IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 11 July 2022, accepted 17 July 2022, date of publication 27 July 2022, date of current version 1 August 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3194115

RESEARCH ARTICLE

Dynamic Traffic Engineering Considering Service Grade in Integrated Service Network

SHOHEI KAMAMURA, (Member, IEEE)
Faculty of Science and Technology, Seikei University, Tokyo 180-8633, Japan
e-mail: shohei-kamamura@st.seikei.ac.jp

ABSTRACT This study proposes a dynamic traffic engineering method to improve network resource utilization efficiency without affecting important services in an integrated service network where multiple grades of service are accommodated. Network resource utilization is biased over time because future fluctuations in traffic demand are unpredictable. Although a service network provider prefers to re-optimize network resource utilization using a dynamic traffic engineering, it is often superfluous for users who do not want their mission-critical communications, for example, executive web meetings or online surgery, to be affected. Therefore, we classified traffic into very important packets and commoners, and only commoners were dynamically optimized. To classify traffic, we assigned the identifier to a packet on the application side. In this approach, only the application and edge nodes of the network required modification; the core network routers were unaffected. Furthermore, we established the architecture of the edge node using a software-defined network controller and P4 switch. We also provided an algorithm for optimization based on linear programming and demonstrated that it is possible to improve network resource utilization efficiency without affecting very important packets and that the computational and memory costs are practical.

INDEX TERMS Linear programming, P4, software defined networking, traffic engineering.

I. INTRODUCTION

In Society 5.0, telecom networks play an important role in providing virtual spaces in which diverse data are closely linked. In telecom networks, to efficiently accommodate a variety of traffic that increases year by year, a network is designed as an integrated service network. The integrated network provides multiple grades of services, such as an access network to the best-effort Internet or a guaranteed virtual private network (VPN).

For the unstoppable traffic growth that has continued over the past few decades in telecom networks, traffic engineering (TE) techniques have been widely studied [1]–[12]. TE techniques optimize traffic routes statically or dynamically to improve network resource utilization. In particular, the recent emergence of technologies that increase network programmability, such as software-defined networks [13] and P4 switch [14], has increased the feasibility of dynamic TE techniques. For example, network resource utilization is biased over time, because future fluctuations in traffic demand are unpredictable. Applying TE techniques to this situation can eliminate resource utilization bias and postpone the timing of capital investment.

Although dynamic TE improves the efficiency of equipment utilization from the perspective of the service network provider, it has a risk of instantaneous quality degradation caused by rerouting. From the user's perspective, if they use mission-critical communications, such as executive web meetings or online surgery, and are satisfied with the current quality of services, they would rather not change their networks. Because network operators understand the user's psychology, TE is difficult to implement. On this basis, we want to solve the problem of increasing the efficiency of network resource utilization while minimizing the risk of quality degradation for users.

To solve the above problem, our simple key idea, illustrated in Fig. 1, is to optimize only the route of traffic that is safe to change and fix the route of important traffic. For example, the traffic of business services, such as IP-VPN, are treated as very important packets (VIPs) traffic, and their routes

The associate editor coordinating the review of this manuscript and approving it for publication was Hosam El-Ocla.

