Comparison of New Solutions in IP Fast Reroute

Jozef Papan[1, *], Ivana Bridova[1], Peter Brida[2], Michal Hraska[1], Slavomir Tatarka[1], Oleksandra Yeremenko[3]

[1]Faculty of Management Science and Informatics, University of Zilina, Univerzitna 8215/1, 010 26 Zilina, Slovakia

[2]Faculty of Electrical Engineering and Information Technology, University of Zilina, Univerzitna 8215/1, 010 26 Zilina, Slovakia

[3]Institute of V.V. Popovskyy, Department of Infocommunication Engineering, Kharkiv National University of Radio Electronics, 61166 Kharkiv, Ukraine

[*]jozef.papan@fri.uniza.sk, ivana.bridova@fri.uniza.sk, peter.brida@feit.uniza.sk, michal.hraska@fri.uniza.sk, tatarka4@stud.uniza.sk, oleksandra.yeremenko@nure.ua

***Abstract*—Currently, network requirements are placed on the efficiency and size of the networks. These conditions can be ensured by modern converged networks that integrate the functions of both data and telecommunication networks. Line or router failures have always been a part of transmission networks, which is no different from converged networks. As a result of outages, which can take from ms to tens of seconds, packets are lost. These outages cause degraded transmission quality, which is undesirable when transmitting real-time multimedia services (Voice over IP, video). To solve the mentioned problems, the IETF organization has developed IP Fast Reroute mechanisms to minimise the time to restore the connection after a line or node failure and, consequently, less packet loss.**

The article reviews and compares the latest IP Fast Reroute mechanisms deployed in the last three years. First, we have Optimistic Fast Rerouting, which calculates optimistic and fallback scenarios. The second is Post-processing Fast Reroute, which decomposes the network according to metrics such as load and route length. Third, Local Fast Reroute focused on low congestion and random access.

***Index Terms*—Fast Reroute; OPFRR; PSFRR; LFRR; MPLS; LSP; OSPF; Post-processing; Random access.**

I. Introduction

From the early days of computer networks to the present day, it has always been important for the network to be reliable. This demand for reliability has increased with the arrival of real-time applications (Voice over IP, Real-time video) and Big Data [1], so networks must be protected against outages. Situations where links or routers fail are common in practise (power failure, congestion, and natural disasters). For these reasons, fast network recovery is currently the subject of research.

The duration of the network convergence (restoration) process, which occurs after a link or router failure, is unpredictable and depends on various circumstances, e.g., the size of the network, the number of nodes, and the type of routing protocol. In the event of an outage, the network converges, and the routers update their routing tables. The convergence process can take seconds, even tens of seconds, and during this time, data are lost and services are unavailable, which brings with it a problem in ensuring the quality of the transmitted service. To mitigate the consequences of outages, Fast Reroute (FRR) mechanisms have been developed and integrated into the network, ensuring the fast restoration of the connection and the re-access of the provided service. General management is critical for temporal data processing, which must ensure time limits for delivery [2], [3].

This document is structured as follows. Section II presents the basic principle of the FRR. Section III provides an overview of existing FRR solutions and their subsections describe the latest FRR mechanisms - OPFRR, PSFRR, and LFRR. Section IV contains a discussion and comparison of the individual mechanisms. Section V presents the conclusions of this article and defines plans for future research.

II. Principle of FRR

The FRR principle is implemented in various FRR mechanisms. These FRR mechanisms differ in the calculation of the alternative route [4]. FRR mechanisms provide an alternative path in cases where a given line or router fails. FRR uses for fast failure detection - the Bidirectional Forwarding Detection (BFD) [5] protocol to detect failures from 30 ms. The BFD protocol uses fast "Hello messages" with an interval of 10 ms, and after three lost "Hello messages", BFD declares the link down - and from this point on, the FRR rerouting process starts. FRR

Manuscript received 8 December, 2022; accepted 26 February, 2023.

