maintain fidelity above the threshold. Figure 7 (c) indicates that PU exhibits the highest resource utilization, while SFTRAP, qpath, qleap and NSPS show similar levels of utilization. Figure 7 (d) demonstrates that qpath's purification strategy results in deep purification, causing its runtime to increase exponentially as fidelity thresholds rise. Although NSPS incurs additional computational overhead from auxiliary graph construction and approximation schemes, it still achieves superior execution efficiency compared to qpath. In contrast, SFTRAP employs an adaptive strategy that limits the number of purification rounds, ensuring a stable runtime. While qleap's exhaustive link purification maximizes

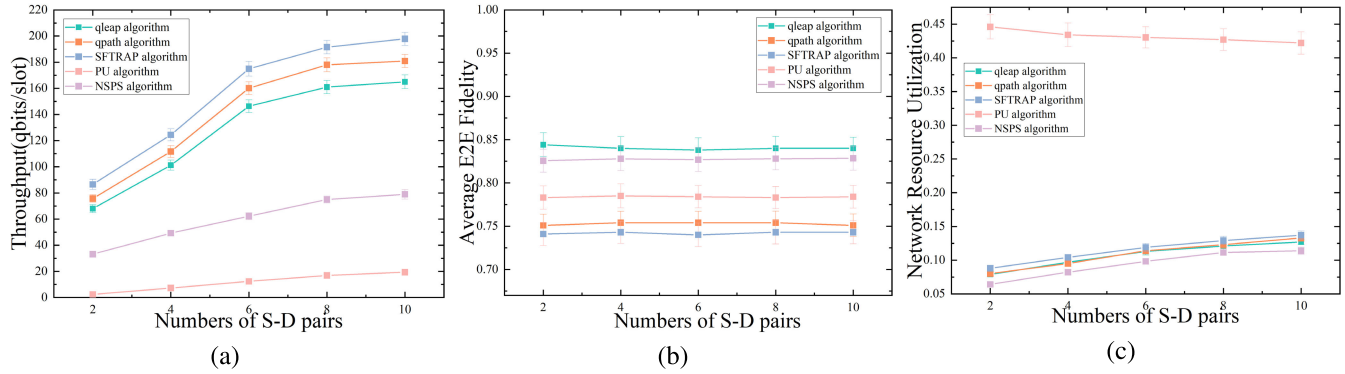


Fig. 10. Performance comparison versus the number of concurrent S-D pair requests. Evaluates (a) total throughput, (b) fidelity, and (c) resource utilization. SFTRAP sustains significantly higher total throughput as network load increases, handling resource contention more effectively than benchmarks.

efficiency, it results in the lower throughput and resource utilization.

2) Performance Comparison vs. Channel Capacity: In Figure 8, with a fidelity threshold of 0.7, node capacity of 200, and 4 requests, throughput increases linearly with link capacity initially, as more entanglement resources become available per link. At a link capacity of 10, the throughput for SFTRAP, qpath, qleap, PU and NSPS is 39.9, 38.6, 38.2, 3.4 and 23.2, respectively. At a link capacity of 40, throughput reaches 168.4, 154.1, 141.3, 57.8 and 87.6, respectively, highlighting SFTRAP's superior performance and scalability in resource-abundant scenarios. This demonstrates SFTRAP's effective resource management in leveraging increased link capacity to support higher aggregate throughput for multiple requests. Beyond this point, throughput plateaus due to quantum memory constraints. In Figure 8 (b), qleap achieves the highest fidelity (i.e., 0.85), while PU, qpath, SFTRAP and NSPS achieve fidelities of 0.784, 0.752, 0.742 and 0.830, respectively, meeting the threshold. Figure 8 (c) shows that PU's resource utilization peaks at 0.453, while the other algorithms' utilization gradually increases with link capacity, plateauing beyond 40 due to quantum memory limitations.