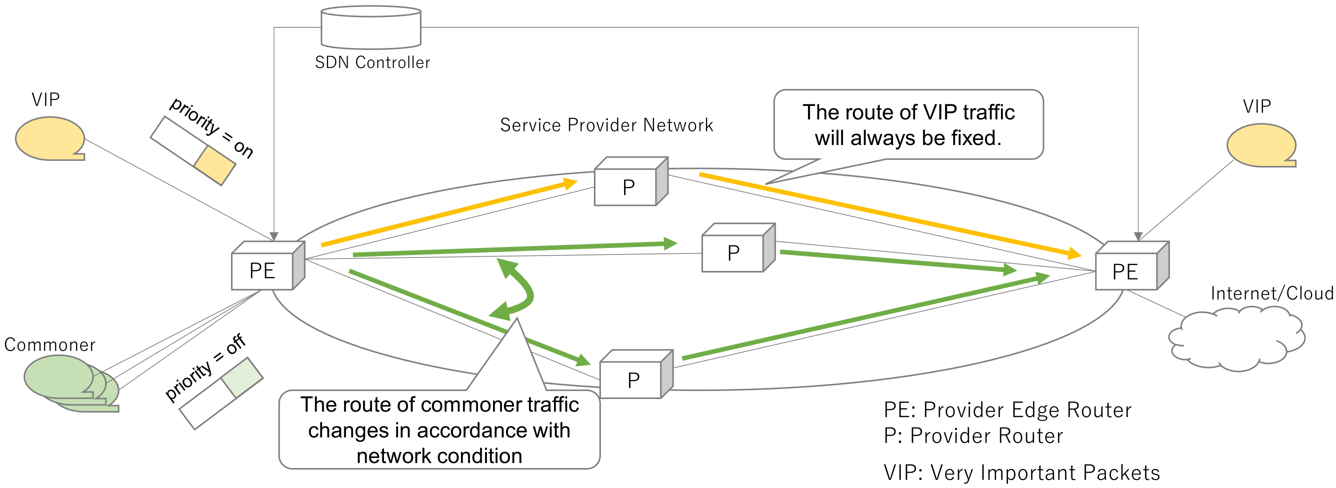


FIGURE 1. Overview of dynamic traffic engineering considering service grade in integrated service network. While the route of VIP traffic will always be fixed, the route of commoner traffic changes in accordance with network conditions.

never change. In contrast, best-effort traffic is regarded as commoner traffic, and its route may be changed by optimization. In similar approaches, focusing on the existence of premium and best-effort traffic, Hai *et al.* proposed a method to differentiate the degree of protection in elastic optical networks [15], [16]. Their work is very important for ensuring reliability at lower optical layer. In the IP layer, on the other hand, although many TE methods [1]–[12] have been proposed, no TEs have solved our problem.

To realize this idea, we present (i) an appropriate classification method for traffic, and (ii) a novel TE algorithm based on this classification. There are two approaches to classification: passive and active. The passive approach, which checks the packet header information of flows in detail observed in routers, cannot reliably classify users owing to the flow dynamics. Therefore, we took a simple active approach, which gives the identifier to a packet on the application side [17] and then classifies the traffic by referring to the identifier. We also propose a network architecture to handle this application extension by extending only the edge routers, without changing the core routers. For the TE algorithm, we propose a simple but practical algorithm based on linear programming (LP). We also propose an algorithm to implement the solution computed by LP into a forwarding table on a P4 switch.

The objective of this study is to provide a practical TE method with two key contributions. First, we propose a simple classification mechanism for services by coordinating the application and edge routers in the network. In this approach, only the applications and edge routers of the network were modified, with no impact on the core network routers. Extending the application was easier than extending the routers, and the edge router was extended without modifying commercial products using a software-defined network (SDN) controller and P4 switch. Second, we propose a novel and practical TE algorithm based on the above classification to optimize only the route of traffic that is safe to change and fix the route of important traffic. In one of the scenarios where the VIP traffic was moderately accommodated, our proposed method never lost any VIP traffic and achieved a performance improvement of approximately 25% compared to the benchmark method. In addition, the performance degradation was within 10% compared with the overall optimization, where the VIP traffic was highly variable.

The remainder of this paper is organized as follows. Section II discusses the related work. In Section III, we provide our network model and the problem statement. To solve our problem of network architecture, the dynamic TE algorithm and the forwarding (match-action) table creation algorithm are presented in Section IV. The effectiveness of the proposed method is evaluated in section V, and the conclusions are presented in Section VI.

II. RELATED WORKS

A dynamic TE consists of a cycle called the closed loop of (i) monitoring, (ii) computation, and (iii) control. The monitoring phase collects the topology and traffic states, which are used as inputs for the optimization. Traffic monitoring is particularly important because traffic exhibits dynamic and complex behavior. For the computation phase, many route-computation algorithms for optimization have been studied. For the control phase, studies on network architecture focused on SDN are a research trend. This section reports studies on traffic monitoring and route computation, which are closely related to our study.

For (i), traffic monitoring technologies are mainly classified into link-by-link traffic monitoring with the simple network management protocol [18] or network telemetry [19], [20] and flow monitoring with xFlow technologies [21]–[30]. Link-by-link monitoring obtains the value of interface counters in a router, whereas flow monitoring computes statistics of flows using the sampling [21]–[24] or probabilistic data structure [25]–[30].

