Poster: <SmartTE: Partially Deployed Segment Routing for Smart Traffic Engineering with Deep Reinforcement Learning>

Zeyu Luan*§, Qing Li†§, Yong Jiang‡§

*Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, Shenzhen, China
†Southern University of Science and Technology, Shenzhen, China
‡Shenzhen International Graduate School, Tsinghua University, Shenzhen, China
§PCL Research Center of Networks and Communications, Peng Cheng Laboratory, Shenzhen, China
luanzy18@mails.tsinghua.edu.cn, liq8@sustech.edu.cn, jiangy@sz.tsinghua.edu.cn

***Abstract*—Segment Routing (SR) provides Traffic Engineering (TE) with the ability of explicit path control by steering traffic passing through specific SR routers along a desired path. However, large-scale migration from a legacy IP network to a full SR-enabled one requires prohibitive hardware replacement and software update. Therefore, network operators prefer to upgrade a subset of IP routers into SR routers during a transitional period. This paper proposes SmartTE to optimize TE performance in hybrid IP/SR networks where partially deployed SR routers coexist with legacy IP routers. We use two centrality criteria in graph theory to decide which IP routers should be upgraded into SR routers under a given upgrading ratio. SmartTE leverages Deep Reinforcement Learning (DRL) to infer the optimal traffic splitting ratio across multiple pre-defined paths between source-destination pairs. Extensive experimental results with real-world topologies show that SmartTE outperforms other baseline TE solutions in minimizing the maximum link utilization and achieves comparable performance as a full SR network by upgrading only 30% IP routers.**

***Index Terms*—Traffic Engineering, Segment Routing, Deep Reinforcement Learning**

I. Introduction

Segment Routing (SR) [1] enables fine-grained explicit path control by specifying a list of SR routers through which traffic has to pass. Traffic Engineering (TE) can distribute traffic across multiple pre-defined paths between source-destination pairs, with an aim to minimize maximum link utilization.

However, large-scale migration from a legacy IP network to a full SR-enabled one requires prohibitive hardware replacement and software update. ISPs may prefer to upgrade a subset of IP routers into SR routers, and then gradually expand the scale of SR router deployment to the whole network. Therefore, partially deployed SR routers will coexist with IP routers during the transitional period. How to implement TE in such hybrid IP/SR networks remains a critical problem.

Node-constrained TE is proven NP-hard [2], in which traffic is constrained to pass through specific SR routers. Therefore, we explore the potential of Deep Reinforcement Learning (DRL) [3] to tackle this NP-hard problem. DRL learns an optimal control policy from experience by interacting with the environment in a trial-and-error manner. SmartTE leverages DRL to infer the optimal traffic splitting ratio across multiple pre-defined paths between source-destination pairs.

Our contributions are as follows:

- We use two centrality criteria from graph theory to decide which IP routers should be upgraded into SR routers under a given upgrading ratio.
- We leverage DRL to infer the optimal traffic splitting ratio across multiple available paths between source-destination pairs.
- Experiments with real-world topologies show that SmartTE outperforms other baselines in minimizing maximum link utilization by upgrading only 30% IP routers.

II. System Overview

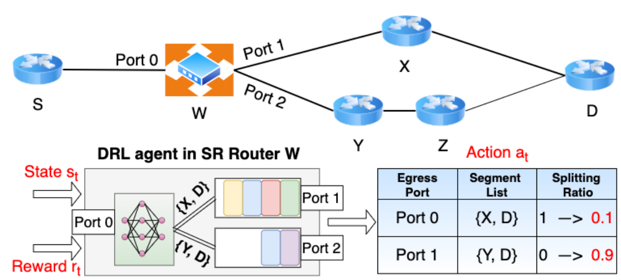


Fig. 1: System overview of SmartTE

Fig.1 shows all IP routers running the OSPF protocol forwards traffic along the shortest path via X. We select to upgrade W into an SR router. In such hybrid IP/SR networks, two paths are available for traffic from the source S towards the destination D. Once detecting congestions on the shortest path, the DRL agent deployed in W will offload part of the traffic to the non-shortest path by adjusting the traffic splitting ratio between Port 1 and Port 2. The SR router W infers an optimal traffic splitting ratio (a_t) based on dynamic link load

Annex to ISBN 978-3-903176-39-3© 2021 IFIP

condition (s_t) and the feedback reward of previous inference (r_t).

III. SR Router Deployment

The first step in SmartTE is to select a subset of legacy IP routers and then upgrade them into SR-enabled ones. We draw upon two centrality concepts [4] from graph theory, namely Group Betweenness Centrality (GBC) and Group Degree Centrality (GDC), to rank IP routers in terms of their contributions to explicit path control.

A. Group Between Centrality

For a network topology $G = (V, E)$, Group Between Centrality (GBC) for a subset of nodes $W \subseteq V$ is defined as the fraction of all-pairs shortest paths passing through any node in W:

$$GBC(W) = \sum_{s,t \in V-W; s \neq t} \frac{\sigma(s,t \mid W)}{\sigma(s,t)} \tag{1}$$

where $\sigma(s,t)$ is the number of shortest paths between the node pair (s,t) and $\sigma(s,t \mid W)$ is the number of these paths passing through any node in W at the same time.