3) Performance Comparison vs. Node Capacity: In Figure 9, we set the fidelity threshold to 0.7, the link capacity to 50, and the number of requests to 4. In Figure 9 (a), with a node capacity of 50, the throughput of SFTRAP, qpath, qleap, PU and NSPS is 42.1, 41.7, 39.6, 3.4 and 19.7, respectively. As the node capacity increases, the throughput of all algorithms rises, with differences becoming more pronounced. For example, at a node capacity of 140, the throughput for SFTRAP, qpath, qleap, PU and NSPS is 137.3, 128.7, 116.3, 41.6 and 86.5, respectively. Beyond this point, link capacity becomes the primary bottleneck. In Figure 9 (b), qleap, NSPS and PU achieve high fidelity, while qpath and SFTRAP maintain similar fidelity levels. Again, achieved fidelity per path remains relatively stable despite varying node capacity. All algorithms meet the fidelity requirement. In Figure 9 (c), PU achieves a resource utilization of approximately 0.453, showing little dependence on node capacity in this range. The other algorithms exhibit increasing resource utilization with higher node capacities, rising from 7% at 50 to nearly 20% at 140. This increase reflects the successful establishment of a higher volume of E2E entanglements across the network enabled by larger node memories, until link capacity constraints limit further utilization growth.

4) Performance Comparison vs. Number of Requests: In Figure 10, we establish the fidelity threshold at 0.7, the link capacity at 50, and the node capacity at 200. In Figure 10 (a), total network throughput generally increases as the number of concurrent requests grows for all algorithms, but with a widening performance gap. SFTRAP consistently achieves significantly higher throughput, demonstrating its superior ability to manage resource contention and allocate resources effectively under increasing network load. Notably, the rate of throughput increase as more requests are added, indicating approaching network resource saturation. In Figure 10 (b), the achieved fidelity remains relatively stable and above the required threshold for all algorithms, even as the network load increases. In Figure 10 (c), PU's resource utilization remains relatively flat, while utilization for SFTRAP, qpath, qleap and NSPS progressively increases with the number of requests, reflecting greater overall network usage. Crucially, SFTRAP converts this increased utilization into substantially higher throughput (Fig 10(a)), highlighting its better efficiency in successfully establishing E2E paths under heavier load conditions.

VII. Conclusion

This paper proposes SFTRAP, a joint routing and purification scheme designed to address core challenges in quantum networks, including long-distance entanglement establishment, E2E fidelity assurance, and reduced computational complexity. SFTRAP employs the K-shortest path algorithm to identify multiple candidate paths for each request, aiming to maximize overall throughput. It jointly optimizes routing and purification to satisfy fidelity requirements while minimizing throughput degradation caused by purification. Moreover, SFTRAP introduces an adaptive purification mechanism that dynamically adjusts the number of purification rounds based on real-time fidelity demands, thereby improving decision-making efficiency. Future work will focus on modeling the decoherence process to more precisely capture the temporal evolution of fidelity in quantum channels and quantum

memories. Building on this foundation, we aim to design fair and robust routing algorithms capable of operating effectively in dynamically changing, decoherence-dominated quantum environments. Meanwhile, we plan to develop more efficient, batch-wise, and adaptive path search heuristics to enhance scalability.

APPENDIX A
PROOF OF THEOREM 1

Proof: We assume that the initial fidelity of all entanglements on the link $l(u,v)$ are F, where $F \in [0.5, 1]$. The fidelity after the 1-th and $(n+1)$-th round of purification is denoted by $F_1 = \frac{F^2}{F^2+(1-F)^2}$, $F_{n+1} = \frac{FF_n}{FF_n+(1-F)(1-F_n)}$ respectively. Inverting the above equation, we get:

$$\frac{1}{F_{n+1}} = 1 + \left(\frac{1}{F} - 1\right)\left(\frac{1}{F_n} - 1\right)$$

