Survivable Entanglement Path Routing Design Model under Node Failures in Quantum Networks

Eji Oki, Masaki Maeda, and Ryuta Shiraki
Graduate School of Informatics, Kyoto University, Kyoto, Japan

***Abstract*—Quantum networks enable end-to-end quantum communication by leveraging inherently probabilistic processes, such as distributing entanglement between distant nodes and performing entanglement swapping at intermediate repeater nodes. Existing work has focused on maximizing the entanglement path success probability under link failures. However, the occurrence of physical node failures poses a significantly greater challenge to maintaining stable entanglement connections. To tackle this problem, it is necessary to develop a path provisioning routing design model that can preserve a high end-to-end entanglement success probability despite the occurrence of node failures. This paper proposes a routing design model to determine node- disjoint entanglement paths in quantum networks, which takes into account both the link-level transmission success probabilities and the node-level entanglement swapping success probabilities at nodes. The proposed model aims to maximize the entanglement path success probability for survivability under node failure. We express the proposed design model in the form of an integer linear programming problem. Numerical results show that the proposed model attains greater survivability compared with the baseline algorithms.**

***Index Terms*—Quantum network, entanglement, path route design, survivability, node failures**

I. Introduction

Quantum communication leverages the uncertainty principle of quantum mechanics to guarantee that information cannot be leaked. At its core lies quantum entanglement, where two or more qubits share strong correlations that persist independently of physical separation. One of the most prominent methods based on entanglement is quantum teleportation [1], [2]. In this process, an entangled pair is first shared between two remote nodes. The sending node holds another qubit whose quantum state needs to be conveyed. By performing a Bell-state measurement on its qubits, the sender obtains a result, which is then communicated to the receiving node over a classical channel. With this information, the receiver applies the appropriate quantum operation, thereby reproducing the sender’s original quantum state.

Distributing entanglement directly across long distances is difficult because of transmission losses [3], [4]. Quantum networks address this limitation through entanglement swapping, as depicted in Fig. 1. At a repeater node, performing a Bell-state measurement on two qubits generates a new entangled pair linking the distant end nodes, effectively extending the entanglement range [5], [6] Both the propagation of entangled qubits through optical channels and the entanglement swapping operations at repeater nodes are inherently probabilistic due to physical constraints in photon transmission and measurement. The overall success probability of such a path is determined by the combined probabilities of successful transmission on each link and successful swapping at the repeater nodes [7], [8]. To manage this uncertainty, heralding protocols are employed to verify whether entanglement distribution has succeeded or failed, enabling entanglement swapping steps to be carried out appropriately.

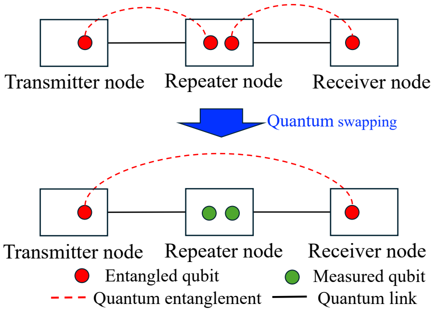


Fig. 1: Process of quantum swapping.

Efficient utilization of quantum network resources has motivated the development of path routing design models [1], [9]–[14]. Comprehensive surveys [1], [9] review entanglement routing techniques, while studies such as [10]–[12] modeled throughput enhancement by incorporating the probabilistic success of entanglement paths and the number of transmitter−receiver pairs. Algorithms presented in [13], [14] further improve end-to-end success probabilities, thereby increasing throughput and reducing quantum memory usage. These studies collectively addressed the probabilistic nature of entanglement routing under the assumption of available physical links.

As quantum networks move toward deployment, they are envisioned for critical infrastructures such as inter-city backbones. Even brief link and node failures can degrade the availability of secure communication, underscoring the need for survivable networks with high transmission success probability.

