Service-centric Segment Routing Mechanism using Reinforcement Learning for Encrypted Traffic

Van TONG*, Sami SOUIHI*, Hai Anh TRAN† and Abdelhamid MELLOUK*
*LISSI-TincNET Research Team, UPEC, France
† SOICT, HUST, Vietnam

***Abstract*—For the past decade, IP (Internet Protocol) routing approaches utilize TCAM (Ternary Content Addressable Memory) for the rule matching in the switches. These approaches are expensive and require more power consumption. Fortunately, the emerging of segment routing can resolve this drawback by encoding a routing path into the packet header to forward the packets to a destination. However, the standard segment routing algorithm has encountered a main problem. Using the shortest path to forward the packets can lead to a high traffic load on these paths and a performance reduction. It results in a decrease in user's perception and some negative economic impacts for ISPs (Internet Service Providers). Therefore, in this paper, we propose a novel service-centric segment routing mechanism using reinforcement learning in the context of encrypted traffic. Our proposal aims to help ISPs to decrease the influence of the network problems and meet the strict user's requirement related to QoE (Quality of Experience). The obtained results under the considered conditions demonstrate that our approach outperforms the standard segment routing algorithm and requires reasonable computational cost.**

I. Introduction

In 2020, the Covid-19 pandemic is happening all over the world, which causes many negative impacts on various fields (e.g. economy, education, tourism, etc.). As a result, some ISPs and network operators [1] have to suffer the high pressure from the significant amount of Internet traffic. During the confinement of the Covid-19 pandemic, traffic engineering, as well as routing approach, are the potential solutions to optimize the network and reduce the influence of the network problems (e.g. improving QoE, etc.).

In much existing research work [2], routing approaches that use TCAM for rule matching in the switches are implemented to reduce affections of problems in the network. Nowadays, with the rapid growth of multimedia services, there is a huge number of rules to be installed in the SDN switches, which requires more TCAM resource consumption [3]. Fortunately, segment routing is a potential promise to solve this disadvantage. Segment routing is a label-switching approach that encodes the routing paths into the packet's headers. The standard segment routing algorithm contains some drawbacks related to high traffic load and performance reduction in the unpredictable network conditions because it considers the shortest paths to forward the packets. Concretely, selecting the shortest paths can lead to a good performance in a short-term period, but these shortest paths can lead to some network problems (e.g. congestion, etc.) after a long-term period. Consequently, in this paper, the routing problem is formulated as a reinforcement learning task because reinforcement learning can learn from the network environment to make appropriate actions to optimize long-term network performance. Moreover, implementing the common routing mechanism for various kinds of services is not effective. Concretely, the QoE-aware routing mechanism identifies the routing paths using the current QoE states of the network topology. These approaches that use a common QoE estimator and same routing strategy for different kinds of services (e.g. video streaming, VoIP, etc.) are not effective because each service has a specific requirements related to QoS (Quality of Service) and QoE.

Therefore, in this paper, **we propose a novel service-centric SR (Segment Routing) mechanism using RL (Reinforcement Learning) in the context of encrypted traffic**. Our proposal aims to help ISPs to reduce the affections of network problems by selecting appropriate paths corresponding to different kinds of services and meet the strict user's requirements related to QoE. Nowadays, network traffic is encrypted to protect the user's privacy and data during the transmission, so the information about the class of service will be hidden. We consider QUIC (Quick UDP Internet Connection) protocol [4] which is a transport layer network protocol designed by Google from 2012. Much existing research work focus on the classification approaches for VPN and TLS/SSL [5], but we concentrate on a novel QUIC traffic classification approach to identify different kinds of services (e.g. VoIP, etc.).

Outline: The remainder of the paper is structured as follows. Section II introduces some related work in segment routing. In Section III, the paper presents the proposed service-centric segment routing mechanism. Section IV describes the experimental results of the proposed mechanism. Finally, the paper concludes with Section V which highlights some future works.

II. Related work

Barakabitze et al. [6] presented QoEMuSoRo, a QoS-based multipath source routing algorithm, to optimize some QoS by forwarding MPTCP (Multi-path TCP) flows using SR over SDN. Concretely, QoEMuSoRo selects the shortest path, which meets the strict requirements related to packet loss and delay.

Lee et al. [7] proposed a segment routing algorithm that can meet the bandwidth requirements. Besides, the proposed routing algorithm also considers the balance of traffic load using the link criticality and the reduction of the extra cost of the packet header's size.