The xFlow technology monitors end-to-end traffic and is comprised of two methods: flow statistics approaches, such as NetFlow [21] and IPFIX [22], and header sampling approaches, such as sFlow [23] and IPFIX IE315 [24]. A flow is defined as a set of IP packets passing an observation point in the network during a certain time interval, such that all packets belonging to a particular flow have a set of common properties [22]. For example, the flow is defined as 5-tuple: source IP address, destination IP address, source port, destination port, and protocol type. Because the volume of the flow is large, packets are observed with sampling to reduce the loads of the router and flow collector. Aggregated flow data are called a traffic matrix and are used as inputs to the optimization computation. Traffic matrix estimation [31], [32] has been proposed as another approach for monitoring end-to-end traffic. Traffic matrix estimation produces end-to-end traffic from link-by-link traffic using pseudo-reverse matrix computation. The drawback of the xFlow technology is the low accuracy caused by the sampling mechanism. To improve accuracy, more efficient monitoring of flow statistics using a probabilistic data structure has been proposed [25]–[30]. Because these novel monitoring mechanisms require router enhancement, they are mainly implemented using the P4 switch [14]. The basic idea of these new flow statistics [25]–[30] is to use a probabilistic data structure, such as the Bloom filter [33] and count-min-sketch [34]. In addition, these methods also reduce the amount of data by appropriately selecting the data to be collected. For example, HashFlow [29] only collects detailed information on elephant flow, which has many packets. Furthermore, because the count-min-sketch is non-invertible, an invertible MV-sketch [30], which allows all heavy flows to be recovered from only the sketch data structure, was proposed.

Next, we introduce the trends for (ii), that is, the route computation for TE. The origin of TE can be traced to [1], where it is shown that MPLS explicit routing can be formulated as an LP with a multi-commodity flow problem. Subsequently, as IP networks became mainstream, the link cost optimization problem for the open shortest path first protocol was defined [2]. Because the link cost optimization problem is NP-hard, a heuristic solution was proposed. At that time, flow monitoring methods were still in their infancy; therefore, it was assumed that obtaining complete traffic information would be difficult. Therefore, there was a lot of research on oblivious routing [3], which optimizes the worst-case scenario by only observing the link traffic at each edge node. Recently, semi-oblivious routing [4] has been proposed as an extension of oblivious routing. Because SDN [13] has become popular, the scalability of the algorithm has become the main issue of TE algorithms. Therefore, there are many scalable route computation and bandwidth resource allocation methods such as B4 [5], [6], SWAN [7], FFC [8], BwE [9], TEAVAR[10], QROUTE[11], and ARROW[12]. In particular, failure-aware TE such as SWAN [7] or TEAVAR [10] has been considered important to avoid the loss of business opportunities. On the other hand, capacity lost due to failure can only be partially restored. Therefore, ARROW [12], a cross-layer optimization of which IP links should be restored and then traffic engineering should be performed, was proposed.

In this study, we consider solving a critical problem for telecom carriers, which is to improve resource utilization efficiency without affecting important traffic (VIP traffic) by integrating the network and the application [17]. Some similar approaches have been studied. At the optical layer, for example, there are studies to differentiate path protection grades by considering traffic characteristics [15], [16]. Also, at the IP layer, traffic engineering methods have been proposed to follow dynamically changing QoS parameters [11]. To the best of our knowledge, however, no TEs have solved our problem.

III. NETWORK MODEL AND PROBLEM STATEMENT

This section describes the network model and problem statement. We adopted the semantically aware mission-oriented (SAMO) network [17] into our network architecture. In the SAMO network, the application assigns an identifier, called semantic information, to the IP packet header. By referring to this identifier, the router realizes forwarding based on its priority. The OpenFlow method [13] works in a similar manner. OpenFlow [13] uses not only IP addresses but also additional information, such as 12-tuple information from Layer 1 to Layer 4 for routing. However, OpenFlow flexibly controls routing within a telecom network in accordance with its routing policy. Specifically, to determine a proper routing according to user requests, it is necessary to link the service provider's routing policy with user-owned information and intentions; this is an indirect method. In contrast, the SAMO network [17] allows applications to embed the user's intent directly into the network and enables network control based on this intent.

Our goal is to increase the efficiency of network resource utilization while minimizing the risk of quality degradation for important users. To achieve this goal, the appropriate classification of traffic and the TE based on this classification are key issues. Considering these issues, traffic classification using SAMO is a promising approach. However, SAMO [17] proposed a framework for passing semantic information from an application to the network. Current large-scale IP networks consist of vendor-proprietary high-speed routers. Although SAMO functions can be implemented in a user's application, it is difficult to prepare an SAMO-capable router in a network. Therefore, our task is to establish a feasible traffic classification architecture based on SAMO and a TE algorithm that works in our architecture.