The work in [15] introduced a survivable routing design model to identify link-disjoint entanglement paths between a source−destination node pair in quantum networks, considering both link-level transmission success

This work was in part supported by JSPS KAKENHI Grant Numbers JP24K17276, 25K22800, and 25KF0055.

probabilities and node-level entanglement swapping success probabilities. The study maximizes the entanglement path success probability with tolerance for up to a given number of link failures (EPSPF). EPSPF quantifies the survivability of an entanglement path for a given source–destination pair subject to a bounded number of link failures. However, no prior study has explored a survivable routing design model to maximize EPSPF against node failures. This raises a natural question: can we develop such a model that tolerates up to a given number of node failures to maximize EPSPF?

This paper proposes a survivable routing design model to find node-disjoint entanglement paths for a source and destination node pair in quantum networks, considering both link-level transmission success probabilities and node-level entanglement swapping success probabilities. The proposed model is to maximize EPSPF, which expresses the survivability of an entanglement path for a given source–destination pair subject to a bounded number of node failures. The proposed routing design model constructs one additional node-disjoint path beyond the number of tolerated node failures; that is, when the network is designed to withstand up to γ node failures, it identifies $\gamma + 1$ node-disjoint paths. This approach ensures that at least one entanglement path with a high success probability remains operational even in the presence of up to γ failing nodes, thereby enhancing the probability of successful entanglement transmission. We express this design as an integer linear programming (ILP) problem. Numerical results indicate that the proposed model outperforms baseline algorithms in terms of EPSPF.

II. PROPOSED PATH ROUTING DESIGN MODEL

We present our proposed path routing design model, along with its problem definition and formulation. We model a quantum network as a directed graph $G = (V, E)$, where V is the set of quantum nodes and E is the set of physical links connecting them. For a source–destination (SD) pair (s, d), let N denote the indices of node-disjoint paths planned between $s \in V$ and $d \in V$. Each link $(u, v) \in E$ has a transmission success probability p_{uv} for an entangled qubit, satisfying $0 < p_{uv} \leq 1$. Additionally, each node $u \in V$ has an entanglement swapping success probability q_u, where $0 < q_u \leq 1$.

The goal of the proposed routing design model is to maximize EPSPF, defined as the minimum end-to-end entanglement success probability across the designed node-disjoint paths, under the assumption that up to γ node failures may occur, where the number of paths is given by $|N| = \gamma + 1$. We introduce the following variables to formulate this optimization problem. Binary variable x^n_{uv} specifies whether link (u, v) is used as part of path n. x^n_{uv} is set to one if link (u, v) is used as part of path n and zero otherwise. Similarly, z^n_u is a binary variable indicating whether node u performs entanglement swapping on path n, determined by the inclusion of neighboring links within that path. z^n_u is set to one if node u performs entanglement swapping on path n and zero otherwise. To eliminate cycles, an auxiliary integer variable h^n_u assigns a topological ordering to the nodes along each path. Finally, c represents the minimum success probability among all designed paths, which corresponds to EPSPF in the optimization problem.

$$
\begin{aligned}
\max \quad & c && \text{(1a)}\\
\text{s.t.} \quad & \sum_{v\in\boldsymbol{V}:(u,v)\in\boldsymbol{E}} x^n_{uv} - \sum_{v\in\boldsymbol{V}:(v,u)\in\boldsymbol{E}} x^n_{vu} = \begin{cases} 1 & \text{if } u = s,\\ -1 & \text{if } u = d,\\ 0 & \text{otherwise,}\end{cases}\\
& \qquad\qquad\qquad\qquad\qquad\qquad \forall n \in \boldsymbol{N} && \text{(1b)}\\
& \sum_{n\in\boldsymbol{N}} x^n_{uv} \leq 1, \quad \forall (u,v) \in \boldsymbol{E} && \text{(1c)}\\
& \sum_{n\in\boldsymbol{N}} \sum_{v\in\boldsymbol{V}:(u,v)\in\boldsymbol{E}} x^n_{uv} \leq 1, \quad \forall u \in \boldsymbol{V} \setminus \{s,d\} && \text{(1d)}\\
& h^n_v - h^n_u \geq 1 - |\boldsymbol{V}|(1 - x^n_{uv}), \quad \forall (u,v) \in \boldsymbol{E}, n \in \boldsymbol{N} && \text{(1e)}\\
& c \leq \prod_{(u,v)\in\boldsymbol{E}:x^n_{uv}=1} p_{uv} \prod_{u\in\boldsymbol{V}:z^n_v=1} q_v, \quad \forall n \in \boldsymbol{N} && \text{(1f)}\\
& z^n_v = \sum_{u\in\boldsymbol{V}:(u,v)\in\boldsymbol{E}} x^n_{uv}, \quad \forall v \in \boldsymbol{V} \setminus \{s,d\}, n \in \boldsymbol{N} && \text{(1g)}\\
& h^n_s = 0, \quad \forall n \in \boldsymbol{N} && \text{(1h)}\\
& x^n_{uv} \in \{0,1\}, \quad \forall (u,v) \in \boldsymbol{E}, n \in \boldsymbol{N} && \text{(1i)}\\
& z^n_u \in \{0,1\}, \quad \forall u \in \boldsymbol{V}, n \in \boldsymbol{N} && \text{(1j)}\\
& c \in \mathbb{R}. && \text{(1k)}
\end{aligned}
$$