According to the OSPF protocol, IP routers forward traffic on the shortest path by default. Therefore, selecting to upgrade a group of routers with larger GBC can maximize the chances of path control for traffic on the shortest path.

B. Group Degree Centrality

Similarly, Group Degree Centrality (GDC) for a subset of nodes $W \subseteq V$ is defined as the fraction of links connected to any node in W:

$$GDC(W) = \sum_{w \in W} \frac{deg(w)}{|E|} \tag{2}$$

where $deg(w)$ is the degree of the node w and $|E|$ is the total number of links in the network topology $G = (V, E)$.

The essence of multi-path routing is to fully exploit available network resources on non-shortest paths, so as to alleviate possible congestions on the shortest path. Therefore, a group of routers with larger GDC have a higher degree of routing flexibility to offload traffic to other non-shortest paths.

IV. SmartTE Algorithm

SmartTE leverages Deep Deterministic Policy Gradient (DDPG) [5] to compute optimal traffic splitting ratio across multiple available paths between source-destination pairs. This section defines the state space, action space, and reward function in DDPG, and presents the detailed DDPG algorithm.

Algorithm 1: DDPG Algorithm for SmartTE

1 Initialize actor network $\mu(s \mid \theta^{\mu})$ and critic network $Q(s, a \mid \theta^{Q})$ with parameters θ^{μ} and θ^{Q};
2 Initialize target networks $\mu'(s \mid \theta^{\mu'})$ and $Q'(s, a \mid \theta^{Q'})$ with parameters $\theta^{\mu'} \leftarrow \theta^{\mu}$ and $\theta^{Q'} \leftarrow \theta^{Q}$;
3 Initialize Replay Buffer $\mathcal{R}$ to store state transitions;
4 Initialize Ornstein-Uhlenbeck random process $\mathcal{N}$ to explore action space;
5 **while** *True* **do**
6   Collect link load condition from OSPF LSA as input state s_t;
7   Apply actor network with random process to generate action $a_t = \mu(s_t \mid \theta^{\mu}) + \mathcal{N}_t$;
8   Execute traffic splitting ratio a_t across available paths;
9   Calculate reward r_t and observe next-state s_{t+1};
10   Store 4-tuple state transitions (s_t, a_t, r_t, s_{t+1}) into Replay Buffer $\mathcal{R}$;
11   Sample a mini-batch of K samples from $\mathcal{R}$;
12   **for** *i = 1 ... K* **do**
13   Apply target networks to compute target value: $y_i = r_i + \gamma \cdot Q'(s_{i+1}, \mu'(s_{i+1} \mid \theta^{\mu'}) \mid \theta^{Q'})$;
14   **end**
15   Update critic network by minimizing the loss: $L = \frac{1}{N} \sum_{i=1}^{K} (y_i - Q(s_i, a_i \mid \theta^{Q}))^2$;
16   Update actor network by policy gradient theorem: $\nabla_{\theta^{\mu}} J \approx \frac{1}{N} \sum_{i=1}^{K} \nabla_a Q(s_i, a_i \mid \theta^{Q}) \nabla_{\theta^{\mu}} \mu(s_i \mid \theta^{\mu})$;
17   Update target networks with soft synchronization:
        $\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}$
        $\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'}$
18 **end**

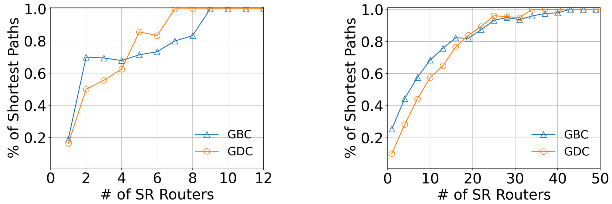
A. State, Action and Reward

At step t, the state space, action space, and reward function are defined as follows:

- **State** s_t is a vector of link load condition across all links that can be derived from OSPF LSAs.
- **Action** a_t is a vector of traffic splitting ratio across multiple available paths between source-destination pairs.
- **Reward** r_t is a negative value of the maximum link utilization across all links in the network.