This publication was carried out with the support of the Operational Programme Integrated Infrastructure in the frame of the project ("Intelligent systems for UAV real-time operation and data processing", code ITMS2014+: 313011V422) and co-financed by the European Regional Development Fund. The work has been supported by the Slovak VEGA grant agency ("Research of a location-aware system for achievement of QoE in 5G and B5G networks", Project No. 1/0588/22).

reroutes packets from the shortest but broken path to a longer precalculated alternative path (see Fig. 1).

FRR defines the basic terminology of routers (see Fig. 1) [4], [6]:

– *Router S* (source router) - the source router that detects a link or connection failure;

– *Router N* - an alternative precalculated next-hop;

– *Router E* - destination router of alternative FRR path;

– *Router R* - this router does not participate in FRR repair but is part of the network;

– *Router D* - the destination router of the original flow, also called "Provider Edge router" (PE router).

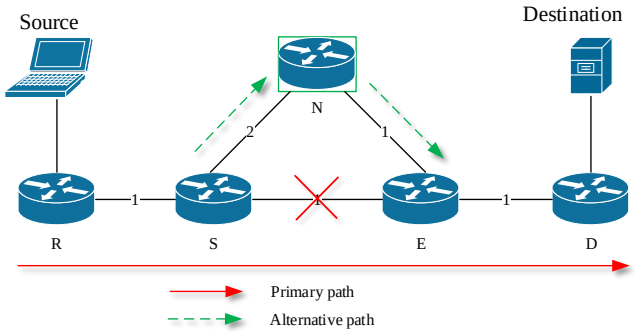

Fig. 1. Principle of Fast Reroute.

After failure detection via the BFD protocol, the FRR mechanisms route the data through an alternate precalculated path, and the network convergence process starts. Therefore, there is no packet loss, and the network continues to operate on the alternate backup path while the primary path is restored. After restoring the primary path, the packets are routed through the original primary path [7].

Today's networks consist of many devices and lines, in which node or connection failures often occur, and many times it is multiple outages. As the authors point out in [8], there are an average of 40 connection failures per day in the data centre. Fixing the faulty connection or device by the administrator himself/herself will take more than a day [8].

Therefore, the issue of fast network recovery/FRR is relevant, and many researchers are currently working on it.

III. Analysis of Existing FRR Solutions

In this chapter, we analyse the latest solutions in IP Fast Reroute in the recent period.

There are different FRR mechanisms in classic IP networks. These mechanisms differ in the way alternative routes are calculated. One of the most used IPFRR mechanisms is Loop-Free Alternates (LFA) [9], Remote LFA (R-LFA) [10], and Equal-Cost Multi-Path (ECMP) [11]. The IP Fast Reroute mechanisms, implemented in routers (LFA, R-LFA), are one of the most used protection mechanisms against outages.

Other experimental IPFRR mechanisms include Multiple Routing Configurations (MRC) [12], [13] and Not-Via Addresses [14], [15]. Furthermore, mechanisms based on tunnelling, Maximally Redundant Trees (MRT) [16], [17], and other IPFRR mechanisms based on alternative trees [18], [19].

The following mechanisms are the subject of recent scientific papers: Optimistic Fast Rerouting (OPFRR) [20], Post-Processing Fast Reroute (PSFRR) [21], Local FRR with low congestion and random access [22]. These three mechanisms are the subject of our further investigation.

A. Optimistic Fast Rerouting (OPFRR)

The idea of OPFRR [20] is to optimise the backup path in the best possible way, which is supposed to ensure connection after the failure of the original path.

Existing FRR solutions have sacrificed the quality of the backup path at the cost of packet delivery. OPFRR uses part of the mechanism of the so-called "EGR" [19], named after the first letters of the names of the authors of the mechanism. OPFRR prioritises the quality of backup routes. If it is impossible to find the best route in the first (optimistic) mode, the second (fallback) mode is used (see Fig. 2). Simulations have shown that the advantage of the OPFRR protocol increases with increasing network complexity.

