A New Traffic Distribution Routing Algorithm for Low Level VPNs

Abdelwahed Berguiga[1], Ahlem Harchay[2], Ayman Massaoudi[3], Radhia Khdhir[4]
Department of Computer Science
Jouf University, Sakakah
Saudi Arabia

***Abstract*—Virtual Private Networks (VPN) constitute a particular class of shared networks. In such networks, the resources are shared among several customers. The management of these resources requires a high level of automation to obtain the dynamics necessary for the well-functioning of a VPN. In this paper, we consider the problem of a network operator who owns the physical infrastructure and who wishes to deliver VPN service to his customers. These customers may be Internet Service providers, large corporations and enterprises. We propose a new routing approach referred to as Traffic Split Routing (TSR) which splits the traffic as fairly as possible between the network links. We show that TSR outperforms Shortest Path Routing (SPR) in terms of the number of admitted VPN and in terms of Quality of Service.**

***Keywords*—Virtual Private Networks (VPN); Quality of Service (QoS); NS-2; Simulations; Shortest Path Routing (SPR); Traffic Split Routing (TSR); Routing algorithm**

I. INTRODUCTION

With the exponential growth of the Internet and increasingly supports various types of applications, especially those calling on multimedia as well as several users simultaneously, the Internet service provider as well as the network operator are called upon to guarantee commitments of quality of service to their subscribers. The simplicity and low cost of IP networks are some of the reasons why users are deploying new types of applications on these networks. However, some types of real-time applications including video conferencing and VoIP which are very sensitive to variations in delay (jitter) and throughput are not guaranteed with IP.

Reservation of resources for multimedia applications is necessary to ensure end-to-end performance. However, this reservation is not supported with IP. Also, real-time applications also require a guarantee on resources such as storage space, CPU time etc. Thus, the packets must be routed based on the required QoS, which is not possible with the Internet today. However, the Internet is by nature "Best Effort" and lacks any control over the quality of service. Traditional Layer 3 routing methods like Routing Information Protocol (RIP), Open Shortest Path First (OSPF) and Border Gateway Protocol (BGP) would become obsolete if we want to support QoS in the Internet.

A more viable alternative to traditional IP routing and which includes the use of technologies and network infrastructures that guarantee QoS, we can cite IP / MPLS, IP over Metro Ethernet or also IP over ATM. However, even if these technologies support QoS, the topology of the network as well as the routes which will carry the traffic must be correctly chosen otherwise the cost of QoS would be prohibitive. Perhaps the best-known example is the problem of the taxi driver who has to find the quickest and easiest way to get from one place to another. Therefore, instead of letting each driver individually make the decision to choose their route, we instead need to inform them in advance which route they should take. Therefore, the optimal choice of routes must be calculated in advance, i.e. the solution to adopt to guarantee QoS is the "proactive" approach and not the "reactive" approach.

In order to find the optimal routes, we have to define the optimality criteria, i.e. the objectives and the constraints. When a network is given, which is the case with an operator who owns the physical infrastructure, the goal is to increase the number of "satisfied" customers and therefore income. When we begin to build a network, as in the case of a service provider that does not have the infrastructure, the goal is to minimize the use of the leased resources and therefore the cost of the network. In both cases, QoS is a constraint. Moreover, Virtual Private Networks (VPN) constitute a particular class of shared networks. In such networks, the resources are shared among several customers. The management of these resources requires a high level of automation to obtain the dynamics necessary for the well functioning of a VPN.

It is in this context that the present research work “Distributed traffic routing for low-level VPNs” falls within this framework, where we will design in an optimized way VPNs based on logical topologies or multipoint virtual circuits such as such as VPLS (Virtual Private LAN Service), E-LAN (Ethernet LAN Services) etc. We propose to study the case of an operator who has the physical infrastructure and who wants to offer this kind of VPN service to its customers. The operator then seeks to maximize the number of customers while providing the QoS required by each of them.

