Improving QoS in Failure Scenarios: Measurement System for Low-Latency Backup Path

Yoshihiko Ito*, Jin Nakazato*, Ryusei Shiiba^Γ, Kensuke Fukuda^Γ,$, Hiroshi Esaki*, Hideya Ochiai*,
*Graduate School of Information Science and Technology, The University of Tokyo, Tokyo, Japan
^Γ Department of Informatics, Graduate University for Advanced Studies (SOKENDAI), Tokyo, Japan
$National Institute of Informatics, Tokyo, Japan

Email: {edoalu-natsu72, jin-nakazato, ochiai}@g.ecc.u-tokyo.ac.jp, {siiba, kensuke}@nii.ac.jp,
hiroshi@wide.ad.jp

***Abstract*—In modern networks, ensuring low latency and high reliability is critical, especially for real-time applications such as voice or video communication applications. Traffic Engineering (TE) enables efficient optimization of resource allocation, leading to low delay. Segment Routing (SR) is utilized to achieve such TE in a simple and scalable manner. Traditional Fast Reroute (FRR) technology in SR networks provides rapid recovery during network failures. However, these technology often ignores the latency of backup paths and there is no architecture to measure latency in FRR network. This study presents a system architecture to measure the latency metrics of the backup paths and enhance their performance of backup paths provided by FRR in SR networks. The architecture allows for the visualization and evaluation of alternative paths by aggregating latency metrics to the controller for path computation. We evaluated our proposed system in a test environment consisting of six virtual routers, and the results demonstrated its ability to identify lower-latency paths compared to a backup path selected by the conventional FRR technology. The ability indicates the possibility of lower-latency rerouting strategies that achieve both high reliability and performance.**

***Index Terms*—Fast Reroute, low latency and Segment Routing**

I. Introduction

In recent years, essential Internet services that require real-time communication, such as video call applications, online gaming, and live streaming, are becoming integral in daily life and business environments. Feldmann et al. [1] noted that video conferencing and online education traffic more than doubled during the COVID-19 lockdowns. Futhermore, the number of Internet users itself is also increasing, Cisco reported that by 2018, global Internet users had surged to 3.9 billion, accounting for 51 % of the world's population [2]. These statistics underscore the rapid global expansion of the Internet and demonstrate the increasing demand for real-time services.

The quality of the networks depends on whether they are highly reliable and low-latency, which will lead to the continuous provision of high-quality and real-time network services. A key metric for network communication quality is Quality of Service (QoS) [3]. QoS is assessed using quantifiable metrics such as latency and jitter. Furthermore, choosing network routes based on these metrics is QoS Routing [4]. In general, IP networks operate under the best-effort principle, making it challenging to handle services sensitive to communication quality such as real-time communications. Dynamic routing protocols such as IS-IS and OSPF, which are link-state-based, select the path with the lowest total cost assigned to each link. QoS Routing considers multiple QoS requirements and selects the path with the lowest cost that meets these requirements. However, algorithms that select paths considering multiple requirements are known as NP-complete [5]. To loosen this complexity, the following step is often considered [6], [7]:

- Collect QoS metrics from routers on the network and pass them to an external controller
- Compute the optimal paths at the controller
- Redistribute the computed paths to the routers

To collect QoS metrics such as latency and jitter, it is necessary to send ICMP packets along arbitrary paths. Measurement tools, such as the Two-Way Active Measurement Protocol (TWAMP) [8] and ping CLI, are commonly used.

Segment Routing (SR), as specified in RFC 8402 [9], is considered as suitable approach for flexible routing. SR is a routing protocol designed to simplify traffic control on a per-user or per-service basis, so it enables flexible Traffic Engineering tailored to the needs of individuals or services . In SR, labels called "segments" are inserted into data packets at source node, and intermediate nodes forward the packets according to the segments. SR enhances the flexibility of network routing and enables efficient control of latency and throughput in End-to-End (E2E) communication paths [10]. Throughout this study, E2E is defined to the path from the ingress to the egress node in an SR network.