Since the algorithm has the optimistic mode and default mode, the transition to the fallback mode ensures the so-called "watchdog". This watchdog monitors the quality of the backup route. In case the watchdog identifies an insufficient number of good quality backup paths found by optimistic mode, OPFRR then switches to fallback mode. The watchdog must carefully and with proper timing monitor backup paths. If the watchdog starts the fallback mode too soon, the quality paths provided by the optimistic mode might be lost. If the watchdog is started late, the backup path may be too long; thus, the delay will increase, and the path speed will decrease.

Advantages of OPFRR against the classic FRR mechanisms:

– Backup path optimisation and packet delivery are handled separately;

– The quality of the backup path in optimistic mode should be high;

– Various existing FRR algorithms can be used in the backup mode and design the optimistic mode accordingly.

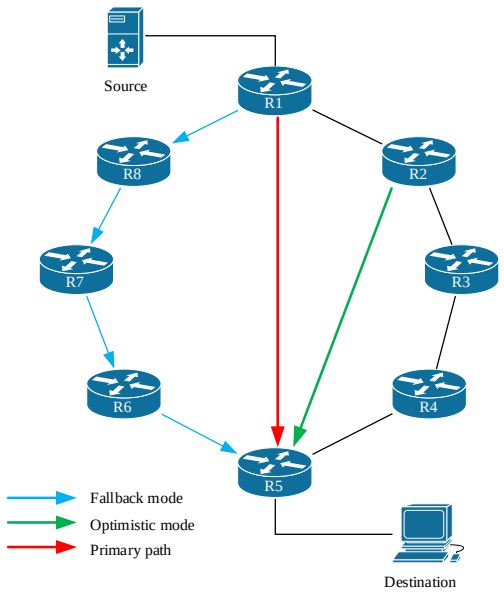

Fig. 2. Principle of OPFRR.

OPFRR comes with a solution in the form of two backup path-switching modes.

First mode - optimistic: When a packet encounters a failed connection, it is forwarded by default in this first optimistic mode. This optimistic mode optimises quality of backup paths in case of failure, i.e., *the shortest route with the lowest delay* is chosen.

If it is impossible to find a high-quality backup route, OPFRR switches to the second mode, fallback.

The second mode - fallback: It uses the EGR protocol to calculate alternative paths and guarantee packet delivery. The authors of OPFRR claim that it is also possible to use any routing algorithm that ensures the delivery of packets in this fallback mode.

1. *Example and motivation for using OPFRR*

Let's have the network with 100 nodes (see Fig. 3(a)) with nine broken connections (see Fig. 3(b)). There is a backup route with 142 connections obtained by the EGR mechanism (see Fig. 3(c)) [19]. At the same time, another backup route takes only two jumps to reach the destination (see Fig. 3(d)). EGR makes it a priority to deliver the packet at any cost, and the main problem EGR solves is delivering the packet. Therefore, it overlooks existing shorter packet delivery paths because it does not consider the quality of the path. The OPFRR algorithm will use a 2-hop path (see Fig. 3(d)) and significantly reduce the delay to the EGR.

In this case, the optimistic OPFRR mode was used, which prioritised the quality of the backup paths over the guarantee of packet delivery.

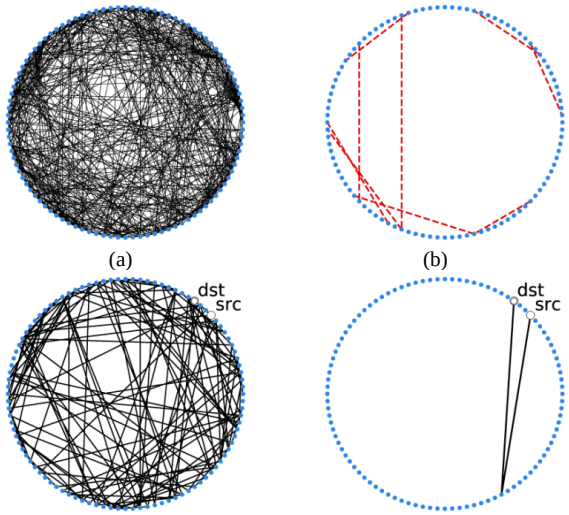

Fig. 3. A motivating example for OPFRR [20]: (a) Original network topology; (b) Failed links; (c) The failover route obtained by EGR; (d) Alternative failover route.