The rest of the paper is organized as follows: Section II provides an overview on the graph generations using Waxman and Brite algorithms. Section III describes the proposed solution. Section IV reports the performance evaluation environment and the simulation methodology. We report and explain simulation results, useful to assess the validity of our proposed traffic distribution algorithm. Finally, the last section draws conclusions.

II. Related Works

The design of virtual private networks brings together a whole set of optimization problems that differ by the constraints imposed, and sometimes by the data considered. Studying these issues is very important for network operators as well as service providers. The methods of solving these problems often call graph theory, performance analysis and optimization, descriptive as diverse as they are complex.

For a supplier, it is about setting up a network that guarantees the delivery of all its customers' requests with quality of service requirements, while minimizing network operating costs. Operators seek customer satisfaction by trying to make the most of all the resources available in their networks.

A. Random Generation of Graphs

The study of large networks is becoming increasingly important, especially thanks to the evolution of telecommunications networks and the Internet. These networks can be of a different nature. To model them, we often use the formal structure of graphs. A graph is modeled by a set of vertices connected by edges. We can enrich the structure of the graph by assigning a cost to each of the edges. It is often difficult to represent a network accurately. We often prefer to represent the local properties of a network, then we generate the graph while respecting these properties as much as possible. Among other things, the generation of graphs allows us to do simulations.

During this research work, we used a tool that generates random graphs known as “Brite” [1]. Brite allows a random generation of several graphs following the “Waxman” model [2]. The latter offers us the possibility of obtaining large networks whose characteristics resemble those of an Internet network. Once a network is generated, the "Waxman" model assigns two parameters to each link: cost and time.

The principle of the "Waxman" method is as follows [2]:

1) Enter the number of nodes to generate.

2) Calculate the probability P (u, v) of adding a link between each pair of nodes u and v.

$$P(\{u,v\}) = \beta\, e^{\frac{-d(u,v)}{L\alpha}}$$

Where:

$d(u,v)$: the distance between node u and v.

L: the maximum distance between two nodes.

α and β: Two parameters that vary in the interval (0.1];

The increase in β results an increase in the density of links in the graph.

The decrease in α results in an increase in the density of the short links between the nodes.

For each $P(u,v)$, draw a random number T between (0,1]. If T <P, then add a link between u and v.

B. Routing Algorithms

New VPN technologies have greatly expanded the range of possibilities for users. On the one hand, they allow very great flexibility for users, and on the other hand, they lead to increasing complexity for operators and service providers. Several considerations must be examined in order to ensure satisfactory quality of service (QoS) following customer requests. Among these considerations, we can cite:

1) Optimal allocation of communication resources according to user needs and available resources in the network.

2) The establishment of reliability control mechanisms.

On the other hand, the quality of service offered to a connection is directly related to the choice of the path between a source and a destination. The route calculation must take into account the various constraints imposed by a connection (speed, variation in delay, loss rate, etc.). In this outcome, it is necessary to set up a routing algorithm whose role is to find the best path between a source and its recipient while respecting the various constraints imposed. We speak of routing with constraints. Because these constraints vary from customer to customer, and from one type of network to another, it is almost impossible to find a routing algorithm that meets all needs. Indeed, it was proved in [3], that the problem of finding a path with multiple constraints is NP-Complete.

Several heuristic proposals were then presented to solve this problem. These proposals can be classified into five categories:

1) The first approach is to minimize a single QoS parameter. The algorithms of Dijkstra and Bellman-Ford are examples of this approach. They find the shortest route between a source and its destination.

2) The second approach is presented by [4]. An algorithm based on the minimization of a QoS parameter subject to a second constraint is proposed. It uses the cost and the delay calculated by a “distance vector” protocol maintained at each node.

3) The third approach is to build a path under two constraints simultaneously (usually time and cost). Chen et al. [5] and Jaff et al. [6] have proposed algorithms to solve the problem with two constraints. The major problem with this proposal is that it is more complex than the other heuristics and it does not guarantee scalability.