IV. DYNAMIC TRAFFIC ENGINEERING CONSIDERING SERVICE GRADE

This section describes the dynamic TE method. The key technologies of our TE consist of the network architecture for feasible traffic classification, dynamic TE algorithm, and forwarding (match-action) table creation algorithm.

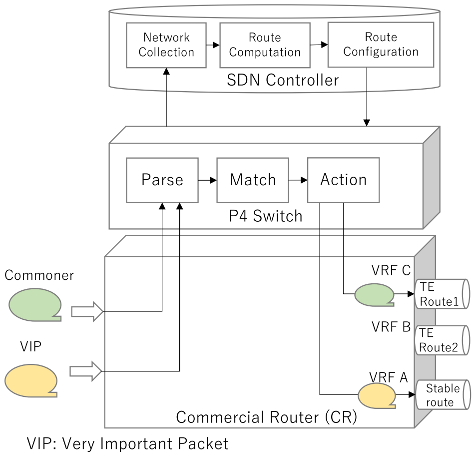


FIGURE 2. Processes in provider edge (PE).

t = t0 (1st week)

Match 1 Priority	Match 2 Destination Prefix	Other Match Actions(*1)	Action 1 SET VID	Action 2 Forwarding
on	10.0.10.0/24		Set VID=A	To CR
off	20.0.10.0/24		Set VID=C	To CR
off	30.0.10.0/24		Set VID=C	To CR

t = t1 (2nd week)

Match 1 Priority	Match 2 Destination Address	Other Match Actions(*1)	Action 1 SET VID	Action 2 Forwarding
on	10.0.10.0/24		Set VID=A	To CR
off	20.0.10.0/24		**Set VID=B**	To CR
off	30.0.10.0/24		**Set VID=B**	To CR

(*1) Source Address, Source/Destination Port, and Protocol Type, If necessary

FIGURE 3. Example of match-action table.

A. NETWORK ARCHITECTURE

An overview of the dynamic TE considering the service grade in an integrated service network is illustrated in Fig. 1. Because the service provider network is integrated, it provides two services in a single network. In this example, the VPN communication between remote locations is regarded as VIP traffic, and the access network to the Internet and cloud is regarded as commoner traffic. Because the VIP traffic has a high priority, the priority bit is turned on through the semantic information, whereas the priority bit is turned off for the commoner traffic. Assigning semantic information is not difficult because it is performed as application programming in a PC or home gateway. For example, the priority bit can be turned on using the DSCP field of IPv4 or the traffic-class field of IPv6. The provider edge (PE) then receives the packet and performs appropriate routing in accordance with the priority bit. In this example, the VIP traffic is forwarded to a fixed route that never changes depending on the situation, whereas the routes for the commoner traffic change depending on the situation. This PE control was implemented using an SDN controller.

Next, we present PE processing, as shown in Fig. 2. Although commercial routers provide high-speed packet forwarding, their internal structure is vendor proprietary, and extending the inner logic to interpret semantic information is not practical. Therefore, we solve this issue by incorporating a commercial router and P4 switch, as shown in Fig. 2. In the commercial router, policy-based routing (PBR) is configured to forward all incoming packets to the P4 switch. The P4 switch then parses the packet header information and determines the routes for each packet based on the priority bit, destination IP prefix, and other information if necessary. In the commercial router, one method of distributing packets to the routes determined by the P4 switch is to use virtual routing and forwarding (VRF). In particular, as many VRFs as the number of routes needed for the TE are configured in the router, and the VLAN ID (VID) corresponding to each VRF is assigned to the P4 switch. The match and action fields, which represent the routing information, in the P4 switch are computed using the SDN controller. The SDN controller (i) collects topology information and traffic information, (ii) computes the optimal routes, and (iii) periodically configures the routes in the P4 switches. Route computation is described in Section 3.2, and match-action table creation is described in Section 3.3.

Fig. 3 illustrates an example of a match-action table in the P4 switch. The VRF is determined by matching conditions, such as the priority bit and destination IP address. Then, the corresponding VID is set and packets are sent back to the commercial router. In this example, the route of packets with the priority bit off is forwarded to a different VRF (VID=B) a week later, whereas the route of packets with the priority bit does not change. The route computation is described in detail in the next section.