2. *Routing algorithms used by OPFRR*

The fallback mode uses the EGR and Basic Routing (BSC) algorithms [23] to calculate alternative paths [20]. These algorithms are tree-based, can tolerate multiple failures, and create backup paths when primary paths fail. EGR and BSC are used separately as routing algorithms. Two different realisations of the optimistic backup framework arise.

3. *Switching between modes - Watchdog mechanism*

The Watchdog mechanism determines the rules for switching between the first and second modes. The watchdog checks that packets between nodes do not remain in a routing loop in optimistic mode. A packet cannot visit the same node twice. Otherwise, it gets stuck in a loop. Therefore, each visited node must be stored in the packet header. The disadvantage is that it causes an additional burden on computing power. If the packet encounters a node, it was already in, the watchdog switches to fallback mode. Watchdog was tested first by starting the backup mode if the length of the backup route exceeded a specific maximum limit. This was inefficient because packets could get stuck in loops, increasing the length of the backup path. If a low threshold was set, the watchdog switched to fallback mode early.

B. *Post-Processing Fast Reroute (PSFRR)*

Most of the current Fast Reroute mechanisms use static mechanisms for fast rerouting of the packet in case of failure of the primary path. However, failures of several nodes or links are still challenging to solve, especially in algorithmisation. The motivation behind PSFRR is to solve the problem of how to optimise failover rules in different directions for all possible failures. It is combinatorics, where different problems can occur in different directions. The latest FRR solutions try to use network decompositions based on arc disjoint trees.

Backup paths, with disjoint arc trees, consistently deliver packets to the destination, even with multiple simultaneous outages. However, the disadvantage is that these paths are not always the shortest and can cause a load on the network. The effort of this algorithm is to use post-processing in favour of FRR decomposition (higher quality of backup routes, reduction of route length, and network load). If a failure occurs, it is necessary to route the packets along the shortest possible path so as not to cause additional load and high delay.

In Fig. 4, there are two t-rooted arc-disjoint spanning arborescence T1 (Fig. 4(a)) and T2 (Fig. 4(b)). In both graphs, one arborescence is drawn with red dotted arrows, and the second arborescence is illustrated with dashed blue arrows.

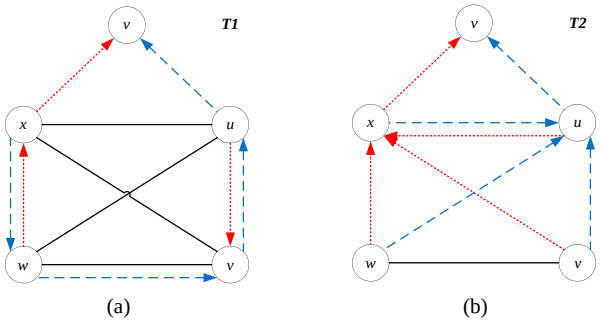

Fig. 4. Network with two different t-rooted arc-disjoint spanning arborescence decompositions [21]: (a) T1; (b) T2.

PSFRR does not allow packet marking or the inclusion of failure information in the packet header. The basis of this algorithm is formed from arc-disjoint arborescent networks, i.e., each destination has its own destination tree structure through which it can send a packet. If the primary path fails, the packet is sent to the next tree structure by a pre-defined

rule. The logic is defined by the tree routing strategy.

Routing is tree-based, and packets are routed through the tree structure to the root of the tree. A packet moves through an arbitrary tree structure if no failure occurs. If the primary path to the root fails, the packet starts to be routed through a different tree structure.

Theorems and individual algorithms are described in the work of the authors in the manuscript in [21]. In Fig. 5, there are 4 t-rooted arc-disjoint arborescences (blue, red, green, and olive).

Three links have failed to connect to node 22, causing reroute through olive arborescence at 22.