The equation can be rearranged as follows:

$$\left(\frac{1}{F_{n+1}} - 1\right) = \left(\frac{1}{F} - 1\right)\left(\frac{1}{F_n} - 1\right)$$

When $n = 0$, we have $\frac{1}{F_1} - 1 = \left(\frac{1}{F} - 1\right)^2$, which implies $\frac{1}{F_n} - 1 = \left(\frac{1}{F} - 1\right)^{n+1}$. Rearranging the equation further, we get:

$$F_n = \frac{1}{\left(\frac{1}{F} - 1\right)^{n+1} + 1}$$

Next, let's assume $b = \frac{1}{F} - 1$, where $b \in [0, 1]$. Substituting this into the equation, we have:

$$F_n = \frac{1}{b^{n+1} + 1}$$

According to the above equation, we can derive:

$$\frac{F_{n+1} - F_n}{F_n - F_{n-1}} = \frac{b\left(b^n + 1\right)}{b^{n+2} + 1}$$

$$\frac{b\left(b^n + 1\right)}{b^{n+2} + 1} - 1 = \frac{(b-1)\left(1 - b^{n+1}\right)}{b^{n+2} + 1} < 0$$

Therefore, we finally conclude that $F_{n+1} - F_n < F_n - F_{n-1}$. This completes the proof. ■

Proof of Theorem 2 Our goal is to find a conservative upper bound for the number of purification rounds, *N*, required for an *m*-link path to meet an E2E fidelity threshold, $F_{k,m}^{th}$. We assume F_1 is the lowest initial fidelity on the path.

The condition to be satisfied is $F_{k,m}^{E2E} \geq F_{k,m}^{th}$. Using the Werner-state swapping model, this becomes $\frac{1}{4} + \frac{3}{4}\prod_{i=1}^{m} \alpha_i^{(N)} \geq F_{k,m}^{th}$, which simplifies to $\prod_{i=1}^{m} \alpha_i^{(N)} \geq \alpha_{th}$, where $\alpha_i^{(N)} = (4F_i^{(N)} - 1)/3$ and $\alpha_{th} = (4F_{k,m}^{th} - 1)/3$.

To find a conservative bound, we solve the stricter inequality for the worst-performing link, whose fidelity improvement is the slowest:

$$\left(\alpha_1^{(N)}\right)^m \geq \alpha_{th}$$

This yields the required performance for the worst link, which we define as the per-link target, α_{target}:

$$\alpha_1^{(N)} \geq (\alpha_{th})^{1/m} = \alpha_{target}$$

The term $\alpha_1^{(N)}$ can be expressed in terms of *N* and the initial fidelity F_1 as $\alpha_1^{(N)} = \frac{3 - b_1^{N+1}}{3(1+b_1^{N+1})}$, where $b_1 = \frac{1}{F_1} - 1$. Substituting this into the target inequality and solving for *N* yields:

$$b_1^{N+1} \leq \frac{3(1 - \alpha_{target})}{1 + 3\alpha_{target}}$$

Taking the logarithm of both sides and noting that $\log(b_1)$ is negative (since $F_1 > 0.5$), we must reverse the inequality sign:

$$N + 1 \geq \frac{\log\left(\frac{3(1-\alpha_{target})}{1+3\alpha_{target}}\right)}{\log(b_1)}$$

Since the number of rounds *N* must be an integer, we take the ceiling of the result to find the smallest integer that satisfies the condition. The required number of purification rounds is therefore:

$$N = \left\lceil \frac{\log\left(\frac{3(1-\alpha_{target})}{1+3\alpha_{target}}\right)}{\log\left(\frac{1}{F_1} - 1\right)} - 1 \right\rceil$$

where $\alpha_{target} = \left(\frac{4F_{k,m}^{th} - 1}{3}\right)^{1/m}$.

This completes the re-derivation. ■

REFERENCES

[1] R. R. Schaller, "Moore's law: Past, present and future," *IEEE Spectr.*, vol. 34, no. 6, pp. 52–59, Jun. 1997.