4) The fourth approach is based on minimizing the different parameters in a specific order. The Widest-Shortest and Shortest-Widest [7] algorithms are examples of this approach.

5) The fifth approach to the routing problem with QoS is to construct paths using a combined metric that is calculated based on two (or more) constraints. Verma et al. [8] combined cost and bandwidth into a single metric.

The routing approaches with QoS presented previously vary from the simplest, such as Dijkstra and Bellman-Ford, which are based on a single constraint, to the more complex exploiting two or more constraints. However, these approaches have a common weakness in that they do not

guarantee the formation of a balanced system when distributing the load. They use an order of priority in the choice of constraints which leads to the construction of unbalanced paths.

In this research work, we consider the problem of a network operator who owns the physical infrastructure and who wishes to deliver VPN service to his customers. These customers may be Internet Service providers, large corporations and enterprises. We propose a new routing approach referred to as Traffic Split Routing (TSR) which splits the traffic as fairly as possible between the network links. We show that TSR outperforms Shortest Path Routing (SPR) in terms of the number of admitted VPN and in terms of Quality of Service.

III. Proposed Algorithm

In what follows, we will present a simple algorithm, Traffic Split Routing (TSR) [9], having as main objective the load sharing in a network. Indeed, with the use of TSR we will try to distribute the traffic in the network as homogeneously as possible. Our approach is to be able to use the network for a balanced sharing of traffic [10], [11]. Our main goal is to avoid overloading some links while others remain unused. This is often achieved by creating disjointed trees and/or paths and small sizes.

We present in what follows the heuristic of traffic distribution used:

Traffic distribution heuristic

Given ls the number of times a link s appears in a VPN tree.

1: Initialize ls ← 0 for all links.

2: Initialize n ← 0 and wait for a new VPN connection request (or a new site to add to an existing VPN).

3: Generate (or complete) a "generic" tree (path) linking all the new VPN sites without using the links whose ls> n.

4: If the tree is not completed, increment n ← n + 1, and go back to step 3.

Otherwise (the tree is complete), increment ls ← ls + 1 for all the links of the new generated tree and go back to step 2.

First, we define a variable called Link-Usage Count [LUC] (LUC refers to the variable "ls" in the previous algorithm) which gives the number of times a link appears in a VPN tree. This variable will be used as a metric for the generation of trees. In fact, every time a link is used in a tree, its LUC [12] is incremented by one. When generating new trees, our algorithm will try to avoid links with the highest LUC.

Obviously, when a VPN connection ends or one of the sites disconnects, the "ls" value of each link belonging to that connection is decremented by one. The generic tree generated in step 3 can be obtained with any algorithm or protocol. For example, we can use the minimum weight tree (MST) [13]. Since this tree is determined according to the value of the variable “ls”, we must verify that the number of jumps in this tree should not be arbitrarily long [11].