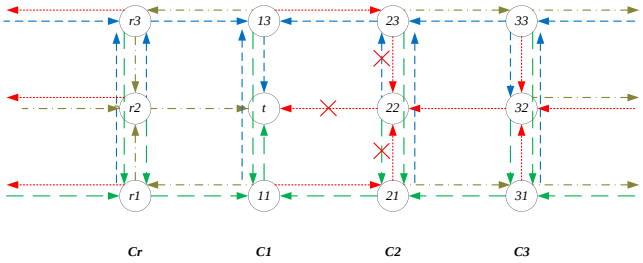

Fig. 5. Example of 4 t-rooted calculated arc-disjoint arborescences.

C. *Local FRR with Low Congestion and Random Access (LFRR)*

FRR backup paths must also resist additional, local, unknown failures in the flow direction. Local FRR mechanisms provide a high degree of resistance against multiple link failures. They ensure low congestion on backup paths. This method uses a random approach and is adapted to highly interconnected networks. Highly meshed networks provide very good resilience against unpredicted link or node failures. Modern technologies, such as 5G networks, have high latency and reliability requirements. This protection mechanism targets scenarios where multiple outages occur simultaneously. In addition to the standard options, it guarantees perfect protection that is resistant to malfunction.

The motivation is also to maintain connectivity between nodes and low load, even under the assumption of a large number of failures.

The main benefits of local FRR include three randomised fast forwarding algorithms. They guarantee high resistance to failure of several lines. Another advantage is lower overhead than any possible deterministic algorithm.

The first approach guarantees (see Fig. 6) [22], under the assumption $\varphi = O(n)$ edge (links) failures, that a load ($O(\log n \log\log n)$) is not exceeded in most nodes while the remaining $O(\text{polylog } n)$ nodes achieve a load of at most $O(\text{polylog } n)$.

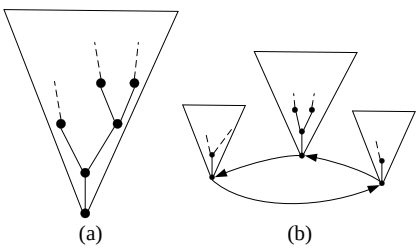

Fig. 6. The structures contained in the subgraph: (a) A tree rooted; (b) A cycle, and each node of the cycle is a root of a tree.

The second algorithm of the authors decreases the resilience to edge failure to $O(n/\log n)$. This algorithm is purely based on destination and has only $O(\log n \log\log n)$ congestion at *any* node [22].

The third algorithm assumes that the nodes have access to polylog n bits of shared information, and the load on the node can be reduced - a maximum load of only $O\left(\sqrt{\log n}\right)$ occurs at *any* node (router) [22].

All three algorithms guarantee loop-free operation and prevent packet reordering.

Local FRRs avoid time-consuming recalculation and failure information collection. Using a random approach, it is possible to reduce the overload from polynomial to polylogic arithmetic with high probability, thus breaking the lower bounds of deterministic overload. Network links are affected by multiple simultaneous failures identified by an adversary. The goal is to pre-define local backup path rules for different nodes to redirect traffic to its destination while balancing the network load. A failover rule is a match action forwarding rule that matches specific header fields of incoming packets (such as the destination IP address). The communication network is modelled as a complete undirected graph, where nodes are routers or switches that need to be configured using static routing rules, and links can fail [22].

IV. Discussion

This paper explored the latest FRR solutions, OPFRR, PSFRR, and Local FRR with low congestion and random access. These solutions have a high computational complexity to calculate alternative paths. Existing FRR mechanisms such as LFA and RLFA are implemented in Cisco and Juniper operating systems. They are implemented due to their low computational complexity and load on the router Central Processing Unit (CPU). The solutions described in this paper are excellent in quality and length of calculating alternative paths, but very complex compared to LFA and RLFA (see Table I).

In recent years, we have published some scientific papers dealing with the analysis of existing FRR solutions in many areas [7], [24], [25], [26]. Based on these analyses, we summarise the essential information in Table I.

In [20], the authors proposed an OPFRR fast forward framework that consists of two modes, an optimistic mode and an emergency mode. Through simulations, they proved that the proposed framework could shorten the failover routes and reduce the delay between endpoints. With their solution, they have improved their previous approach.