Barakabitze et al. [3] proposed a QoS-aware SDN-based MPTCP/SR approach to improve the QoS for multimedia services over the 5G networks. This approach controls the network flows using the SDN controller and performs the source routing using SR.

978-3-903176-31-7 © 2020 IFIP

The current research works on segment routing consider the same solutions for various services, which is not effective with the rapid growth of many multimedia services. In this paper, **the proposed service-centric segment routing algorithm aims to implement a specific routing strategy for each service and meet the strict requirements of end-users in terms of QoE**.

III. Proposed Service-centric SR Mechanism for Encrypted Traffic

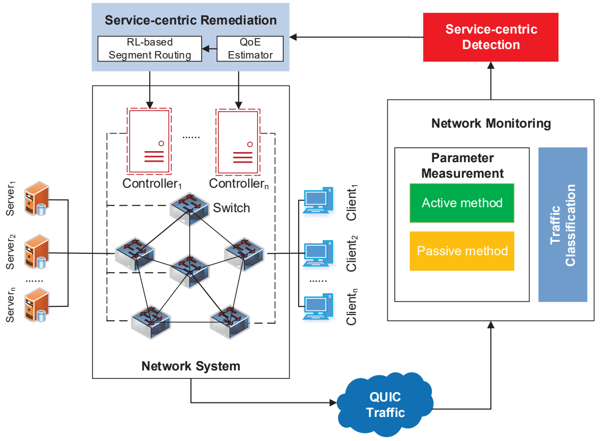


Fig. 1: The service-centric SR framework for encrypted traffic.

A. Overview of Service-centric SR Framework

In this paper, we present the overview of the service-centric SR framework for encrypted traffic (Fig. 1) and concentrate on a novel service-centric segment routing mechanism using reinforcement learning in the service-centric remediation module. The major components of the framework are as follows:

1) *Network Monitoring* contains two main modules, including parameter measurement and traffic classification modules. The parameter measurement module is to monitor and collect some network parameters, while the traffic classification module is to classify the encrypted traffic into different kinds of services.
2) *Service-centric Detection* considers the class of service and some network parameters from network monitoring to identify the abnormal symptoms of problems in the network.
3) *Service-centric Remediation* aims to reduce affection of problems by implementing different routing strategies corresponding to various services which meet the strict user's requirements in term of QoE.

B. Network Monitoring

1) Traffic Classification

Nowadays, network traffic is encrypted to protect data and user's privacy during transmission. Therefore, we present a novel traffic classification approach to identify different kinds of services including video streaming, file transfer and VoIP. After investigating some service's characteristics, we present a classification approach using deep learning and hybrid features (handcrafted features and implicit features). First, network traffic is classified into mice flows (VoIP) or elephant flows (video streaming and file transfer) using the random forest algorithm and some handcrafted features (flow-based features). The elephant flows will be then classified into video streaming or file transfer using convolutional neural network and implicit features (packet-based features). The detail of this approach is described in detail in [8].

2) Parameter Measurement

Similar to the traffic classification module, we implement a parameter measurement module to estimate several network parameters on each link in the network. Many service level agreements (SLAs) of service providers rely on some performance metrics containing packet loss and latency [9]. Consequently, we consider here several network parameters, including latency, packet loss, and link utilization [10].

C. Service-centric Detection

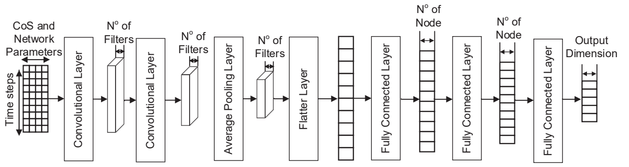


Fig. 2: The service-centric symptom detection approach.

In this section, we present a service-centric detection approach to identify abnormal symptoms. The detail of this approach is described in Fig. 2. The service-centric detection approach considers the class of services and some time-series network parameters on network flows such as latency, packet loss, and link utilization as the input. This information will then be analyzed using a convolutional neural network (CNN) to detect the abnormal symptoms of problems (e.g. increase of latency, packet loss, etc.). The input of this module is the time-series data, so some traditional machine learning algorithms (e.g. MLP, etc.) is not effective for the analysis. Moreover, some of the time-series network parameters are not effective for symptom detection, so 2D (2 dimensional) convolutional neural network is implemented to learn effective representations to enhance the accuracy of symptom detection. Next, this representation will be processed in 2D average pooling layer and flatter layer before fed into the fully connected layer to classify the network states into normal or abnormal symptoms.