To ensure highly reliable network, there are two key technologies available. The first involves duplicating packets at the headend router and forwarding them via multicast, with the duplicated packets being processed at the destination [15]. Although this approach can reduce the packet loss rate to 0 %, it has limitations, such as requiring a network with abundant resources to avoid impacting service quality and necessitating exceptional processing capabilities at the destination. The second method involves pre-configuring both primary and backup paths and switching to the backup path

TABLE I: Comparison between previous studies about performance measurement and evaluation of SR networks

Reference	Evaluation Target	TI-LFA	QoS	Metric Target	Measurement Tool	SDN
Ref. [11]	Only Primary Path	×	×	topology, flow	pcep [12]	✓
Ref. [13]	Backup Path	✓	×	topology, igp link-state	pcep or igp	✓
Ref. [14]	Primary and Backup Path	×	✓	latency and packet loss rate	TWAMP	✓
This study	Backup Path	✓	✓	latency and packet loss rate	gRPC and ICMP	✓

in case of failure [16]. The service downtime will occur due to path switching, but the pre-installing paths can minimize the downtime. Fast Reroute (FRR) [17], [18] is used as the technology to achieve this.

FRR is useful for minimizing downtime in the event of a failure, but it has an issue with QoS. FRR pre-installs alternate routes in the routing tables of each router for cases where adjacent routers or links fail, allowing for fast rerouting at the router adjacent to the failure. Traditional FRR technology aims to reduce service downtime but does not guarantee the QoS of the backup paths it configures. This lack of the QoS could significantly degrade user experience. To avoid this drawback, we combine the concept of QoS Routing with SR and FRR technologies to provide highly reliable networks and low latency.

Our goal is to provide networks with minimal downtime, high reliability, and low latency. We propose incorporating our novel system into FRR to ensure both high reliability and low latency of backup paths. This study evaluates the latency of backup paths provided by FRR and explores the availability of lower-latency backup paths. The contributions of this study are as follows:

- We propose a framework for measuring QoS of backup paths using gRPC and SR.
- We demonstrated the usefulness of the system by deploying the proposed system on a network of six routers and collecting actual latency metrics. We explore backup paths and visualize the latency for each path.

The structure of this study is as follows. Section II introduces related technologies and studies, including SR technology, and research on QoS routing. Section III organizes this study's system requirements and architecture. Section IV describes the experimental results of the proposed system. Finally, Section V concludes this study and explores potential future directions.

II. Related Work

In this section, we describe several studies about measuring and improving network quality using SR. Table I summarizes the differences between related studies and this study. TI-LFA is one of the FRR computation algorithms available for SR networks, offering complete coverage against a single failure. We focus on this technology and investigate whether existing methods can adequately support it. Many researchers have attempted to measure and improve network performance using SDN controllers to manage the network centrally in SR networks, such as [11], [13],and [14]. Although Eryc *et al.* [11] succeeded in enhancing network quality, they focused on the performance of primary paths without considering the paths used during failures.

On the other hand, Vitor *et al.* [13] worked on improving the performance of backup paths provided by TI-LFA and improved the hop count and Segment List size. However, they did not take QoS into account. Zhenlin *et al.* [14] proposed a method for constructing network slices in the SRv6 environment. They employed a path computation strategy that considers latency and packet loss constraints for primary and backup paths. While they succeeded in providing backup paths that meet network quality requirements, their approach does not support Local Repair, which switches paths at the node adjacent to the failure, such as TI-LFA.

As mentioned above, several studies have enhanced network performance, and evaluated paths during failures. However, when using TI-LFA, we assume the solutions need to be completed. Hence, in this study, we propose a system for measuring the latency of all potential candidate paths, including those provided by TI-LFA, that could be traversed during Local Repair.

III. Proposed Method

In this section, we propose a system designed to measure all potential candidate paths that can be traversed using FRR in the event of a protection scenario.

A. Overview

The main components of the proposed system, as shown in Fig. 1, include below functions:

- Topology Manager (TMG) manages network topology information.
- Metrics Collector (MC) collects latency metrics from routers.
- Path Computation Element (PCE) manages SR Policies (SRPs). SRP is a mechanism that defines explicit policies using Segment List to control the path traversed by packets for specific traffic flows.

Given topology information, the TMG generates a graph. It then establishes gRPC sessions to all routers to collect SIDs and reflect them in the graph. Next, it calculates the paths considering the given protection scenario and creates SRPs to route packets along these paths. The generated SRPs are then applied to the PE routers via the PCE. The procedures mentioned above constitute the processes that should be performed before measurement. Once the preparation is complete, the MC generates ICMP packets associated with the SRPs and instructs the routers to send them to destination router. The measurement is carried out according to this flow.