OPFRR is trying to shorten the length of backup paths with an optimistic mode. Due to this, these paths are shorter, reducing the response. If the paths were long, there would be network congestion, reducing the performance of functional network paths. If it is impossible to find the optimal paths, OPFRR switches to the fallback mode, the objective of which is to deliver the packet even at the cost of increasing the response and reducing the quality of other network flows [20].

Post-processing improves the arbitrary decomposition of the network in terms of basic traffic engineering metrics

such as load and route length.

TABLE I. COMPARISON OF EXISTING FRR SOLUTIONS.

Existing solution	100 % repair coverage	Pre-computing	Packet modification	Link-state dependency	Compu-tational complexity
ECMP FRR	No	Yes	No	No	Low
Directed LFA	Yes	Yes	Yes	Yes	High
LFA	No	Yes	No	No	Low
MPLS-TE FRR	No	Yes	Yes	No	High
MRC	Yes	Yes	Yes	Yes	High
MRT	Yes	Yes	Yes	Yes	High
Not-Via Addresses	Yes	Yes	Yes	Yes	High
Remote LFA	No	Yes	Yes	Yes	Average
TI-LFA	Yes	Yes	Yes	Yes	High
OPFRR	Yes	Yes	No	Yes	High
PSFRR	Yes	Yes	No	Yes	High
LFRR	Yes	Yes	No	Yes	High

Note: *MPLS Traffic Engineering (TE) Fast Reroute (FRR); Topology Independent LFA (TI-LFA).*

This framework is also suitable for optimising many destinations. In addition, the framework can also be used to increase the resilience of Shared Link Risk Groups (SRLG), which is an essential extension in practise [21].

In [22], are three LFRR random access algorithms, the main benefit of which is high resistance to multiple link or node failures, but also an exponentially lower load compared to the deterministic algorithm.

All three analysed solutions have a similar property and have a high computational complexity. From a higher point of view, it is very hard to develop an FRR mechanism that has 100 % repair coverage and low computational complexity. Therefore, if mechanisms should provide 100 % repair coverage, they will also have a high computational load on CPUs.

A critical factor in fast network recovery technology is the rapid detection of an outage and subsequent reporting to the remaining routers affected and disrupted by that outage. In FRR mechanisms, this information is sent out as follows:

– By modifying special bits in the IPv4 header;

– By encapsulating the packet with another header;

– Depending on the interface on which the packet was received.

Note that packet modification can cause various compatibility problems and can also affect the Maximum Transmission Unit (MTU) on some network links.

Another important fact is that several FRR mechanisms require topological information about the network from a database of link-state routing protocols to compute an alternative path. This property limits the applicability of FRR mechanisms to networks that use only primary link-state routing protocols. Most existing FRR mechanisms currently depend on information from link-state routing protocols.

The analysed FRR mechanisms do not mention the use of the IPv6 protocol. This topic requires increased attention, as these mechanisms do not include compatibility and leveraging the benefits of IPv6.

IPv6 provides a significantly different packet header and more possibilities, such as extension headers. These headers make IPv6 more attractive for the IPFRR domain, as it provides new use effective solutions in FRR.

V. Conclusions

This paper provides a technical review of the latest FRR solutions (OPFRR, PSFRR, and Local FRR). We present basic principles of analysed solutions and compare them with each other and also with other existing FRR mechanisms.

This analysis is very important for our further investigation.

OPFRR solution consist of two algorithms - optimistic and fallback mode. The optimistic mode finds the best alternative path with the best delay parameter. If it is not possible to find such a high-quality route, fallback mode finds any possible route. PSFRR decomposes network diagrams in terms of basic traffic engineering metrics such as load and route length to calculate the best possible alternative path.

LFRR presents three algorithms to calculate alternative paths. Each algorithm has its own conditions for calculating alternative FRR paths. These algorithms give exponentially lower load compared to the deterministic algorithm.