[2] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proc. 8th Annu. ACM Symp. Theory Comput.*, Jun. 1996, pp. 212–219.

[3] N. Abdelgaber and C. Nikolopoulos, "Overview on quantum computing and its applications in artificial intelligence," in *Proc. IEEE 3rd Int. Conf. Artif. Intell. Knowl. Eng. (AIKE)*, Laguna Hills, CA, USA, Dec. 2020, pp. 198–199.

[4] D. Cuomo, M. Caleffi, and A. S. Cacciapuoti, "Towards a distributed quantum computing ecosystem," *IET Quantum Commun.*, vol. 1, no. 1, pp. 3–8, Jul. 2020.

[5] X. Lv, S. Rani, S. Manimurugan, A. Slowik, and Y. Feng, "Quantum-inspired sensitive data measurement and secure transmission in 5G-enabled healthcare systems," *Tsinghua Sci. Technol.*, vol. 30, no. 1, pp. 456–478, Feb. 2025.

[6] X. Guo et al., "Distributed quantum sensing in a continuous-variable entangled network," *Nature Phys.*, vol. 16, no. 3, pp. 281–284, Mar. 2020.

[7] M. Sasaki, "Quantum key distribution and its applications," *IEEE Secur. Privacy*, vol. 16, no. 5, pp. 42–48, Sep. 2018.

[8] M. Krelina, "Quantum technology for military applications," *EPJ Quantum Technol.*, vol. 8, no. 1, p. 24, Dec. 2021.

[9] C. Couteau, "Spontaneous parametric down-conversion," *Contemp. Phys.*, vol. 59, no. 3, pp. 291–304, Jul. 2018.

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

[11] E. Shchukin and P. van Loock, "Optimal entanglement swapping in quantum repeaters," *Phys. Rev. Lett.*, vol. 128, no. 15, Apr. 2022, Art. no. 150503.

[12] X. M. Hu et al., "Progress in quantum teleportation," *Nat. Rev. Phys.*, vol. 5, pp. 339–353, Apr. 2023.

[13] S. Liu, Y. Lou, Y. Chen, and J. Jing, "All-optical entanglement swapping," *Phys. Rev. Lett.*, vol. 128, no. 6, Feb. 2022, Art. no. 060503.

[14] S. Wehner, D. Elkouss, and R. Hanson, "Quantum internet: A vision for the road ahead," *Science*, vol. 362, Oct. 2018, Art. no. eaam9288.

[15] J. L. Liu et al., "Creation of memory-memory entanglement in a metropolitan quantum network," *Nature*, vol. 629, pp. 579–585, Mar. 2024.

[16] H. J. Kimble, "The quantum internet," *Nature*, vol. 453, pp. 1023–1030, Jul. 2008.