Equation (1a) defines the objective, which maximizes the minimum success probability among all entanglement paths. Equation (1b) imposes the flow conservation condition. Equation (1c) enforces link-disjointness, ensuring that no two chosen paths share the same physical link. Equation (1d) enforces node-disjointness, so that paths do not overlap at intermediate nodes. Equation (1e) prevents the formation of cycles. Equation (1f) restricts the variable c to be no greater than the minimum end-to-end success probability over all paths. Here, the right-hand side corresponds to the success probability of path n, expressed as the product of the link-level transmission probabilities and the entanglement swapping probabilities along that path. Equation (1g) activates the swapping variable z^n_v whenever path n includes node v. Finally, (1h)–(1k) specify the conditions for the variables h^n_s, x^n_{uv}, z^n_u, and c, respectively.

To linearize the nonlinear constraint in (1f), we apply a logarithmic transformation to both sides, yielding:

$$
\log c \leq \sum_{(u,v)\in\boldsymbol{E}} x^n_{uv} \log p_{uv} + \sum_{v\in\boldsymbol{V}\setminus\{s,d\}} \sum_{u\in\boldsymbol{V}:(u,v)\in\boldsymbol{E}} x^n_{uv} \log q_v, \quad \forall n \in \boldsymbol{N}. \tag{2}
$$

As maximizing c is equivalent to maximizing $\log c$, we define a new variable $c' = \log c$.

We reformulate the optimization problem in (1a)–(1k), where we replace c in (1a) with $\log c$ and (1f) with (2), and add $c' \in \mathbb{R}$. The reformulated optimization problem forms an ILP problem, which can be solved using standard ILP solvers such as CPLEX [16] or Gurobi [17].

III. BASELINE ALGORITHMS TO FIND NODE-DISJOINT PATHS

To provide reference points for comparison with the proposed routing design model, this section presents two baseline entanglement path routing algorithms: MaxSPG-ND, which greedily selects node-disjoint paths to maximize the immediate success probability, and MaxSPA-ND, which optimizes the overall success probability considering all candidate node-disjoint paths.

A. MaxSPG-ND algorithm

The MaxSPG-ND algorithm is a heuristic designed to generate multiple node-disjoint paths between a source and a destination using a sequential greedy strategy. It constructs the set of node-disjoint paths, N, by repeatedly applying Dijkstra's algorithm and removing links associated with nodes already included in previously discovered paths [18]–[21].

To adapt Dijkstra's algorithm for quantum networks, we modify it to account for the probabilistic nature of entanglement distribution and swapping. The end-to-end success probability of a path is calculated as the product of link success probabilities and the success probabilities of entanglement swapping along the path. Applying the logarithmic transformation as in (2), the problem reduces to minimizing the sum of link and node costs. Each link (u, v) is assigned a weight $-\log p_{uv}$, while each node u is assigned $-\log q_u$. Since Dijkstra's algorithm only handles link weights, half of each node' s cost is distributed to its incident links, giving the effective link weight:

$$-\log p_{uv} - \frac{1}{2}(\log q_u + \log q_v). \tag{3}$$

Regarding computational complexity, each iteration of Dijkstra's algorithm has a cost of $O((|E| + |V|)\log|V|)$, leading to a total complexity of $O(|N|(|E| + |V|)\log|V|)$ for MaxSPG-ND. The path search dominates the computation, while eliminating links routed on the previously found paths contributes negligibly in terms of asymptotic complexity.

