Poster: An FPTAS for Shortest-Longest Path Problem

Jianwei Zhang*, Xin Li, Bowen Cui
School of Information Science and Engineering
Yunnan University
Kunming, 650500, China
*janyway@outlook.com, {lixin5, cuibowen}@stu.ynu.edu.cn

Chunling Yang
Information Technology Center
Zhejiang University
Hangzhou, 310058, China
chly@zju.edu.cn

***Abstract*—Motivated by multi-domain Service Function Chain (SFC) orchestration, we define the Shortest-Longest Path (SLP) problem, prove its hardness, and design an efficient Fully Polynomial Time Approximation Scheme (FPTAS) using the scaling and rounding technique to compute an approximation solution with provable performance guarantee.**

***Index Terms*—QoS routing, approximation algorithm, service function chain (SFC), virtual network function (VNF)**

I. Introduction

Suppose that we want to deploy an SFC across multiple domains, the total computing resources offered by each domain and accumulated along the path must be sufficient to host all the VNFs in the SFC [2]. Furthermore, the total cost (or delay, IGP weight) along the path is expected to be minimized. With the assistance of *Topology Aggregation* (TA) [3], networking and computing resource information in a domain can be abstracted into virtual edges among border routers [2]. Then, a natural question is how to find a path simultaneously satisfies the cost constraint and the resource constraint over the abstracted virtual network.

Existing studies [4]–[6] related to multi-constrained QoS routing problems mainly focus on the case that all the metrics belong to the *upper-bound type*, i.e., the metric sum along a path cannot be larger than some upper bound. However, the aforementioned resource metric belongs to the *lower-bound type*, i.e., the metric sum along a path cannot be smaller than some lower bound. Although both of the two types are *additive* metrics, they cannot be solved using existing approaches. This is the theoretical motivation of this paper.

In this paper, we define the *Shortest-Longest Path* (SLP) problem, prove its hardness, and design an efficient *Fully Polynomial Time Approximation Scheme* (FPTAS) using the *scaling and rounding* technique to compute an approximation solution with provable performance guarantee.

II. Model

The network is modeled as a directed graph $G = (V, E, w_S, w_L)$, where V represents the vertex set and E the edge set. The number of vertices and edges are denoted by n and m, respectively. Each edge e is associated with a performance metric pair $(w_S(e), w_L(e))$. The S-metric can be cost, delay, IGP weight, and etc. The L-metric can be computing (or other types) resource. The request $r = (s, t, W_S, W_L)$ is a unicast from s to t whose routing path p satisfies $w_S(p) \leq W_S$ and $w_L(p) \geq W_L$.

Definition 1. *Feasible path*: The path p from s to t in G that satisfies $w_S(p) \leq W_S$ and $w_L(p) \geq W_L$ is *feasible*.

Definition 2. *Approximately feasible path*: The path p from s to t in G that satisfies $w_S(p) \leq (1+\epsilon) W_S$ and $w_L(p) \geq (1-\epsilon) W_L$ is *approximately feasible* (Fig. 1), where $\epsilon \in (0, 1]$.

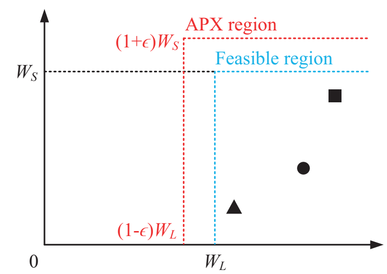


Fig. 1: Feasible region and approximation region.

Definition 3. *Shortest-Longest Path (SLP) problem*: Given a graph $G = (V, E, w_S, w_L)$ and a request $r = (s, t, W_S, W_L)$, the SLP problem is to compute a feasible path p for r in G.

Theorem 1. The SLP problem is NP-complete.

Proof: Please see the technical report [1]. □

Fig. 2 is a motivation example. The left part is a multi-domain network after TA. The right part is the TA details for domain B; other domains are omitted for clarity. It is easy to verify that the only feasible solution for request $(s, t, 9, 9)$ is the path $s - a - b - c - d - t$. Note that our problem is based on the network after TA. However, the TA strategy permits great design flexibility and is beyond the scope of this paper.