[17] Z. Li et al., "Entanglement-assisted quantum networks: Mechanics, enabling technologies, challenges, and research directions," *IEEE Commun. Surv. Tut.*, vol. 25, no. 4, pp. 2133–2189, 4th Quart., 2023.
[18] S. Wei et al., "Towards real-world quantum networks: A review," *Laser Photon. Rev.*, vol. 16, no. 3, 2022, Art. no. 2100219.
[19] Z. Ali, Z. Rezki, and H. Sadjadpour, "Maximization of entanglement sharing in quantum communication networks with fidelity requirements," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Cape Town, South Africa, Dec. 2024, pp. 2809–2815.
[20] J. Rabbie, K. Chakraborty, G. Avis, and S. Wehner, "Designing quantum networks using preexisting infrastructure," *Npj Quantum Inf.*, vol. 8, no. 1, p. 5, Jan. 2022.
[21] S. Shi, X. Zhang, and C. Qian, "Concurrent entanglement routing for quantum networks: Model and designs," *IEEE/ACM Trans. Netw.*, vol. 32, no. 3, pp. 2205–2220, Jun. 2024.
[22] Y. Zhao and C. Qiao, "Redundant entanglement provisioning and selection for throughput maximization in quantum networks," in *Proc. IEEE Conf. Comput. Commun.*, May 2021, pp. 1–10.
[23] F. Hahn, A. Pappa, and J. Eisert, "Quantum network routing and local complementation," *Npj Quantum Inf.*, vol. 5, no. 1, p. 76, Sep. 2019.
[24] M. Pant et al., "Routing entanglement in the quantum internet," *Npj Quantum Inf.*, vol. 5, no. 1, p. 25, Mar. 2019.
[25] Y. Wang, X. Yu, Y. Zhao, and J. Zhang, "Purification-enabled routing with guaranteed fidelity in entanglement distribution networks," in *Proc. Opto-Electron. Commun. Conf. (OECC)*, Shanghai, China, Jul. 2023, pp. 1–4.
[26] C. Li, T. Li, Y.-X. Liu, and P. Cappellaro, "Effective routing design for remote entanglement generation on quantum networks," *Npj Quantum Inf.*, vol. 7, no. 1, p. 21, Jan. 2021.
[27] J. Li et al., "Fidelity-guaranteed entanglement routing in quantum networks," *IEEE Trans. Commun.*, vol. 70, no. 10, pp. 6748–6763, Oct. 2022.
[28] B. He, D. Zhang, S. W. Loke, S. Lin, and L. Lu, "Building a hierarchical architecture and communication model for the quantum internet," *IEEE J. Sel. Areas Commun.*, vol. 42, no. 7, pp. 1919–1935, Jul. 2024.
[29] P. G. Kwiat, K. Mattle, H. Weinfurter, A. Zeilinger, A. V. Sergienko, and Y. Shih, "New high-intensity source of polarization-entangled photon pairs," *Phys. Rev. Lett.*, vol. 75, no. 24, pp. 4337–4341, Dec. 1995.
[30] W. J. Munro, K. Azuma, K. Tamaki, and K. Nemoto, "Inside quantum repeaters," *IEEE J. Sel. Topics Quantum Electron.*, vol. 21, no. 3, pp. 78–90, May 2015.
[31] T. Zhang, H. Li, J. Li, S. Zhang, and H. Shen, "A dynamic combined flow algorithm for the two-commodity max-flow problem over delay-tolerant networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 12, pp. 7879–7893, Dec. 2018.
[32] S.-M. Huang, T.-M. Hsu, J.-J. Du, J.-J. Kuo, and C.-Y. Wang, "Near-optimal swapping and purifying strategy for all-optical-switching entanglement routing," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Cape Town, South Africa, Dec. 2024, pp. 2803–2808.
[33] J. Y. Yen, "Finding the k shortest loopless paths in a network," *Manage. Sci.*, vol. 17, no. 11, pp. 712–716, Jul. 1971.
[34] S. Orlowski, R. Wessäly, M. Pióro, and A. Tomaszewski, "SNDlib 1.0—Survivable network design library," *Networks*, vol. 55, no. 3, pp. 276–286, May 2010.
[35] W. Dür, H.-J. Briegel, J. I. Cirac, and P. Zoller, "Quantum repeaters based on entanglement purification," *Phys. Rev. A, Gen. Phys.*, vol. 59, no. 1, pp. 169–181, Jan. 1999.
[36] C. H. Bennett, G. Brassard, S. Popescu, B. Schumacher, J. A. Smolin, and W. K. Wootters, "Purification of noisy entanglement and faithful teleportation via noisy channels," *Phys. Rev. Lett.*, vol. 76, no. 5, pp. 722–725, Jan. 1996.
[37] M. L. Fredman and R. E. Tarjan, "Fibonacci heaps and their uses in improved network optimization algorithms," in *Proc. 25th Annu. Symp. Found. Comput. Sci.*, 1984, pp. 338–346.