Since in the past, we have dedicated ourselves to improving the original FRR mechanism proposed in [27], we are currently preparing a new FRR mechanism for the Wireless Sensor Networks (WSN) area. Moreover, we want to simulate analysed FRR mechanisms (OPFRR, PSFRR, and Local FRR) for scientific data.

As there are several different networks in use today, and they are still growing, despite the continuous progress in the field of FRR, there is no ideal solution in the case of multiple outages. This is why this research area is still highly relevant and interesting to researchers.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] G. Koman, M. Kubina, M. Holubčík, and J. Soviar, "Possibilities of application a big data in the company innovation process", *Knowledge Management in Organizations. KMO 2018. Communications in Computer and Information Science*, vol 877. Springer, Cham, 2018, pp. 646–657. DOI: 10.1007/978-3-319-95204-8_54.

[2] M. Kvet and M. Kvet, "Relational pre-indexing layer supervised by the DB-index-consolidator background process", in *Proc. of 2021 28th Conference of Open Innovations Association (FRUCT)*, 2021, pp. 222–229. DOI: 10.23919/FRUCT50888.2021.9347573.

[3] M. Kvet, "Autonomous temporal time zone management", in *Proc. of ECON 2021 - 47th Annual Conference of the IEEE Industrial Electronics Society*, 2021, pp. 1–6. DOI: 10.1109/IECON48115.2021.9589547.

[4] M. Shand and S. Bryant, "IP Fast Reroute Framework", RFC 5714, 2010. DOI: 10.17487/rfc5714.

[5] D. Katz, D. Ward, S. Pallagatti, and G. Mirsky, "Bidirectional Forwarding Detection (BFD) for Multipoint Networks", RFC Editor, 2019. DOI: 10.17487/RFC8562.

[6] A. Kamisinski, "Evolution of IP Fast-Reroute strategies", in *Proc. of 2018 10th International Workshop on Resilient Networks Design and Modeling (RNDM 2018)*, 2018, pp. 1–6. DOI: 10.1109/RNDM.2018.8489832.

[7] J. Papan, P. Segeč, M. Moravčík, M. Konštek, L. Mikuš, and J. Uramova, "Overview of IP Fast Reroute solutions", in *Proc. of 2018 16th International Conference on Emerging eLearning Technologies and Applications (ICETA)*, 2018, pp. 417–424. DOI: 10.1109/ICETA.2018.8572205.

[8] P. Gill, N. Jain, and N. Nagappan, "Understanding network failures in data centers: Measurement, analysis, and implications", *Proc. of the ACM SIGCOMM 2011 Conference (SIGCOMM'11)*, 2011, pp. 350–361. DOI: 10.1145/2018436.2018477.

[9] P. Sarkar, U. Chunduri, S. Hegde, J. Tantsura, and H. Gredler, "Selection of Loop-Free Alternates for multi-homed prefixes", RFC Editor, 2019. [Online]. Available: https://www.rfc-editor.org/info/rfc8518.

[10] J. Abhishek Singh, K. M. R. Sachin, and K. S. Shushrutha, "Implementation of topology independent Loop Free Alternate with segment routing traffic", in *Proc. of 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2021, pp. 1–5. DOI: 10.1109/ICCCNT51525.2021.9579530.

[11] L. Csikor and G. Rétvári, "On providing fast protection with remote loop-free alternates", *Telecommun. Syst.*, vol. 60, pp. 485–502, 2015. DOI: 10.1007/s11235-015-0006-9.

[12] S. Cevher, M. Ulutas, S. Altun, and I. Hokelek, "Multiple routing configurations for Fast Re-route in software defined networks", in *Proc. of 2016 24th Signal Processing and Communication Application Conference (SIU)*, 2016, pp. 993–996. DOI: 10.1109/SIU.2016.7495909.

[13] E. Okada, K. Hirata, and T. Tachibana, "Design of a load balancing method for multiple routing configurations", in *Proc. of 2022 IEEE International Conference on Consumer Electronics - Taiwan*, 2022, pp. 215–216. DOI: 10.1109/ICCE-TAIWAN55306.2022.9869133.