MaxSPG-ND follows a greedy selection procedure, choosing paths one after another without coordinating globally among the $|N|$ node-disjoint paths. After selecting a path, all links connected to its nodes are removed from the graph, and the next shortest path is found independently in the remaining graph. Consequently, the last path constructed, the $|N|$th path, generally has the lowest end-to-end success probability among all paths. Denoting the success probability of the ith path as g_i, the greedy selection ensures: $g_1 \geq g_2 \geq \cdots \geq g_{|N|}$. While MaxSPG-ND is straightforward to implement, the greedy strategy can lead to a reduction in EPSPF.

B. MaxSPA-ND algorithm

The MaxSPA-ND algorithm serves as a path routing design approach that identifies $|N|$ node-disjoint paths to maximize the overall success probability. For the special case of two node-disjoint paths ($|N| = 2$), MaxSPA-ND can be efficiently implemented using Suurballe's algorithm [18], [22], which is an algorithm to find link-disjoint paths while minimizing the total link cost.

We explain how MaxSPA-ND finds two node-disjoint paths. Suurballe's algorithm is a classical method for computing a pair of link-disjoint paths that minimizes their total link cost. It begins by applying Dijkstra's algorithm to find the first minimum-cost path on the original graph. Using the found minimum-cost path, we transform the original graph into a modified graph for finding the overall success probability [18]. The graph transformation involves splitting each node on the found path into two nodes and reassigning the original links to the split nodes, while adjusting the link directions. Next, for the modified graph, it reweights all link costs and reverses the directions of the links on the first path according to:

$$w'(u, v) = w(u, v) + d(u) - d(v), \tag{4}$$

where $w(u, v)$ is the link cost and $d(u)$ is the minimum accumulated cost from the source to node u. Dijkstra's algorithm is then executed on the modified graph to find the second path. The resulting paths are combined by removing any overlapping links, yielding a pair of node-disjoint paths.

When more than two paths are required ($|N| \geq 3$), the above procedure for finding two node-disjoint paths while minimizing the total link cost can be extended to generate $|N|$ node-disjoint paths in polynomial time [18]. In this setting, the objective is to minimize the total sum of end-to-end path costs over all $|N|$ paths. As with MaxSPG-ND, the logarithmic cost transformation is applied to adapt the algorithm to quantum networks.

Regarding computational complexity, each iteration of Dijkstra's algorithm requires $O((|E| + |V|)\log|V|)$ time [23], and since it is executed $|N|$ times, MaxSPA-ND has an overall complexity of $O(|N|(|E| + |V|)\log|V|)$. The path search dominates the computation, while splitting each node routed on the previously found paths into two nodes and reassigning the original links to the split nodes while adjusting the link directions, contributes negligibly in terms of asymptotic complexity.

MaxSPA-ND produces a set of node-disjoint paths that minimizes the total cost, which is equivalent to maximizing the product of end-to-end success probabilities across all paths. However, minimizing the total cost does not guarantee balanced success probabilities among the paths. Since our objective is to maximize the minimum success probability across the $|N|$ paths, MaxSPA-ND, despite improving the overall success probability, may yield suboptimal results in terms of EPSPF when the success probabilities of selected paths are uneven.

IV. Numerical results

This section assesses the EPSPF values, namely the entanglement path success probabilities that remain achievable under a given number of node failures, as produced by the proposed model in comparison with the baseline algorithms MaxSPG-ND and MaxSPA-ND. We also report their computation times. We solve the optimization problem defined in (1a)–(1k) with CPLEX [16] after transforming it into an ILP problem as described in II. The experiments are conducted on a machine equipped with an Intel Xeon Gold 6314U running at 2.3 GHz with 32 cores and 128 GB of RAM. We develop the implementation in Python.