Note that the processing between the commercial router and P4 switch is a real-time process, whereas the processing between the P4 switch and SDN controller is a batch process. In particular, frequent dynamic TE, even for commoner

traffic, can destabilize the network and should be avoided as much as possible. Therefore, as shown in Fig. 3, we assume that TE will be performed once a week, for example, during the maintenance window at midnight on Sundays when the impact on users is low.

B. DYNAMIC TRAFFIC ENGINEERING ALGORITHM

The dynamic TE algorithm in the SDN controller is performed using the following steps. In the SDN controller, network topology information is given as a graph $G = (V, E, C)$, where V is the set of nodes, E is the set of links, and C is the capacity of each link. It is also assumed that flow measurement protocols, such as IP flow information export (IPFIX) [22], are used to obtain the end-to-end traffic matrix among all nodes.

STEP1: Classify the traffic into the traffic matrix set $T_p = \{t_p^1, t_p^2, \cdots, t_p^K$ whose priority bit are on, and traffic matrix set $T_n = \{t_n^1, t_n^2, \cdots, t_n^K$ whose priority bit are off.

STEP2: From T_p and its routing matrix A_p, the link traffic of the VIP traffic X_p is calculated using the following equation [31]:

$$X_p = A_P T_p \tag{1}$$

STEP3: Compute graph $G' = (V, E, C')$, where X_p is subtracted from the original link capacity C.

STEP4: Compute the optimal routes $A_n = (A_{ij})_{|E|\times|K|}$ with $G' = (V, E, C')$ and T_n as input, and the following linear programming (LP) formulations.

$$min \ \ \alpha \tag{2}$$

$$\sum_{j:(i,j)\in E} A_{ij}^k - \sum_{j:(j,i)\in E} A_{ji}^k = \begin{cases} 1, i = s_k \\ 0, i \neq s_k, t_k \quad k \in K \\ -1, i = t_k \end{cases} \tag{3}$$

$$\sum_{k\in K} t_n^k A_{ij}^k \leq C'_{ij}\alpha, (i,j) \in E \tag{4}$$

$$0 \leq A_{ij}^k \leq 1 \tag{5}$$

$$\alpha \geq 0 \tag{6}$$

In STEP2, A_p is computed using the LP optimization shown in STEP4 or is determined statically according to the business policy as the default route. Because the routes for VIP traffic are operated in a fixed manner, default routes are always used in principle. Conversely, the routes for the commoner traffic are always updated by the LP optimization calculations in STEP4 based on the available capacity computed in STEP3. In STEP4, the objective function α is the maximum link utilization, and load balancing is achieved by minimizing it in (2). Because s_k and t_k are the start and end nodes of flow k, respectively, (3) represents the flow conservation law: the traffic flowing into a node has to equal the traffic flowing out of the node for any node excepting s_k and t_k. Equation (4) is the capacity constraint to ensure that link usage does not exceed the available capacity $C' = (C'_{ij})_{|E|}$. This means that the total amount of bandwidth consumed by all traffic on a link should not exceed α times the total capacity of the link. Therefore, minimizing α minimizes the maximum link utilization.

C. CREATION OF MATCH-ACTION TABLE

Because the LP approach produces solutions with a real number, as shown in (5), it may calculate multiple routes between the start and end node pairs k. For this case, this section provides an algorithm for creating a match-action table to appropriately assign the input traffic to multiple routes.

First, suppose that M routes are computed for some start and end node pairs k. Providing these M routes with actual devices or protocols can be realized using explicit routing technologies, such as segment routing over multi-protocol label switching (MPLS) [35]. On the other hand, an arbitrary traffic distribution ratio R_m, $\sum_{m=1}^{M} R_m = 1.0$ is defined for M routes in the range from 0 to 1. In other words, a proper match-action table must be created to distribute traffic in accordance with ratio R_m. Here, we focus on the fact that P4 switches can identify flows with a 5-tuple and create a match-action table using the following procedure.

STEP1: Computing $R_m = round\,(R_m, N)\,, m = 1, 2, \cdots M$.

STEP2: Creating a hash table of size 10^N.

STEP3: Mapping the flows to the hash table using an arbitrary hash function.