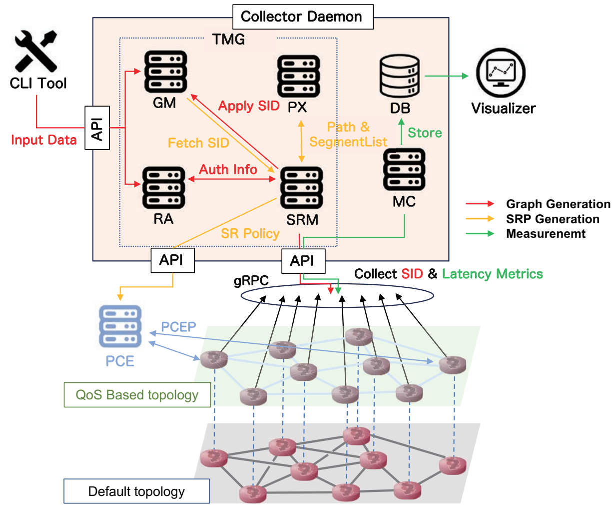

Fig. 1: System architecture for collecting latency metrics (TMG: Topology Manager, GM: Graph Manager, RA: Router Authenticator, SRM: Segment Routing Manager, MC: Metrics Collector)

We describe the functionalities that the proposed system must possess in the subsection III-B and the roles of the individual components within the system in the subsection III-C. In the subsection III-D, we explain the overall process and workflow of the system. Finally, we outline the candidate path exploration algorithm in the subsection III-E.

B. System Requirements

*1) **Ability to Measure latency**:* The system must be able to utilize gRPC communication since it has a small overhead. In this study, gRPC is used for communication between the orchestrator and routers during the collection of SIDs and latency metrics.

*2) **Determination of Measurement Points**:* The system requires the functionality to determine candidate paths for measurement based on the network topology information and protection scenarios. This involves calculating potential backup paths provided by FRR, using information about the topology ’ s nodes and links, as well as data on which nodes or links are considered down.

*3) **Ability to Use SR**:* The system must be able to measure latency by sending measurement packets along the candidate paths. Consequently, the orchestrator is responsible for determining the routing of these measurement packets. SRPs, which are tailored for each candidate path, are employed to guide the transfer of measurement packets according to these SRPs.

*4) **Ability to Use PCE**:* The system has the capability to interface with the PCE, which is utilized to apply the SRPs generated by the SRM on the routers.

C. Architecture

*1) **Topology Manager**:* The TMG manages network topology information and comprises several key components: Graph Manager (GM), which generates graphs from topology information; Router Authenticator (RA), responsible for managing authentication information to establish gRPC sessions with routers; Path Explorer (PX), which searches for paths based on the graph and protection scenarios generated by GM, and Segment Routing Manager (SRM), which collects SIDs from routers and generates SRPs from candidate paths provided by PX. In addition, PX generates Segment Lists associated with the calculated candidate paths.

*2) **Metrics Collector**:* The MC collects latency metrics from routers and stores them in a database. Based on the SRPs generated by TMG, the MC generates ICMP packets and sends them in parallel along each candidate path to collect latency metrics.

*3) **Input Data**:* The input includes network topology information, gRPC authentication information, and protection scenarios. The topology information provides router hostnames, IP addresses, and adjacency details. Protection scenarios include link and node failures, where the former provides the nodes and interface names at both ends of the downlink, and the latter offers the node names.

D. System Flow

*1) **Graph Generation Process**:* The system inputs topology information, router authentication information, and protection scenarios from the user. Initially, the GM creates a graph based on the topology information. Subsequently, RA establishes gRPC sessions with routers using the authentication information. Once these sessions are established, SRM collects Node SIDs and Adj-SIDs from the routers and registers them in the graph. At this point, the graph contains information about which interface to exit from each node to reach adjacent nodes, along with the SID information associated with those interfaces. Finally, the protection scenarios are registered on the graph, completing the graph generation process.