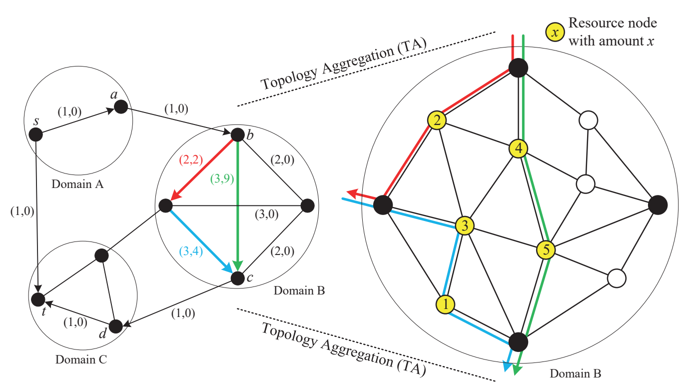


Fig. 2: Motivation example. Only TA for domain B is shown.

979-8-3503-5171-2/24/$31.00 ©2024 IEEE

Algorithm 1 FPTAS-SLP

Input: G; r; ϵ.

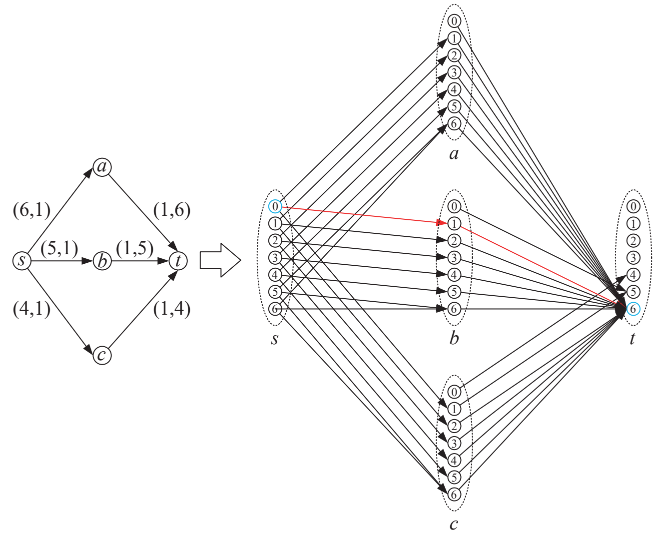
Output: An approximately feasible path p^τ for r in G with approximation ratio $(1+\epsilon, 1-\epsilon)$, if there exists one.

1: Set the new weights $w_k^\tau(e) = \left\lceil \frac{w_k(e)}{W_k} \cdot \tau \right\rceil, k = S, L$ and new bounds $W_k^\tau = \lceil \tau \rceil, k = S, L$, where $\tau := \frac{m}{\epsilon}$.

2: Construct an auxiliary graph $G^\tau = \{V^\tau, E^\tau\}$. The vertex set is $V^\tau = V \times \{0, 1, ..., \lceil \tau \rceil\}$. The edge set E^τ contains directed edges from vertex (u, x) to (v, y) such that $y = x + \min\{w_L^\tau(u, v), \lceil \tau \rceil - x\}$. The weight of all such edges is $w_S^\tau(u, v)$.

3: Calculate the shortest path p^τ in G^τ from $(s, 0)$ to $(t, \lceil \tau \rceil)$, while ERD is performed in G.

4: If $w_S^\tau(p^\tau) \le W_S^\tau$, output p^τ; Else, output no solution.



(a) Original graph (b) Auxiliary graph

Fig. 3: Illustrative example. $r = (s, t, 6, 6)$. $\epsilon = 1$.