[14] M. Nagy, J. Tapolcai, and G. Rétvári, "Optimization methods for improving IP-level fast protection for local shared risk groups with Loop-Free Alternates", *Telecommun. Syst.*, vol. 56, pp. 103–119, 2014. DOI: 10.1007/s11235-013-9822-y.

[15] M. Menth, M. Hartmann, R. Martin, T. Čičić, and A. Kvalbein, "Loop-free alternates and not-via addresses: A proper combination for IP fast reroute?", *Computer Networks*, vol. 54, no. 8, pp. 1300–1315, 2010. DOI: 10.1016/j.comnet.2009.10.020.

[16] K. Kuang, S. Wang, and X. Wang, "Discussion on the combination of Loop-Free Alternates and Maximally Redundant Trees for IP networks Fast Reroute", in *Proc. of 2014 IEEE International Conference on Communications (ICC)*, 2014, pp. 1131–1136. DOI: 10.1109/ICC.2014.6883473.

[17] A. Atlas, C. Bowers, and G. Enyedi, "An architecture for IP/LDP Fast Reroute Using Maximally Redundant Trees (MRT-FRR)", RFC 7812, 2016.

[18] O. Lemeshko, A. Kinan, and M. A. jabbar A. Wahhab, "Multicast fast re-route schemes for multiflow case", *The Experience of Designing and Application of CAD Systems in Microelectronics*, 2015, pp. 422–424. DOI: 10.1109/CADSM.2015.7230892.

[19] T. Elhourani, A. Gopalan, and S. Ramasubramanian, "IP fast rerouting for multi-link failures", *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 3014–3025, 2016. DOI: 10.1109/TNET.2016.2516442.

[20] H.-K. Tan and T.-W. Kuo, "Optimistic fast rerouting", in *Proc. of IEEE International Conference on Communications*, 2022, pp. 1692–1697. DOI: 10.1109/ICC45855.2022.9838260.

[21] K.-T. Foerster, A. Kamisinski, Y.-A. Pignolet, S. Schmid, and G. Tredan, "Improved Fast Rerouting using postprocessing", *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 1, pp. 537–550, 2022. DOI: 10.1109/TDSC.2020.2998019.

[22] G. Bankhamer, R. Elsasser, and S. Schmid, "Local fast rerouting with low congestion: A randomized approach", *IEEE/ACM Transactions on Networking*, vol. 30, no. 6, pp. 2403–2418, 2022. DOI: 10.1109/TNET.2022.3174731.

[23] M. Chiesa *et al.*, "On the resiliency of static forwarding tables", *IEEE/ACM Transactions on Networking*, vol. 25, no. 2, pp. 1133–1146, 2017. DOI: 10.1109/TNET.2016.2619398.

[24] J. Papan, P. Segec, and P. Paluch, "Analysis of existing IP Fast Reroute mechanisms", in *Proc. of 2015 International Conference on Information and Digital Technologies*, 2015, pp. 291–297. DOI: 10.1109/DT.2015.7222986.

[25] J. Papan, P. Segec, M. Drozdova, L. Mikus, M. Moravcik, and J. Hrabovsky, "The IPFRR mechanism inspired by BIER algorithm", in *Proc. of 2016 International Conference on Emerging eLearning Technologies and Applications (ICETA)*, 2016, pp. 257–262. DOI: 10.1109/ICETA.2016.7802053.

[26] J. Papán, P. Segeč, P. Palúch, Ľ. Mikuš, and M. Moravčík, "The survey of current IPFRR mechanisms", *Proceedings of the 2015 Federated Conference on Software Development and Object Technologies. SDOT 2015. Advances in Intelligent Systems and Computing*, vol 511. Springer, Cham, 2017, pp. 229–240. DOI: 10.1007/978-3-319-46535-7_18.

[27] J. Papan, P. Segec, O. Yeremenko, I. Bridova, and M. Hodon, "Enhanced Multicast Repair Fast ReRoute Mechanism for smart sensors IoT and network infrastructure", *Sensors*, vol. 20, no. 12, p. 3428, 2020. DOI: 10.3390/s20123428.