*2) **SRP Generation and Application Process**:* The PCE is used to apply the generated SRPs to routers. Initially, the RA establishes a gRPC session with the PCE. Subsequently, the PX performs path exploration. Since FRR detects failures only at the adjacent nodes (PLR), it switches the path to the bypass route after packets are transferred to the PLR during a failure. Therefore, PX uses a Depth First Search (DFS) algorithm that allows one backtrack to calculate candidate paths that can be used as backup paths provided by FRR based on the topology and protection scenario information registered in the graph. Segment Lists are also generated using the Adj-SIDs registered in the graph to ensure packets can be forwarded along the candidate paths. Next, the SRM generates SRPs for each candidate path, and PX calculates the Segment List. After verifying that the PCEP session is established between the headend node and PCE, SRM adds the SRPs to the routers via the PCE.

*3) **Measurement Process**:* After confirming that the SRPs associated with the target candidate paths are active on the routers, the MC generates ICMP packets associated with these SRPs and distributes them to the routers. The routers execute

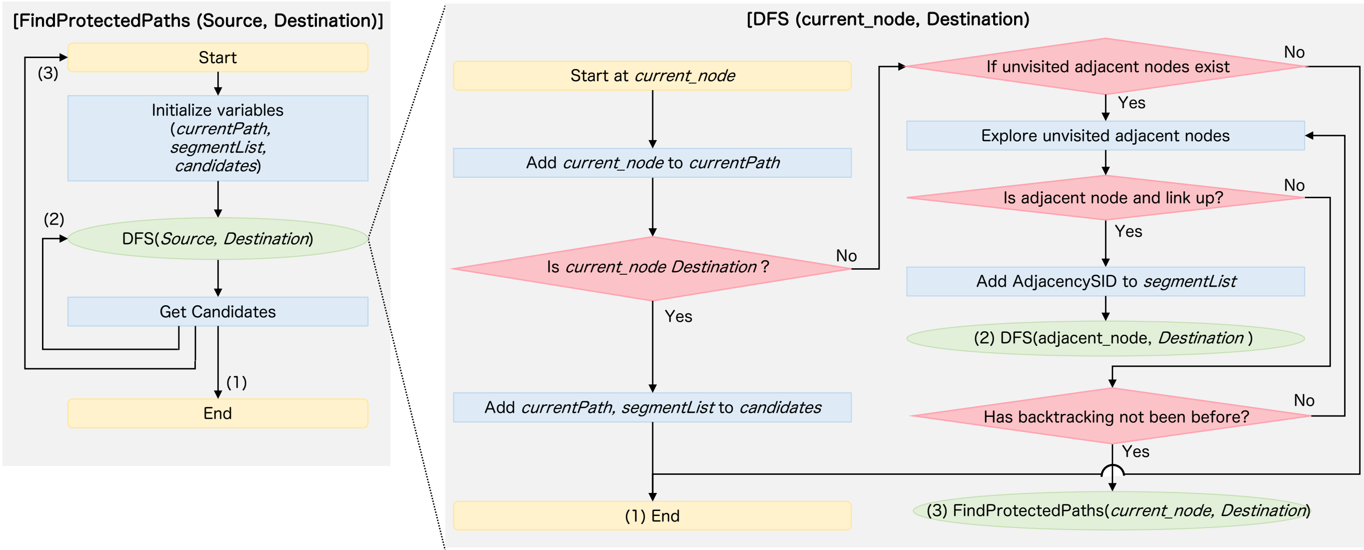

Fig. 2: Path exploration algorithm flowchart: this algorithm is based on DFS and allows backtracking only once.

TABLE II: Details of systems versions used in experiments

Name	OS	Kernel Version
Cisco XRv9000	IOS-XR7.9.1	
Traffic Control Agent (TCA)	Ubuntu22.04	5.15

these commands, and the latency metrics obtained are returned to the MC, which stores the latency metrics in the database. From confirming SRP activation to storing the data, this entire flow is repeated for each candidate path at regular intervals.

E. Path Exploration Algorithm

PX performs path exploration based on the generated graph and failure scenarios, following the process shown in Fig. 2. The exploration proceeds from the source node to the destination node using DFS. Considering that FRR involves rerouting at the node adjacent to the failure (Point of Local Repair, PLR), the search history is reset once at the PLR during the exploration, and the search is restarted from the PLR as the new starting point. Simultaneously, with path exploration, the Segment List is generated.