D. Service-centric Remediation

Although there are many remediation approaches (e.g. routing approaches, load balancing, etc.), some existing researches [2], [11] implement the routing aspect to reduce the influence of network problems in the network. Therefore, in the service-centric remediation module, a service-centric segment routing mechanism using reinforcement learning is proposed to reduce the affection of problems by resolving it temporarily. The term *'service-centric'* means that we consider different kinds of strategies corresponding to a variety of services to improve the performance of the remediation module.

1) QoE Estimator

The rapid growth of the Internet leads to various kinds of multimedia services (e.g. video streaming, chat, etc.). Implementing the same QoE model for different kinds of services is not effective. Consequently, we implement a QoE estimator [12] to calculate the QoE of different services in the context of encrypted traffic using the class of service from the previous module (traffic classification). According to the traffic classification module, the QoE estimator can apply the appropriate QoE models corresponding to the specific services (video streaming, file transfer and VoIP) to enhance the accuracy of QoE estimation.

2) RL-based Segment Routing

The segment routing algorithm is considered here as a reinforcement learning task that takes into account QoE as environment feedback. The reinforcement learning model contains *agent, state, action, reward, and policy*. The detail is described as follows:

Agent: An entity in the network system implement a learning algorithm to execute its tasks. In the routing algorithm, an agent aims to identify the appropriate routing paths to maximize the reward.

State: A snapshot of the network environment observed by an agent.

Action: An action describes how an agent responds to the network environment. In the routing algorithm, an action is a routing path between a source and a destination in the network.

Policy: A policy is a map between a state and an action in the network environment.

Reward: A reward is feedback in which the network environment returns to the agent. In the routing algorithm, the agent observes the network state s and implements an action a from the routing policy. Next, the agent moves to next state s' and receives a reward *r*. We consider here QoE of the chosen path as the reward in which the network environment returns to the agent. The QoE is calculated via QoE estimator (section III-D1).

The objective of reinforcement learning task is to optimize the objective function O_f in order to maximize the expected cumulative reward (Eq. 1):

$$O_f = Max\ E[\sum_{t=0}^{\infty} \gamma_t \times r_t]. \quad (1)$$

where $\gamma_t \in$ [0, 1] is the discount factor.

We present the routing information in a Q-table. Each value in the Q-table represents a Q-value of an action (a routing path) in the network environment. According to the centralized architecture of SDN, all routing paths in the network can be extracted in the SDN controller. Our approach is based on reinforcement learning to learn the actions corresponding to the network states to make control decisions. The agent executes an action and receives the immediate reward r. Using this immediate reward and the long-term reward, it can update the Q-table that influences on the future routing policy. When an action is chosen, the Q-value of this action in the Q-table is updated as in Eq. 2:

$$Q(s, a) = (1 - \alpha)Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a')]. \quad (2)$$

where α is the learning rate and γ is the discount factor.

3) Exploration and Exploitation Tradeoff

To obtain the optimal cumulative reward, reinforcement learning needs to balance between the exploration and exploitation phase. We can not systematically implement an exploitation phase that selects the action with maximal Q-value. Each routing path needs to be evaluated many times to obtain the reliable expected reward. Therefore, we consider the tradeoff between the exploration and exploitation phase. This tradeoff is formalized as a MAB problem (Multi-Armed Bandit). In this paper, we implement some popular algorithms to resolve the MAB problem including *ϵ-greedy, softmax* and *UCB1* (Upper Confidence Bounds) [13].

IV. Experimental results

A. Experiment Setup

To evaluate the performance of the novel service-centric segment routing mechanism, we set up a testbed with *mininet* v2.2 and *ONOS* controller v2.4. The uniform link capacity is set to 10 Mbps, and we generate a large amount of traffic (e.g. 60-100 Mbps, etc.) between clients and servers to emulate the network problems (high sending rate). The network topology contains five spine nodes and two leaf nodes which connect to every spine node. In the reinforcement learning algorithm, α and γ are set to 0.7 and 1, respectively. These parameters are chosen via the experiments. The source code of the service-centric segment routing framework is available at [14].