The assumptions and notation for our evaluation are as follows. We allow at most one node failure in the network, so each source–destination (SD) pair is provisioned with $|N| = 2$ node-disjoint paths. We generate random network topologies using the method in [24], with $|V|$ nodes and average degree D. Links are placed uniformly at random, ensuring that every node has at least two neighbors. We do not execute SD pairs that cannot support $|N|$ disjoint paths. For each link $(u, v) \in E$, the transmission success probability p_{uv} is drawn uniformly from $[0.7, 1.0]$, and each node $u \in V$ is assigned an entanglement swapping success probability q_u of 0.9. $T_{|V|,D}$ denotes the set of topologies generated with parameters $|V|$ and D, and for each $t \in T_{|V|,D}$, C_t is the set of valid SD pairs.

To highlight the distinctions between the proposed model and the baseline algorithms, we measure the deviation of the EPSPF obtained by each baseline from that of the proposed model. We introduce the deviation metric $\delta(|V|, D)$ as:

$$\delta(|V|, D) = \frac{1}{|T_{|V|D}||C_t|} \sum_{t \in T_{|V|D}} \sum_{c \in C_t} \frac{p_{ct}^{\text{pro}} - p_{ct}^{\star}}{p_{ct}^{\text{pro}}}, \tag{5}$$

where p_{ct}^{pro} and $p_{ct}^{\star}$ represent the EPSPF values obtained by the proposed model and a given baseline algorithm for SD pair $c \in C_t$ in topology $t \in T_{|V|D}$, respectively.

We repeat each evaluation multiple times to estimate both EPSPF and computation time. For EPSPF, we ensure that the 95% confidence interval for each trial does not exceed 5% of its sample mean. The results reported are the average EPSPF values and computation times across all trials.

Figure 2 shows the EPSPFs for all path routing design approaches depending on the average node degree D with $|V| = 15$. Across all approaches, the EPSPFs increase as D grows from 3.5 to 5. This is because, as illustrated in Fig. 3, a higher average node degree reduces the number of hops in the worst paths, leading to fewer link success probabilities and entanglement swapping operations being multiplied along the paths, which in turn raises the EPSPFs.

Figure 4 shows the deviation of the EPSPFs of MaxSPG-ND and MaxSPA-ND from the proposed model, depending on the average node degree, D, with the number of nodes, $|V| = 15$ and 30. The deviation of the EPSPFs of MaxSPA-ND is smaller than that of MaxSPG-ND over all average node degrees, D. This means that MaxSPA-ND has a value of the EPSPF closer to the proposed model and a higher value than MaxSPG-ND. For both MaxSPG-ND and MaxSPA-ND, the deviations with $|V| = 30$ are larger than those of $|V| = 15$ over all values of D. When $|V|$ increases, the effectiveness of the proposed model enhances; the proposed model gets more flexibility.

Figure 5 shows the computation time of the proposed model, MaxSPG-ND, and MaxSPA-ND depending on the number of nodes, $|V|$, with $D = 10$. The computation time of each approach increases as $|V|$ gets large. The proposed model takes at most 217 times longer than MaxSPG-ND and MaxSPA-ND in our examined cases. The maximum computation time of the proposed model is 5.95 seconds with $|V| = 100$. The proposed model is still practical.

V. Conclusions

This paper proposed a routing design model to determine node- disjoint entanglement paths in quantum networks, where it incorporates both the link-level transmission success probabilities and the node-level entanglement swapping success probabilities at nodes. The objective of the proposed model is to maximize the entanglement path success probability for survivability under node failure. We formulated the proposed design model as an ILP problem. Numerical results observed that the proposed model obtains higher survivability compared with the baseline algorithms. Whereas the proposed model requires up to 217 times more computation time than the baselines, it remains practical.

References

[1] A. Abane, M. Cubeddu, V. S. Mai, and A. Battou, “Entanglement routing in quantum networks: A comprehensive survey,” *IEEE Trans. Quantum Eng.*, vol. 6, pp. 1–39, 2025.

[2] R. V. Meter, *Quantum Networking.* Hoboken, NJ, USA: Wiley, 2014.