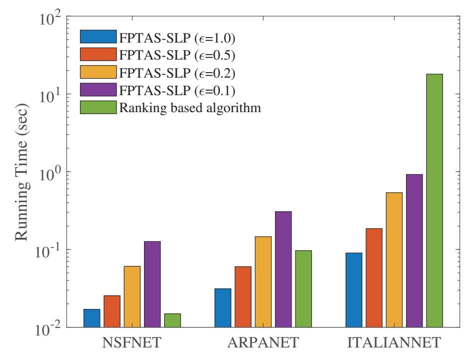


Fig. 4: Comparison of running time. y-axis is logarithmic.

III. Algorithm

FPTAS-SLP includes four steps [4]–[6]. Step 1 is to scale and round the edge weights in the original graph. Step 2 constructs the auxiliary graph. Step 3 computes a shortest path w.r.t. the S-metric while executing *Edge Repetition Detection* (ERD) on the original graph. Step 4 checks the feasibility.

Fig. 3 illustrates how to construct an auxiliary graph. The auxiliary graph construction is a dynamic programming process. The key differences between the auxiliary graph and the one in [4] are two-fold. First, the edges between adjacent sub-vertices in a vertex are removed. For instance, there are no edges between $(s, 0)$ and $(s, 1)$. Second, some edges are newly added. For instance, there is an edge between $(s, 6)$ and $(c, 6)$. We note that the auxiliary graph *may not be* a directed acyclic graph. It is worthwhile noticing an important feature of the auxiliary graph that any path reaching the last vertex is feasible w.r.t. the scaled and rounded L-metric bound, but not necessarily the S-metric bound. For example, both of the paths $s \to a \to t$ and $s \to b \to t$ can reach the last vertex $(t, 6)$ and therefore satisfy the L-metric constraint, while the former one violates the S-metric constraint and the latter one is the only feasible path in this example.

Theorem 2. The proposed FPTAS-SLP is guaranteed to compute a $(1+\epsilon, 1-\epsilon)$-approximation solution for the SLP problem, where $\epsilon \in (0, 1]$, suppose that it is feasible. The time complexity is $O\left(\tau n \log \tau n + \tau m^2\right)$, where $\tau := \frac{m}{\epsilon}$.

Proof: Please see the technical report [1]. □

IV. Simulation

The simulation was run on a PC with Intel G5400 at 3.70GHz and 8GB RAM. The testing Internet topologies include NSFNET (14 nodes and 21 edges), ARPANET (20 nodes and 32 edges), and ITALIANNET (33 nodes and 67 edges) [5], [6]. Fig. 4 compares the average running time to find a feasible path for a given request. The running time of FPTAS-SLP increases linearly as the network size grows up, and also increases linearly with the growth of ϵ for the same network. Compared to the proposed FPTAS-SLP, the ranking based algorithm in [2] does not scale well.

V. Conclusion

In this paper, we for the first time define the SLP problem and design an approximation algorithm (FPTAS-SLP) with provable performance guarantee. Simulation result demonstrates its computing efficiency advantage. In the future, we will first design BGP supported TA strategies, and then apply FPTAS-SLP into multi-domain SFC orchestration scenarios.

References

[1] J. Zhang, "An FPTAS for shortest-longest path problem," *arXiv preprint*, arXiv:2404.13488, 2024.

[2] G. Sun, Y. Li, D. Liao, *et al.*, "Service function chain orchestration across multiple domains: A full mesh aggregation approach," *IEEE Transactions on Network and Service Management*, vol. 15, no. 3, pp. 1175–1191, 2018.

[3] B. Yan, Y. Zhao, X. Yu, *et al.*, "Service function path provisioning with topology aggregation in multi-domain optical networks," *IEEE/ACM Transactions on Networking*, vol. 28, no. 6, pp. 2755–2767, 2020.

[4] J. Huang and Y. Tanaka, "QoS routing algorithms using fully polynomial time approximation scheme," in *Proc. IEEE IWQoS*, 2011, pp. 1–3.

[5] G. Xue, A. Sen, W. Zhang, *et al.*, "Finding a path subject to many additive QoS constraints," *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, pp. 201–211, 2007.

[6] G. Xue, W. Zhang, J. Tang, *et al.*, "Polynomial time approximation algorithms for multi-constrained QoS routing," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 656–669, 2008.