B. Benchmark

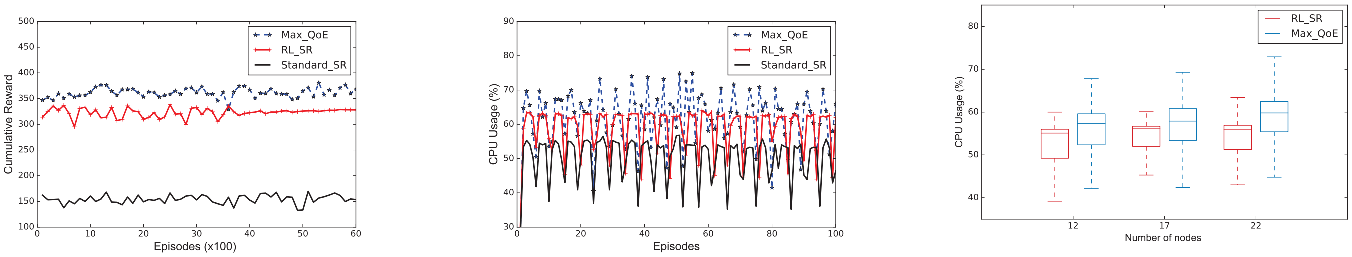
To validate the performance of the proposed segment routing mechanism, we compare our proposal with some benchmark including standard SR (*Standard_SR*) and SR with maximal QoE (*Max_QoE*). *Standard_SR* selects the shortest paths between clients and servers to forward the packets. *Max_QoE* identifies all available routing paths between clients and servers using SDN controllers. After that, it calculates the QoE of these paths and selects the path with maximal QoE (maximal reward). In the proposal (*RL_SR*), we only calculate the QoE of the chosen path which is selected via some selection algorithms to reduce the computational cost.

These approaches are evaluated via some performance metrics including *cumulative reward* and *CPU usage*. *Cumulative reward* is to evaluate the performance of the algorithm over time while *CPU usage* is measured as a percentage of the CPU's capacity which is measured via *psutil* library in Python [15].

C. Performance Analysis

In this section, we evaluate the performance of several segment routing algorithms (*Standard_SR, Max_QoE and RL_SR*). We consider here video streaming as a use-case to evaluate the performance of these approaches, and the figure for other services (file transfer and VoIP) will be considered in our future work.

There are some selection algorithms (e.g. UCB1, softmax, etc.), so we evaluate the performance of these algorithms to choose the appropriate selection algorithm. Fig. 4 illustrates the cumulative reward of *UCB1, softmax*, and *ϵ-greedy* in proposed segment routing mechanism. The cumulative rewards of *softmax* algorithm is better than the others. During the first 4,000 episodes, the cumulative reward of three algorithms fluctuate between 270 and 340. After that, the cumulative



(a) Cumulative reward.

(b) CPU usage.

(c) CPU usage against N^o nodes.

Fig. 3: Comparison between three kinds of segment routing mechanism.

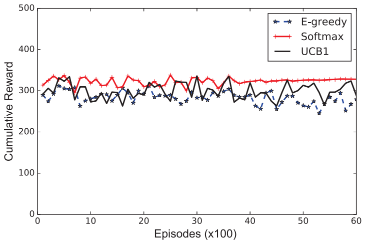


Fig. 4: Cumulative reward of three selection algorithms in proposed segment routing mechanism.

reward of *softmax* algorithm converges around 325.6 while the other algorithms continue to fluctuate. ϵ*-greedy* algorithm selects the action with maximal reward in the following stage, so some less-used action will not be chosen frequently. As a result, the figure for ϵ*-greedy* is lowest in three algorithms. *softmax* algorithm selects the action based on the probability function of reward, so the less-used action will be chosen more frequently. Therefore, the cumulative reward of *softmax* algorithm is better than the others, and we use *softmax* algorithm as the selection algorithm in reinforcement learning for the following experiments.

After choosing the selection algorithm in RL, we analyze the performance of different segment routing algorithms including *Max_QoE, RL_SR*, and *Standard_SR* (Fig. 3a). *Standard_SR* algorithm selects the shortest path to forward the packets between clients and servers, so the cumulative reward is lower than the others. *Max_QoE* algorithm systematically selects the paths with maximal reward, so the figure of this algorithm is slightly higher than the figure for our proposal (*RL_SR*). However, *Max_QoE* algorithm incurs a high computational cost, so it is not effective with the large-scale network. The CPU usages of three segment routing algorithms are described in Fig. 3b.

The CPU usage of *Max_QoE* algorithm is the highest in three segment routing mechanisms, and it requires up to approximately 74 percent of CPU's capacity. *RL_SR* algorithm requires about 62 percent of CPU's capacity, and the required CPU's capacity usage reduces to 55 percent for *Standard_SR* algorithm. *Max_QoE* algorithm needs to measure the network parameters of all links in the network, and then calculate the QoE of all available routing paths. As a result, it requires more computational cost. *RL_SR* algorithm only needs to measure the network parameters in the links of a chosen routing path and calculate the QoE for this path, so it requires less computational cost in comparison with *Max_QoE* algorithm. A noticeable feature from Fig. 3b is that the CPU usage falls to the bottom periodically. The reason for it is that we set a small sleep duration after a given time of packet generation between clients and servers.