[3] S. Muralidharan, L. Li, J. Kim, N. LÃ¼tkenhaus, M. D. Lukin, and L. Jiang, “Optimal architectures for long distance quantum communication,” *Sci. Rep.*, vol. 6, no. 1, p. 20463, 2016.

[4] W. J. Munro, K. Azuma, K. Tamaki, and K. Nemoto, “Inside quantum repeaters,” *IEEE J. Sel. Top. Quantum Electron.*, vol. 21, no. 3, pp. 78–90, 2015.

[5] M. Żukowski, A. Zeilinger, M. A. Horne, and A. K. Ekert, ““event-ready-detectors” bell experiment via entanglement swapping,” *Phys. Rev. Lett.*, vol. 71, pp. 4287–4290, Dec 1993.

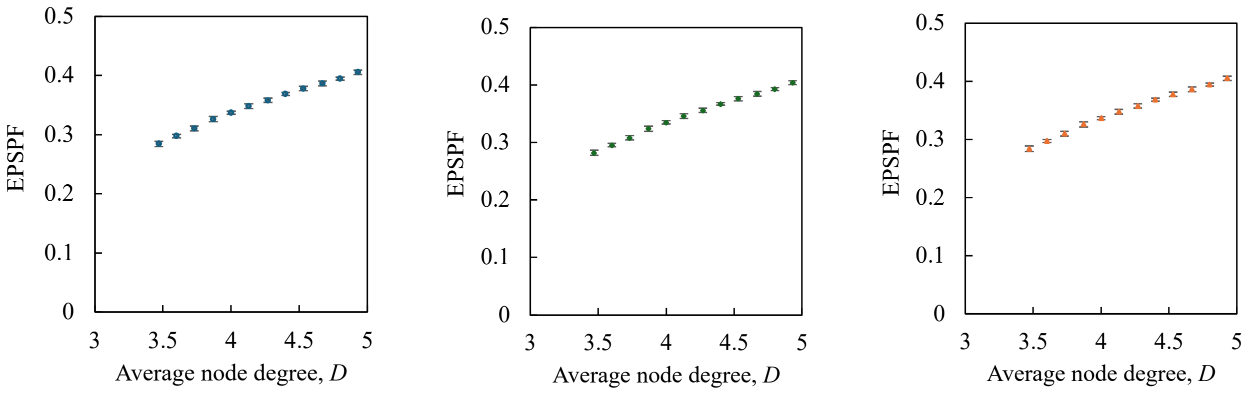
[6] A. Chang and G. Xue, “Order matters: On the impact of swapping order on an entanglement path in a quantum network,” in *IEEE Int. Conf. Comput. Commun. Workshops*, 2022, pp. 1–6.

[7] B. Kar and P. Kumar, “Routing protocols for quantum networks: Overview and challenges,” *arXiv preprint arXiv:2305.00708*, 2023.

[8] T. N. Nguyen, K. J. Ambarani, L. Le, I. Djordjevic, and Z.-L. Zhang, “A multiple-entanglement routing framework for quantum networks,” arXiv preprint arXiv:2207.11817, 2022.

[9] F. Dupuy, C. Goursaud, and F. Guillemin, “A survey of quantum entanglement routing protocols challenges for wideâ area networks,” *Adv. Quantum Technol.*, vol. 6, no. 5, p. 2200180, 2023.

[10] 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, 2024.



(a) Proposed model (b) MaxSPG-ND (c) MaxSPA-ND

Fig. 2: EPSPFs for all path routing design approaches with $|N| = 2$. Each error bar represents the 95% confidence intervals.

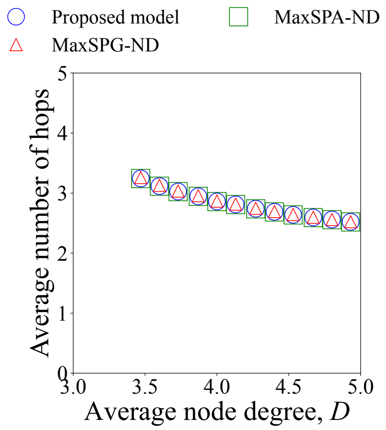


Fig. 3: Average number of hops on average node degree with $|N| = 2$ and $|V| = 15$.