STEP4: Getting the routes (VID) of flows by referring to the hash value route mapping table.

In STEP1, $round\,(X, N)$ is the process of rounding digits X to N. To simplify the calculations, N is set to 2. To increase accuracy, N is increased. In STEP2, a hash table with a size of 10^N is created. In STEP3, a hash value is obtained using the flows, which are identified by packet header information, such as a 5-tuple, as input. Fig. 4 shows an example of the hash value route-mapping table in STEP4. For example, given three TE routes and their R_m, as shown in Fig. 4(a), 100 hash values are calculated when N = 2. The hash value route-mapping table, in which these hash values are assigned according to the ratio of each route, is shown in Fig. 4(b). Referring to this table, the corresponding routes, or VID, of each flow can be obtained.

Although the P4 switch can identify flows at 5-tuple granularity, the 5-tuple match condition may result in a large table size. Therefore, if the destination IP prefix alone provides the resolution necessary to achieve the distribution ratio, the match condition should be the priority bit and destination IP prefix. If this is insufficient, the reference field can be increased gradually. Thus, unnecessary increases in the match table can be prevented.

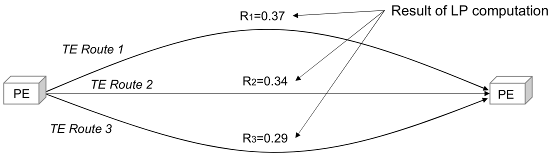
V. EVALUATION

The purpose of the evaluation was to quantitatively confirm the effect of fixing the allocation of the VIP traffic and optimizing only the commoner traffic allocation. The evaluation topology used was the Google B4 topology [6] and the wide-area data center network shown in Fig. 5. The traffic matrix is given by the gravity model [32], based on the population distribution in each country. Specifically, the amount of input and output link traffic at a node is proportional to the population, and the end-to-end traffic $T\,(a \to b)$, which represents traffic from node a to node b, is calculated as follows:

$$T\,(a \to b) = X_a^{in} \times \frac{X_b^{out}}{\sum_i X_i^{out}} \tag{7}$$

X_a^{in} represents the incoming traffic to **a**, and $\mathbf{X_b^{out}}$ represents the outgoing traffic from b.

The VIP and commoner traffic matrices are computed independently using (7). The link traffic (X_i^{in} and X_j^{out}) of the commoner traffic is the square of the VIP traffic, such that the amount of commoner traffic is more intensely distributed. These traffic matrices are assumed to be at time t when the traffic distribution is unbalanced and dynamic TE is performed at time t. We prepared two comparison methods for our proposed method. One is the conventional



(a) Network example

Hash value	Route	VID
From 01 To 37	TE Route1	C
From 38 To 71	TE Route2	B
From 72 To 100	TE Route3	A

(b) Example of hash-value route mapping table

FIGURE 4. (a) Network example (b) Example of hash-value route mapping table.

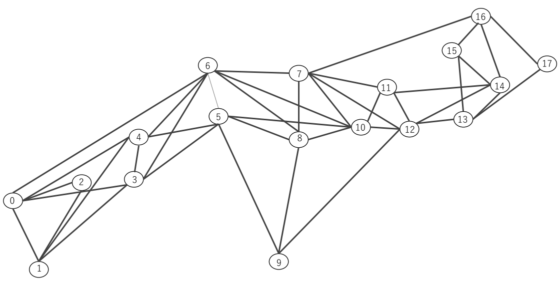


FIGURE 5. Evaluation topology (18 nodes, 80 links).

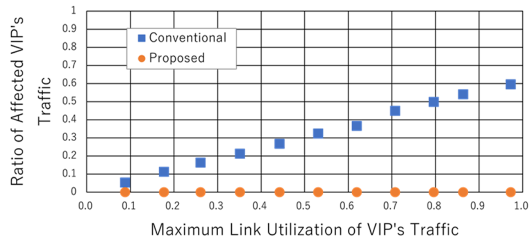


FIGURE 6. Ratio of affected very important packets (VIPs) traffic when the maximum link utilization of VIP traffic is varied.