To evaluate the computational cost of *RL_SR* and *Max_QoE* algorithms thoroughly, we investigate the CPU usage of these algorithms against the increase of a number of nodes in the network topology (Fig. 3c). When we increase the number of nodes in the network topology, the CPU usage of *RL_SR* algorithm is more stable than the figure for *Max_QoE* algorithm. The variation of CPU usage in *Max_QoE* algorithm will increase against the growth of a number of nodes in the network topology. Therefore, *Max_QoE* algorithm is not effective in the large-scale network (e.g. a network with few thousands of nodes, etc.).

V. Conclusion

In this paper, we proposed a novel service-centric segment routing mechanism for encrypted traffic. The proposal can reduce the affection of network problems by selecting the appropriate routing paths that meet strict user's requirements related to QoE. The experimental results show that the proposal obtains better cumulative rewards in comparison with the standard one.

In the future, concerning the scalability, we plan to investigate the proposed routing mechanism's performance in the large-scale network to evaluate it thoroughly.

Acknowledgment

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.02-2019.314.

References

[1] AT&T, "At&t: Sdn, nfv helped meet covid-19 traffic demands," 2020. [Online]. Available: https://www.lightreading.com/cloud-native-nfv/atandt-sdn-nfv-helped-meet-covid-19-traffic-demands/d/d-id/758661?

[2] P. Jurkiewicz, R. Wójcik, J. Domżał, and A. Kamisiński, "Testing implementation of famtar: Adaptive multipath routing," *Computer Communications*, vol. 149, pp. 300–311, 2020.

[3] A. A. Barakabitze, L. Sun, I.-H. Mkwawa, and E. Ifeachor, "A novel qoe-centric sdn-based multipath routing approach for multimedia services over 5g networks," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–7.

[4] A. Langley, A. Riddoch, A. Wilk, A. Vicente, C. Krasic, D. Zhang, F. Yang, F. Kouranov, I. Swett, J. Iyengar *et al.*, "The quic transport protocol: Design and internet-scale deployment," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*. ACM, 2017, pp. 183–196.

[5] P. Wang, F. Ye, X. Chen, and Y. Qian, “Datanet: Deep learning based encrypted network traffic classification in sdn home gateway,” *IEEE Access*, vol. 6, pp. 55 380–55 391, 2018.

[6] A. A. Barakabitze, I.-H. Mkwawa, L. Sun, and E. Ifeachor, “Qualitysdn: Improving video quality using mptcp and segment routing in sdn/nfv,” in *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*. IEEE, 2018, pp. 182–186.

[7] M.-C. Lee and J.-P. Sheu, “An efficient routing algorithm based on segment routing in software-defined networking,” *Computer Networks*, vol. 103, pp. 44–55, 2016.

[8] V. Tong, H. A. Tran, S. Souihi, and A. Mellouk, “A novel quic traffic classifier based on convolutional neural networks,” in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[9] D. Frost and S. Bryant, “Packet loss and delay measurement for mpls networks,” RFC 6374, september, Tech. Rep., 2011.

[10] K. Phemius and M. Bouet, “Monitoring latency with openflow,” in *Proceedings of the 9th International Conference on Network and Service Management (CNSM 2013)*. IEEE, 2013, pp. 122–125.

[11] G. Kim, C. Kim, J. Jeong, M. Parker, and J. Kim, “Contention-based congestion management in large-scale networks,” in *2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 2016, pp. 1–13.

[12] L. Amour, V. Tong, S. Souihi, H. A. Tran, and A. Mellouk, “Quality estimation framework for encrypted traffic (q2et),” in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[13] H. A. Tran, S. Hoceini, A. Mellouk, J. Perez, and S. Zeadally, “Qoe-based server selection for content distribution networks,” *IEEE Transactions on Computers*, vol. 63, no. 11, pp. 2803–2815, 2013.

[14] V. Tong, “Service-centric segment routing using reinforcement learning,” July 2020. [Online]. Available: https://github.com/vanvantong/rl-sr

[15] PyPI, “Measurement for cpu usage.” [Online]. Available: https://pypi.org/project/psutil/