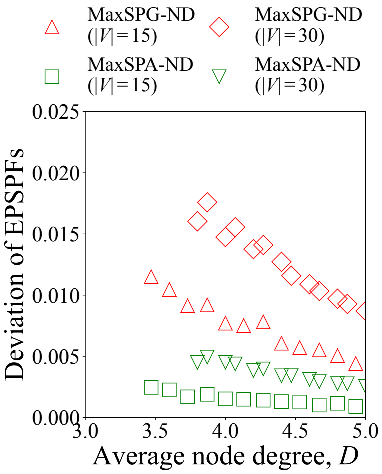


Fig. 4: Deviation of EPSPFs depending on average node degrees with $|N| = 2$ and $|V| = 15$ and 30.

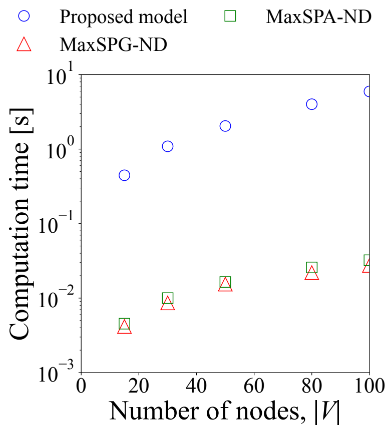


Fig. 5: Computation time dependent on number of nodes with $|N| = 2$ and $D = 10$.

[11] Y. Zhao and C. Qiao, "Redundant entanglement provisioning and selection for throughput maximization in quantum networks," in *IEEE Int. Conf. Comput. Commun.*, 2021, pp. 1–10.

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

[13] G. Zhao, J. Wang, Y. Zhao, H. Xu, L. Huang, and C. Qiao, "Segmented entanglement establishment with all-optical switching in quantum networks," *IEEE/ACM Trans. Netw.*, vol. 32, no. 1, pp. 268–282, 2024.

[14] S. Zhang, S. Shi, C. Qian, and K. L. Yeung, "Fragmentation-aware entanglement routing for quantum networks," *J. Lightw. Technol.*, vol. 39, no. 14, pp. 4584–4591, 2021.

[15] M. Maeda, R. Shiraki, and E. Oki, "Modeling and analysis of entanglement path provisioning for survivability under link failure," in *Proc. Asia-Pacific Netw. Oper. Manage Symp.*, 2025, pp. 1–4.

[16] "CPLEX optimization studio," available at: https://www.ibm.com/products/ilog-cplex-optimization-studio (accessed on 2025-10-12).

[17] "Gurobi optimization," available at: https://www.gurobi.com (accessed on 2025-10-12).

[18] R. Bhandari, *Survivable Networks: Algorithm for Diverse Routing*. Springer, 1999.

[19] E. Oki, N. Matsuura, K. Shiomoto, and N. Yamanaka, "A disjoint path selection scheme with shared risk link groups in gmpls networks," *IEEE Commun. Lett.*, vol. 6, no. 9, pp. 406–408, 2002.

[20] E. Oki and N. Yamanaka, "A recursive matrix-calculation method for disjoint path search with hop link number constraints," *IEICE Trans. Commun.*, vol. 78, no. 5, pp. 769–774, May 1995.

[21] E. Oki, N. Yamanaka, and F. Pitcho, "Multiple-availability-level atm network architecture," *IEEE Communi. Mag.*, vol. 33, no. 9, pp. 80–88, 1995.

[22] J. W. Suurballe and R. E. Tarjan, "A quick method for finding shortest pairs of disjoint paths," *Networks*, vol. 14, no. 2, p. 325â 336, Jun. 1984.

[23] J. Kleinberg and Éva Tardos, *Algorithm Design*, 2nd ed. Pearson Education, Limited, 2022.

[24] D. Dunn, W. Grover, and M. MacGregor, "Comparison of *k*-shortest paths and maximum flow routing for network facility restoration," *IEEE J. Sel.. Areas Commun.*, vol. 12, no. 1, pp. 88–99, 1994.