IV. Experiments

A. Experimental Setup

An experiment will be conducted with a six-router configuration to investigate the feasibility of the proposed system by measuring the backup path. We used VirtualBox 6.1.50 and Cisco XRv9000 vRouters summarized in Table II. Fig. 3 illustrates the network configuration, which included two Provider Edge (PE) routers and four Provider (P) routers. All routers were configured and communicated with the orchestrator via gRPC. In addition, the PE routers also interacted with the PCE using PCEP. Here, we adopted POLA PCE v1.3.0 as the PCE. We simulated a protection scenario with a link failure between R2 and P3 in this experiment, assuming traffic from PE1 to PE2. First, using a Command Line Interface (CLI) tool, we

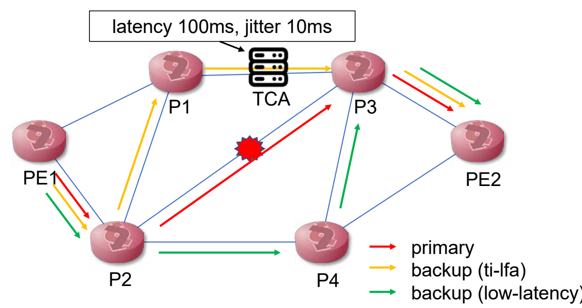

Fig. 3: Topology for the experiment: when the P2-P3 link protection scenario: the yellow line represents the backup path provided by TI-LFA, and the green line indicates the low-latency alternative path.

configured the network topology, protection scenario, and PCE setting. The orchestrator then set up the SRPs. We started the latency measurement after confirming the active SRPs were properly configured. Regarding to latency measurements, we utilized the ping CLI to collect QoS metrics, including latency. In the latency measurement process, the MC measured the latency of arbitrary paths by using gRPC to issue commands in the YANG data model to the PE router, instructing it to execute the ping CLI using the registered SRP.

To emulate network latency and jitter, we used ”tc” command. A device called the Traffic Control Agent (TCA), which executes the tc command, was placed between the P1 and P3 link as shown in Fig. 3, where a latency of 100 ms and a jitter of 10 ms were configured. When a primary path transfers traffic from PE1 to PE2 via P3, and the link between P2 and P3 goes down, P2 switches the traffic towards P1 as an alternate route according to the TI-LFA calculation algorithm. However, since the TCA causes a latency between P1 and P3, it would be more efficient to switch the traffic to P4 to achieve

Fig. 4: Visualization results: TCA caused a latency at 14:58 and resolved at 15:08. During that time, a lower-latency backup path existed compared to the path provided by TI-LFA.

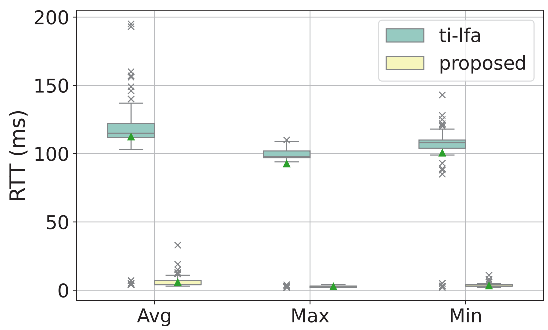

Fig. 5: Latency results: on the TI-LFA backup path, RTT increased to about 100 ms due to latency caused by the TCA. In contrast, the detour path via P4 resulted in a shorter RTT.

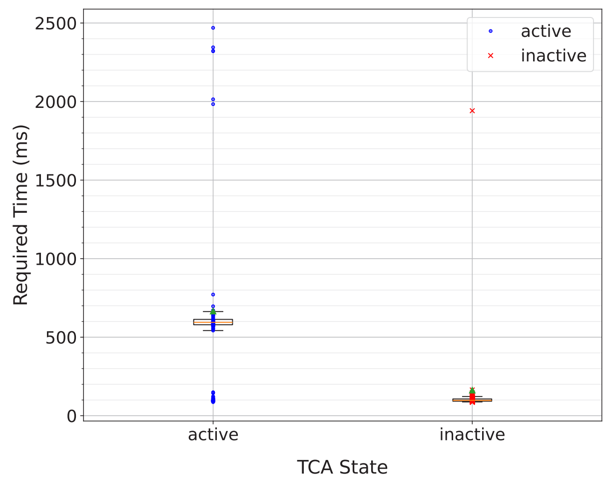

Fig. 6: Required time to collect latency metrics along the path provided by TI-LFA. Active: The TCA has caused a 100 ms latency. Inactive: No latency is present.

lower latency. This experiment used a proposed measurement system to discover a lower-latency path than the backup path originally calculated by TI-LFA.