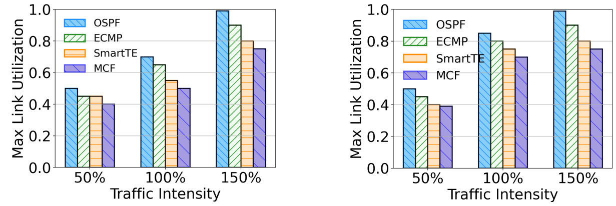
B. DDPG Algorithm

Algorithm 1 shows the details of DDPG algorithm in SmartTE. Line 1-2 are parameter initialization for actor-critic networks and target networks. Line 3 initializes a replay buffer $\mathcal{R}$ to store state transitions. Line 4 initializes a random process called Ornstein-Uhlenbeck (O-U) for exploration in the action space, which is a common practice in continuous control task. Line 6 defines the input state s_t as link load condition across all links by collecting the information from periodic OSPF LSA broadcasting. Line 7 obtains the action a_t by adding the O-U random process $\mathcal{N}_t$ to the output of the actor



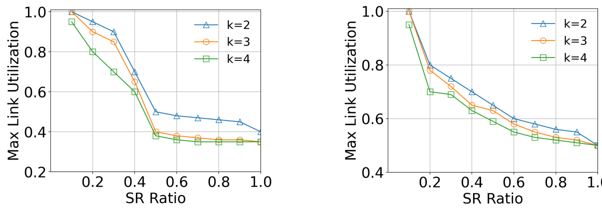
(a) Abilene (b) Germany

Fig. 2: All-pairs shortest paths containing SR Routers



(a) Abilene (b) Germany

Fig. 4: TE performance with increasing traffic intensity



(a) Abilene (b) Germany

Fig. 3: Maximum link utilization with SR Ratio

network $\mu(s_t \mid \theta^{\mu})$. Line 8 executes the traffic splitting ratio a_t across available paths between source-destination pairs. Line 9 calculates the reward r_t according to the maximum link utilization and observes the next state s_{t+1}. Line 10 stores a 4-tuple state transition in the replay buffer $\mathcal{R}$, including (s_t, a_t, r_t, s_{t+1}). Line 11 samples a mini-batch of K state transitions from the replay buffer. Line 12-14 computes the target value y_i for each sample. Line 15 updates the parameters of the critic network by minimizing the loss function L, which is the square error between y_i and the Q-value $Q(s_i, a_i \mid \theta^Q)$. Line 16 updates the parameters of the actor network by policy gradient theorem.

V. Performance Evaluation

We evaluate the performance of SmartTE with real-world network topologies from SNDlib [6], including Abilene (12 nodes 15 links) and Germany (50 nodes 88 links). We use PyTorch framework to train the DRL agent of SmartTE. The actor-critic network in DDPG consists of two hidden layers with 20 and 10 neurons, respectively. The activation function is ReLU for the two hidden layers and Softmax for the output layer. The learning rate is 0.001 for Adam optimizer in both actor and critic network. The soft update coefficient is 0.001 for parameter synchronization of target networks. We set the discounted factor to 0.99 to calculate the target Q-value.

Fig.2 demonstrates the percentage of all-pairs shortest paths containing at least one SR router with an increasing deployment of SR routers. The result shows that over 80% of shortest paths will contain at least one SR router after upgrading 50% IP routers. It means traffic between a majority of node pairs will encounter at least one SR router during their transmission along the shortest path, and have a chance to be re-routed to other non-shortest paths.

Fig.3 shows that the maximum link utilization decreases with the increasing ratio of SR router deployment. When the ratio equals 1, the hybrid IP/SR network will become a full SR one. A growing number of SR routers enable a higher degree of routing flexibility for traffic demands. However, the improvement becomes slight after exceeding a certain threshold, indicating a hybrid IP/SR network is sufficient for comparable performance as a full SR-enabled one.

Fig.4 compares the performance of SmartTE with other baseline TE solutions, including OSPF, ECMP, and Multi-Commodity Flow (MCF). MCF can be regarded as a theoretical optimum in which every router can fractionally split traffic with an arbitrary splitting ratio. In SmartTE, only SR routers can split traffic across multiple available paths but achieve near-optimal performance as MCF under all traffic intensities.

VI. Conclusion

We proposed SmartTE to optimize TE performance by exploring the potential of Deep Reinforcement Learning. We determined the upgrading priority of IP routers by two criteria, namely Group Betweenness Centrality and Group Degree Centrality. Experimental results with real-world network topologies showed that less than 30% partially deployed SR routers can achieve comparable TE performance as a full SR network.

Acknowledgment

This work is supported by the National Key Research and Development Program of China under Grant No. 2018YFB1804704, National Natural Science Foundation of China under grant No. 61972189, and the Shenzhen Key Lab of Software Defined Networking under grant No. ZDSYS20140509172959989.

References

[1] "Segment Routing Architecture," RFC 8402, IETF, July 2018.
[2] G. Trimponias, Y. Xiao, X. Wu, H. Xu, and Y. Geng, "Node-constrained traffic engineering: Theory and applications," *IEEE/ACM Transactions on Networking*, vol. 27, no. 4, pp. 1344–1358, 2019.
[3] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.
[4] J. Zhang and Y. Luo, "Degree centrality, betweenness centrality, and closeness centrality in social network," in *2017 2nd International Conference on Modelling, Simulation and Applied Mathematics (MSAM2017)*.
[5] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.
[6] S. Orlowski, M. Pióro, A. Tomaszewski, and R. Wessäly, "SNDlib 1.0–Survivable Network Design Library," in *Proceedings of the 3rd International Network Optimization Conference (INOC 2007)*.