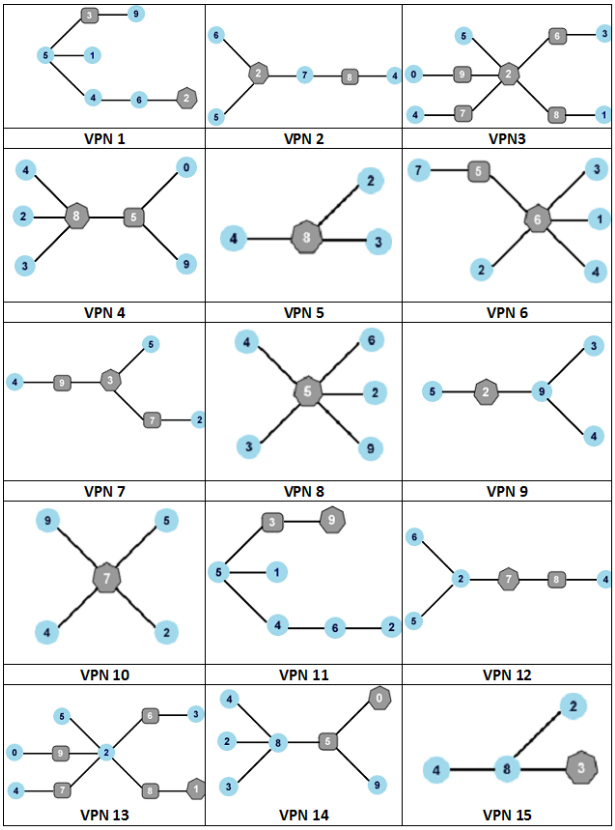

Fig. 1. Example of VPNs Generated by the TSR Algorithm.

IV. Simulations and Performance Analysis

This section presents the simulation input parameters for the different simulated VPN networks. All the simulation parameters are given in Table I. For accuracy and compliance, we ran each simulation scenario six times and averaged the measurements. Note that each of the six measurements conforms to the simulation parameters already described. To study the behavior of the two routing algorithms SPR and TSR according to the traffic intensity in the network, we varied the number of VPNs to be simulated. Fig. 1 gives an example of a VPN network that we have simulated. Each VPN is made up of a source and a set of destinations. Nodes in the form of a hexagon represent the sources. The nodes in the form of a circle represent the destinations. Rectangular nodes schematize transit nodes (Steiner) used to reach stations belonging to the same VPN.

We assume that data streams are sent from one source to a destination within the same VPN [14]-[21]. By applying the two heuristics SPR and TSR important differences in traffic distribution are remarkable. Both Fig. 2 and Fig. 3 represent an example of a scenario to be simulated where we have fixed the source and the destination while applying the two routing algorithms already described.

Fig. 2 schematizes a scenario where we have called the shortest path algorithm. By analyzing this scenario, we can see that the traffic going from source node 2 to destination node 4 is always focused on the same path, the shortest path (2-6 and 6-4).

TABLE I. SIMULATION PARAMETERS

Parameters	Values
Number of nodes	10
Link capacity	10 Mb/s
Delay transmission	10 ms
Number of source VPN	From 4 to 24 source nodes
Maximal size of window congestion	32
Simulation Time	400 seconds
Application type	FTP
Packet size	1000 bytes

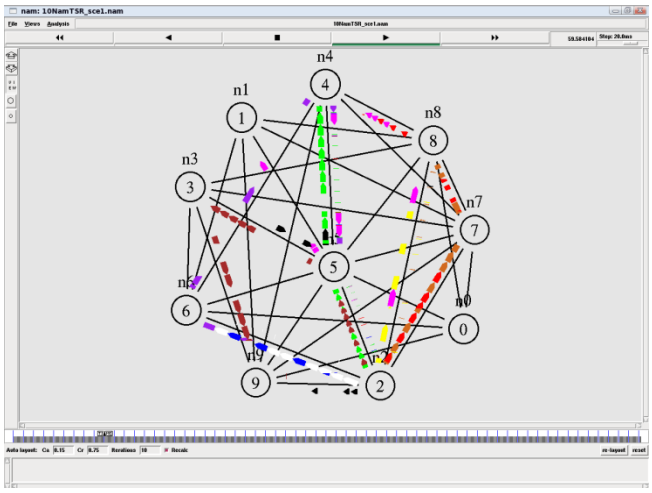

Fig. 2. SPR Traffic.

On the other hand, using the TSR heuristic, in Fig. 3, we notice that the traffic is shared in a more equitable way in the network. In fact, from source node 2 we can reach destination 4 by taking different paths (2-6 and 6-4 or even 2-5 and 5-4, etc…). This approach allows the maximum use of the network links. In order to be able to compare the two heuristics SPR and TSR in a more rigorous way, we will calculate the quality of service parameters: the average reception rate, the delay, the loss rate and the data flow sent.