B. Results

The proposed system was demonstrated to be more effective than existing solutions in enabling real-time detection and handling of issues such as latency and jitter. Figs. 4 and 5 display the results of executing the ping CLIs for two candidate paths. These figure compare the performance of a backup path calculated by TI-LFA with that of an alternative route discovered by our implemented system. As depicted in Fig. 4, it can be observed that approximately two minutes after the start of the measurement, latency and jitter were introduced by the TCA became apparent. The measurement continued for about 10 minutes, after which the latency and jitter issues were resolved. From this experiment, the proposed system allows users to verify the poor network quality of the path provided by TI-LFA and obtain better-quality information about paths.

Furthermore, the overhead of the proposed system will be discussed. Fig. 6 details the time required for the latency measurement process. When a 100 ms latency was set by the TCA, the measurement process took approximately 600 ms. In contrast, under conditions with no latency setup, the process required only about 100 ms. Besides, in some instances, we observed an overhead of nearly 2s.

C. Discussion

As described in subsection IV-B, our proposed system successfully measured the latency metrics for both the backup and other candidate paths. However, there are still some challenges. Specifically, improvements are necessary in addressing the following three issues:

1) The RTT problem associated with latency measurement needs refinement.
2) The scalability of the proposed system requires enhancement.
3) The overhead of the proposed system has been approximately 2s.

1) RTT problem in measurement process: When using the ping CLI, if we only specify the SRP for the outgoing path to the ingress node, the return path will follow the routing table by default. As a result, the forward and return paths may differ. To avoid this, it is necessary to also specify the SRP on the egress node and configure the appropriate routing policies.

2) Scalability of proposed system: Additionally, scalability is an issue. We used the DFS algorithm for path exploration, which has a computational complexity that is a linear function of the number of nodes and links. However, since we have imposed the constraint that one backtrack is allowed, the computational complexity is higher than that. As such, more efficient exploration algorithms need to be considered. Furthermore, ICMP packets are only transferred from the ingress to the egress node due to the PCEP session being limited to the PE routers. As the network topology expands, the number of candidate paths will inevitably increase, leading to a higher load and more measurement traffic on the PE routers. This challenge could be resolved by configuring SRPs on P routers, which would allow for measurement across more finely divided topology segments.

3) Overhead of proposed system: Currently, in this study, we executed the ping CLI using gRPC, but there is an overhead of approximately 100 ms. In addition, there are occasional instances where an overhead of 2s occurs, which needs to be mitigated. Previous research has used TWAMP to collect metrics such as latency and jitter [14]. Moving forward, we plan to experiment with TWAMP for metric collection to potentially improve efficiency and accuracy.

V. Conclusion

This study proposed and evaluated a system designed to measure and reduce the latency of the backup paths provided by FRR in SR networks. By integrating FRR with QoS metrics, the system enables rapid recovery during network failures and allows for the visualization and evaluation of alternative paths. Experimental results demonstrated that this system could identify more optimal paths than those provided by conventional FRR, emphasizing the critical importance of incorporating network performance in rerouting strategies.

Future work will focus on enhancing the system's scalability and efficiency with large-scale networks. A detailed analysis will be conducted to identify and mitigate the causes of 2s overhead in the proposed system. In addition, we plan to introduce more advanced measurement tools, such as TWAMP, and to optimize path exploration algorithms to reduce computational complexity. Ultimately, this research contributes to the development of a more robust and high-quality network infrastructure that can meet the growing demands of real-time applications.

Acknowledgment

This research was supported by JST ASPIRE JPMJAP2325.

References

[1] Anja Feldmann, Oliver Gasser, Franziska Lichtblau, Enric Pujol, Ingmar Poese, Christoph Dietzel, Daniel Wagner, Matthias Wichtlhuber, Juan Tapiador, Narseo Vallina-Rodriguez, Oliver Hohlfeld, and Georgios Smaragdakis. The lockdown effect: Implications of the covid-19 pandemic on internet traffic. In *Proceedings of the ACM Internet Measurement Conference*, IMC ’20. ACM, October 2020.