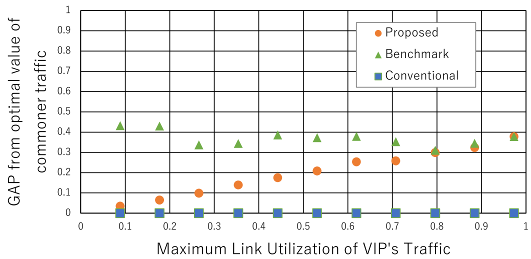


FIGURE 7. GAP from optimal value when the maximum link utilization of VIP traffic is varied.

method, which includes VIP traffic in the optimization. The other is the benchmark method, which does not change the routes of the VIP traffic but simply assigns the shortest path to the commoner traffic. The evaluation metric is the ratio of the affected VIP traffic to the GAP value. Affected here means that even a part of the routes is changed as a result of optimization. The GAP value is calculated as the difference between the maximum link utilization when optimizing only the commoner traffic using the proposed method and the maximum link utilization when optimizing the entire traffic, including VIP traffic, by the conventional method; the smaller the GAP value, the better the performance. As the ratio of VIP traffic increases, the available bandwidth for the commoner traffic decreases. Therefore, for fairness, the maximum link utilization computed by the proposed method is calculated under the updated link capacity, where the bandwidth used by the VIP traffic is eliminated.

A. EFFECTIVENESS OF SERVICE-GRADE CONSIDERATION

Fig. 6 shows the ratio of affected VIP traffic for the conventional and proposed methods. The horizontal axis represents the maximum link utilization of the VIP traffic, and an increase in this value indicates an increase in the percentage of VIP traffic in the network. The benchmark method is not shown here because it yields the same results as those of the proposed method. In the proposed method, the traffic of the VIP is excluded from the optimization, and the traffic of the affected VIP is always zero because its routes never change. However, the results of the conventional method show that the amount of traffic affected by the VIP increases as the maximum link utilization of the VIP traffic in the network increases. These results suggest an obvious but very important point: to optimize network usage, the telecom carrier must not interfere with VIP traffic in any way. The results of the conventional method imply that TE has been difficult to implement in an actual network, although various techniques have been studied for more than 20 years.

Fig. 7 shows the GAP values for each method when the maximum link utilization of VIP traffic is varied. For the proposed method, the GAP from the conventional method, which optimizes all traffic, ranges from 3% to 38%. As can be observed from the comparison between the proposed method and the benchmark method, optimization of the commoner traffic alone is sufficiently effective for regions with low maximum link utilization of 0.1 to 0.7. For example, performance improvements of up to 40% were observed when the maximum link utilization of VIP traffic was 0.1.

However, it is also confirmed that the advantage of the proposed method over the benchmark method disappears when the maximum link utilization ratio is 0.8 to 1.0. This is because the available resources are no longer sufficient, and thus the optimized routes are similar to the simple shortest routes. Note that when the maximum link utilization of the VIP traffic was one, the routes of the proposed method and the conventional method matched exactly.

Large-scale telecom networks are designed to maintain the link utilization under 50% when accommodating the entire traffic, including VIP traffic and commoner traffic. Assuming that VIP traffic accounts for at least half of the total traffic, the maximum link utilization of VIP traffic is of approximately 0.25. In such a situation, the proposed method achieves a performance improvement of approximately 25% compared with the benchmark method. In addition, the performance degradation was within 10% compared with the conventional method, where the VIP traffic is highly variable; the proposed method is highly effective under actual conditions.

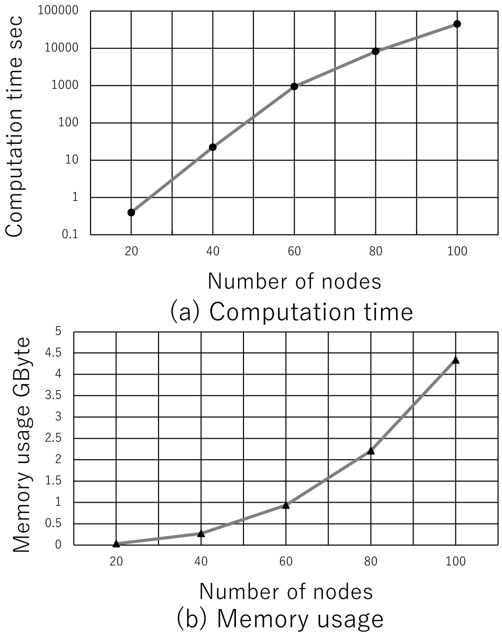


FIGURE 8. **Computation time and memory usage in SDN controller when number of nodes changes.**

B. FEASIBILITY ANALYSIS

Next, the feasibility of the proposed method is discussed from the implementation perspective. There are two issues: memory cost and computational cost. For the memory cost, increased table size based on additional match conditions referring to the priority bit and other fields in the P4 switch becomes the main issue. In terms of the computational cost, the computation time for route optimization using the LP approach in the SDN controller is dominant.