Indeed, the routing technique will have a great influence on these parameters. This influence will be presented and highlighted later by the simulation results which concern the cases of 4 to 24 VPN sources representing respectively low and high traffic intensities.

A. Average Reception Data Rate

Fig. 4 details the flow variation for the two heuristics TSR and SPR. It clearly illustrates the speed changes depending on the number of VPN sources. Indeed, with 4 VPN sources, the heuristic SPR offers a throughput of 6.56Mbps while with TSR the throughput is 6.16Mbps. We can thus conclude that under a low traffic intensity, the application of the shortest path algorithm for traffic routing is more efficient than the application of traffic distribution.

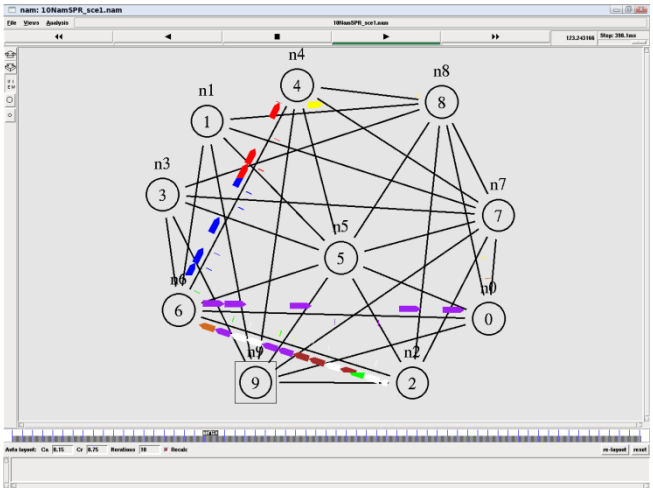

Fig. 3. TSR Traffic.

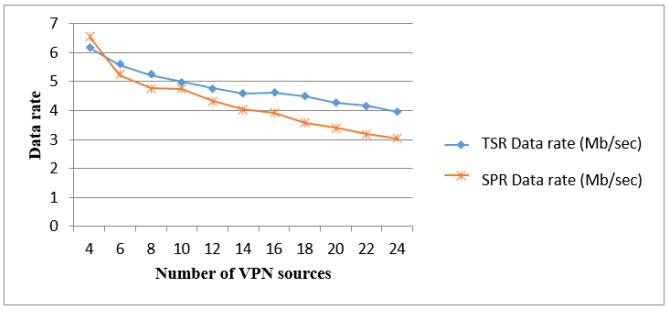

Fig. 4. Average Data Rate Reception.

Subsequently, with a high number of VPNs, the throughput with the SPR heuristic was significantly reduced to 4.74Mbps with 10 VPN traffic sources and 3Mbps with 24 VPN traffic sources. This decrease in throughput is due to the amount of traffic that is focused precisely on the shortest path.

On the other hand, with the TSR heuristic, we can notice that for 10 VPN sources the throughput is 5Mbps and for 24 VPN sources it can reach a value of 3.95Mbps. By comparing these results with the previous ones we deduce that the TSR algorithm offers a higher throughput especially for a large volume of traffic.

The analysis of the average reception rate allowed us to deduce that the TSR algorithm tends to use the maximum number of links, unlike the shortest path algorithm where the traffic always takes the shortest path which, in steady state, causes some links to become overloaded, leaving others unused. This had an influence on the flow.

Moreover, for a given throughput, the number of VPN sources admissible by the TSR method is significantly higher than that obtained with SPR. For example, if we want a speed of 4Mbps, with TSR we can admit up to 24 VPN sources while with SPR, this number is 14 sources.

B. Average End-To-End Delay

In this part we measure the average time taken for a 1000byte size packet to be transferred from a source to a destination. Fig. 5 shows the average delay for the two routing techniques used. This delay is given according to the number of VPN sources. Network reactions to the increased number of VPN sources for the two routing techniques are diverse. In fact, the average delay for low traffic intensity calculated with the shortest path algorithm exhibits a brief variation compared to that determined by the TSR algorithm. Indeed, for 4 VPN sources the average delay determined by the SPR approach is 24ms while with TSR this delay is 25ms.