[2] Cisco. Cisco annual internet report (2018–2023) white paper. Technical report, Cisco, 2020.

[3] Rafal Stankiewicz, Piotr Cholda, and Andrzej Jajszczyk. Qox: What is it really? *IEEE Communications Magazine*, 49(4):148–158, 2011.

[4] Kavita Jaiswal and Veena Anand. An optimal qos-aware multipath routing protocol for iot based wireless sensor networks. In *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*, pages 857–860, 2019.

[5] Zheng Wang and J. Crowcroft. Quality-of-service routing for supporting multimedia applications. *IEEE Journal on Selected Areas in Communications*, 14(7):1228–1234, 1996.

[6] Taehong Kim and Thang Nguyen-Duc. Oqr: On-demand qos routing without traffic engineering in software defined networks. In *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*, pages 362–365, 2018.

[7] Yu Zhang, Mengze Cui, Mina Abadeer, and Sergei Gorlatch. A qos-aware routing mechanism for sdn-based integrated networks. In *2023 International Conference on Information Networking (ICOIN)*, pages 287–292, 2023.

[8] Yung-Chang Lai, Jhih-Dao Jhan, Wen-Che Yang, Fei-Hua Kuo, and Tai-Chueh Shih. Quality of service measurement mechanism of cloud-based network architecture. In *2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pages 1–4, 2019.

[9] Clarence Filsfils, Stefano Previdi, Les Ginsberg, Bruno Decraene, Stephane Litkowski, and Rob Shakir. Segment Routing Architecture. RFC 8402, July 2018.

[10] Koichiro Kanaya, Yasunobu Toyota, Wataru Mishima, Hiroki Shirokura, and Hiroshi Esaki. Qoe-aware content oriented path optimization framework with egress peer engineering. In *2022 Tenth International Symposium on Computing and Networking (CANDAR)*, pages 36–45, 2022.

[11] Eryk Schiller, Chao Feng, Rafael Hengen Ribeiro, Francesco Marino, Martin Buck, and Burkhard Stiller. Demo: Utilizing srv6 to optimize the routing behavior for tactical networks. In *2023 IEEE 24th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 361–363, 2023.

[12] Irena Šeremet and Samir Čaušević. An analysis of reconvergence delay when using bgp-ls/pcep as southbound protocols. In *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 415–420, 2019.

[13] Vítor Pereira, Miguel Rocha, and Pedro Sousa. Traffic engineering with three-segments routing. *IEEE Transactions on Network and Service Management*, 17(3):1896–1909, 2020.

[14] Zhenlin Tan, Zhuojun Huang, Peiyong Ma, Shuangfeng Lan, Yirong Zhuang, Yu Jiang, and Xiaobin Liang. A low-latency and high-reliability slice for ip backbone network based on srv6. In Qi Liu, Xiaodong Liu, Jieren Cheng, Tao Shen, and Yuan Tian, editors, *Proceedings of the 12th International Conference on Computer Engineering and Networks*, pages 242–253, Singapore, 2022. Springer Nature Singapore.

[15] Alan Ford, Costin Raiciu, Mark J. Handley, Olivier Bonaventure, and Christoph Paasch. TCP Extensions for Multipath Operation with Multiple Addresses. RFC 8684, March 2020.

[16] Jozef Papán, Pavel Segeč, Marek Moravčík, Jakub Hrabovský, Ľudovít Mikuš, and Jana Uramova. Existing mechanisms of ip fast reroute. In *2017 15th International Conference on Emerging eLearning Technologies and Applications (ICETA)*, pages 1–7, 2017.

[17] Andrzej Kamisiński. Evolution of ip fast-reroute strategies. In *2018 10th International Workshop on Resilient Networks Design and Modeling (RNDM)*, pages 1–6, 2018.

[18] Klaus-Tycho Foerster, Andrzej Kamisix0144;ski, Yvonne-Anne Pignolet, Stefan Schmid, and Gilles Tredan. Improved fast rerouting using postprocessing. *IEEE Transactions on Dependable and Secure Computing*, 19(1):537–550, 2022.