By increasing the number of VPNs we can notice changes in the shape of the two curves. In fact, the average delay increases as a function of the number of VPN sources in an almost logarithmic fashion. However, these two curves look almost the same except that the average delay calculated by the TSR algorithm remains lower than that calculated by the SPR approach. Take for example the case of 16 VPNs where the average delay determined by the TSR heuristic is 27ms compared to that of SPR which is 37ms.

We define the gap as the difference between the average delay calculated for the same number of VPN sources for each of the two TSR and SPR heuristics. We notice that this gap is growing depending on the VPN sources (Fig. 5). From these results, we deduce that the difference between the delays obtained with the two approaches SPR and TSR is quite remarkable. This delay is reduced by applying the TSR heuristic because of the distribution of traffic over a large number of links, which offers more chances of going through small queues.

The fact of going through small queues means that the delay variation is smaller. With TSR, this is illustrated in Fig. 6. This figure gives a variation of delay for the two heuristics SPR and TSR. Indeed, as we have already mentioned during the throughput evaluation, the application of the SPR heuristic under a low traffic intensity is more efficient than that of traffic distribution. The curve presented in Fig. 6 confirms this result.

C. Loss Rate

We propose in the following to estimate the rate of lost packets for each routing technique. As shown in Fig. 7, a large gap between the loss rates obtained with the two heuristics TSR and SPR is perceived. Indeed, we can see that with low traffic intensity, the rate of packets lost by applying the shortest path algorithm is negligible. This rate increases as the number of VPN sources increases to reach a rate of 17×10^{-4} packets lost with 14 sources and 46×10^{-4} packets lost with 24 sources.

However, the traffic distribution algorithm has a higher loss rate than shortest path, which is 4×10^{-4} packets lost for 4 sources. Adding 10 more sources increased this loss rate to 16×10^{-4} lost packets. Similarly, we can notice that from this number of VPN sources (14 sources) the loss rate resulting from the use of the TSR heuristic becomes significantly lower than that obtained by the SPR algorithm.

From the results of the packet loss rate, we can conclude that the TSR algorithm gives lower loss rates for high load networks. Likewise, from Fig. 8, we see that the number of packets sent over the network using the distributed traffic routing technique is larger than that sent by applying the shortest path technique.

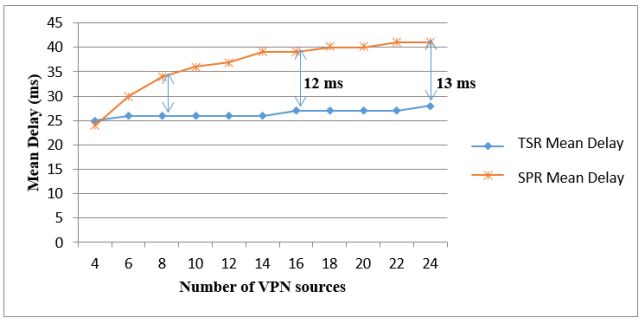

Fig. 5. Mean Delay.

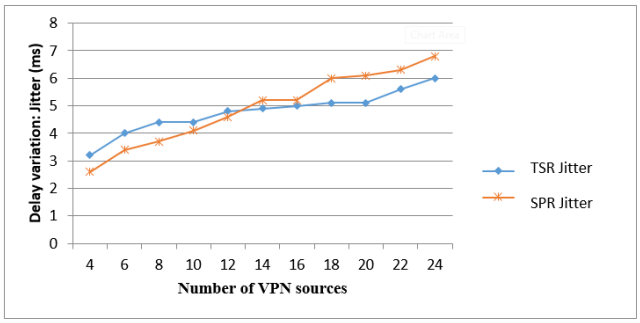

Fig. 6. Delay Variation.

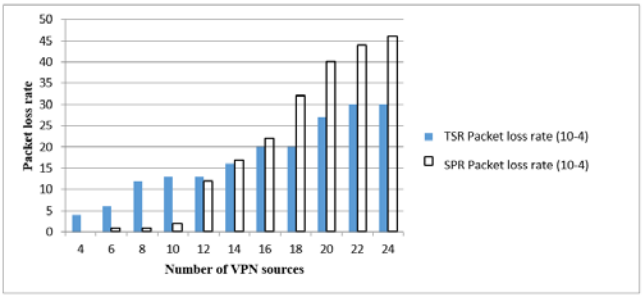

Fig. 7. Rate of Lost Packets.

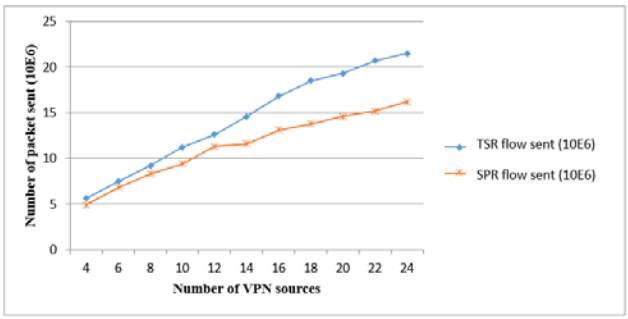

Fig. 8. Flow of Sent Data.

In fact, with the use of the shortest path algorithm, packets have a high chance of passing through overloaded queues, thus rejecting excess packets and therefore retransmission of rejected packets. Furthermore, with the use of the traffic distribution heuristic, there is a high probability of going through lightly loaded paths which results in a shorter routing time, a lower loss rate, as well as more packets sent.

V. Conclusion

This paper has been devoted to present a new routing approach, TSR (Traffic Split Routing) and to compare it to the classic routing of the shortest path SPR (Shortest Path Routing). In the first part, we presented different performance evaluation techniques. Then, we were interested in presenting different simulation tools and we justified our choice for the NS-2 tool. We went on to define the various quality of service parameters that we evaluated. Then, we detailed and analyzed the different simulation results obtained with different scenarios for the two approaches SPR and TSR.

Simulation results presented have demonstrated the effectiveness of TSR in the case of high traffic intensity. Thus, we were able to demonstrate that our approach, TSR, is more satisfactory for ensuring a better quality of service for certain types of applications such as real-time multimedia applications and VOIP which are very sensitive to the variation of speed and delay.

On the other hand, from the simulations, we noticed that the application of the traffic distribution algorithm allowed us to use a maximum of network resources. Indeed, for a scenario with 14 VPNs, we observed that the TSR heuristic used 69% of the network links. While with the shortest path heuristic, only 41% of all links in the network are used. In addition, the TSR approach makes it possible to accommodate a larger number of VPNs for a given objective (given loss rate, given throughput, etc.).

References

[1] B. WAXMAN. “Routing of Multipoint Connections”. IEEE J. on Selected Areas in Communications, numéro 9, volume 6, décembre, 1988, pages 1617-1622.

[2] Erdal Akin and Turgay Korkmaz. An Efficient Binary-Search Based Heuristic for Extended Unsplittable Flow Problem. In Computing, Networking and Communications (ICNC), 2017 International Conference on, pages 831–836. IEEE, 2017.

[3] Erdal Akin and Turgay Korkmaz. Routing Algorithm for Multiple Unsplittable Flows Between Two Cloud Sites with QoS Guarantees. In Computing, Networking and Communications (ICNC), 2017 International Conference on, pages 917–923. IEEE, 2017.

[4] Michael R Garey and David S Johnson. A Guide to The Theory of NP-Completeness. WH Freemann, New York, 70, 1979.

[5] Chen S. and Nahrstedt K., “An Overview of Quality of Service Routing for Next-Generation High Speed Networks: Problems and Solutions”. IEEE Network, 1998.

[6] Jaffe J. M., “Algorithms for Finding Paths with Multiple Constraints”. Networks, 1984. vol. 14: p. 95–116.

[7] J. W. Guck, A. Van Bemten, M. Reisslein, and W. Kellerer. Unicast QoS Routing Algorithms for SDN: A Comprehensive Survey and Performance Evaluation. IEEE Communications Surveys Tutorials, 20(1):388– 415, Firstquarter 2018.

[8] Verma S. Pankaj R., and leon-Garcia A., “QoS Based Multicast Routing for Multimedia Communications”. IEEE Workshop on QoS, 1997.

[9] Wu, Wenfei, Yoshio Turner, and Mike Schlansker. "Routing optimization for ensemble routing." 2011 ACM/IEEE Seventh Symposium on Architectures for Networking and Communications Systems. IEEE, 2011.

[10] J. Homer, S. Zhang, X. Ou, D. Schmidt, Y. Du, S. R. Rajagopalan, and A. Singhal. Aggregating vulnerability metrics in enterprise networks using attack graphs. Journal of Computer Security, 21(4):561–597, 2013.

[11] X. Ou, W. F. Boyer, and M. A. McQueen. A scalable approach to attack graph generation. In Proceedings of the 13th ACM conference on Computer and communications security, pages 336–345. ACM, 2006.

[12] LeMay, E., Scarfone, K., Mell, P.: The common misuse scoring system (CMSS): Metrics for software feature misuse vulnerabilities. US Department of Commerce, National Institute of Standards and Technology (2012).

[13] Ou, X., Singhal, A.: Security risk analysis of enterprise networks using attack graphs. In: Quantitative Security Risk Assessment of Enterprise Networks, pp. 13–23. Springer (2011).

[14] P. S. Sarker, V. Venkataramanan, D. S. Cardenas, A. Srivastava, A. Hahn and B. Miller, "Cyber-Physical Security and Resiliency Analysis Testbed for Critical Microgrids with IEEE 2030.5," 2020 8th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems, Sydney, Australia, 2020, pp. 1-6, doi: 10.1109/MSCPES49613.2020.9133689.

[15] M. Bagaa, D. L. C. Dutra, T. Taleb and K. Samdanis, "On SDN-Driven Network Optimization and QoS Aware Routing Using Multiple Paths," in IEEE Transactions on Wireless Communications, vol. 19, no. 7, pp. 4700-4714, July 2020, doi: 10.1109/TWC.2020.2986408.

[16] Faycal Bensalah and Najib El Kamoun, “Novel Software-Defined Network Approach of Flexible Network Adaptive for VPN MPLS Traffic Engineering” International Journal of Advanced Computer Science and Applications(IJACSA), 10(4), 2019.

[17] R. A. Mishra, A. Kalla, K. Shukla, A. Nag and M. Liyanage, "B-VNF: Blockchain-enhanced Architecture for VNF Orchestration in MEC-5G Networks," 2020 IEEE 3rd 5G World Forum (5GWF), Bangalore, India, 2020, pp. 229-234.

[18] Kohei Arai, “Routing Protocol based on Floyd-Warshall Algorithm Allowing Maximization of Throughput” International Journal of Advanced Computer Science and Applications(IJACSA), 11(6), 2020.

[19] J Wognin Vangah, Sié Ouattara, Gbélé Ouattara and Alain Clement, “Global and Local Characterization of Rock Classification by Gabor and DCT Filters with a Color Texture Descriptor” International Journal of Advanced Computer Science and Applications(IJACSA), 10(4), 2019.

[20] V. Q. Rodriguez, F. Guillemin and A. Boubendir, "5G E2E Network Slicing Management with ONAP," 2020 23rd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN), Paris, France, 2020, pp. 87-94.

[21] S. Yucel, "Algorithmic Framework for QoS and TE in Virtual SDN Services," 2019 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2019, pp. 1494-1499.