Regarding memory cost, the table size is at least doubled because the proposed architecture refers to the priority bit in addition to conventional routing that looks up the destination IP address. If this condition is not acceptable, it is possible to fix the VIP traffic with a specific single route. Because packets with the priority bit turned on can be assigned a fixed specific VID without destination IP address matching processing, the number of entries to be increased in the P4 switch can be only one. In addition, when handling information equivalent to an Internet full route, for example, 880,000 IPv4 routes, these sizes are sufficiently larger than the hash table (see III.C). That is, the destination IP address (prefix) is sufficient as the only field to be referenced, other than the priority bit.

Regarding computational cost, the computation time and memory used for optimization by LP are shown in Fig. 8. The topology used in the evaluation was the Barabási-Albert model [36], whose node-degree distribution follows a power law. The average node degree was set as 4.0, and the number of nodes was increased to 100. The computing environment used was an Intel Core i7-10700 CPU 2.9GHz (8-core) with 16GB of memory. There were no major concerns regarding memory (Fig. 8 (b)). While the computation time is less than 1 s at 20 nodes, as the number of nodes increases, the computation time increases exponentially, reaching 44500 seconds (12.5 h) for 100 nodes, as shown in Fig. 8(a). However, as described in Section 3, our proposed scheme does not change routes in real time but periodically, such as

once a week during the maintenance window period. Therefore, we believe that the computation time for 100 nodes is practical.

VI. CONCLUSION

We proposed a dynamic TE method to improve network resource utilization efficiency without affecting important services in an integrated service network. Because the existing passive classification cannot reliably classify services owing to flow dynamics, we took a simple active approach, which gives the identifier to a packet on the application side and then classifies the traffic by referring to the identifier in the edge router. We also proposed a network architecture to handle this application extension by extending only the edge routers, without changing the core routers. In addition, the edge router is extended without modifying the commercial parts using an SDN controller and P4 switch. We also provided a simple and practical algorithm for optimization based on LP and showed that it is possible to improve network resource utilization efficiency without affecting VIP traffic, and that the computational and memory costs are also practical.

For future work, we would like to implement the proposed network using an SDN controller and P4 switch and then demonstrate its effectiveness in mission-critical services, such as executive web meetings or online surgery. In addition, our current method is based on a simple LP method for the optimization of commoner traffic. Therefore, we believe that it can be further extended by integrating it with other promising TE algorithms like network coding [37].

REFERENCES

[1] Y. Wang and Z. Wang, "Explicit routing algorithms for internet traffic engineering," in *Proc. 8th Int. Conf. Comput. Commun. Netw.*, Sep. 1999, pp. 582–588.

[2] B. Fortz and M. Thorup, "Internet traffic engineering by optimizing OSPF weights," in *Proc. IEEE INFOCOM Conf. Comput. Commun. 19th Annu. Joint Conf. IEEE Comput. Commun. Societies*, Mar. 2000, pp. 519–528.

[3] M. Kodialam, T. V. Lakshman, J. B. Orlin, and S. Sengupta, "Oblivious routing of highly variable traffic in service overlays and IP backbones," *IEEE/ACM Trans. Netw.*, vol. 17, no. 2, pp. 459–472, Apr. 2009, doi: 10.1109/TNET.2008.927257.

[4] P. Kumar, Y. Yuan, C. Yu, N. Foster, R. Kleinberg, P. Lapukhov, C. L. Lim, and R. Soulé, "Semi-oblivious traffic engineering: The road not taken," in *Proc. USENIX NSDI*, Apr. 2018, pp. 157–170.

[5] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, and J. Zolla, "B4: Experience with a globally-deployed software defined WAN," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 4, pp. 3–14, 2013, doi: 10.1145/2534169.2486019.

[6] C.-Y. Hong, S. Mandal, M. Al-Fares, M. Zhu, R. Alimi, C. Bhagat, S. Jain, J. Kaimal, S. Liang, K. Mendelev, S. Padgett, F. Rabe, S. Ray, M. Tewari, M. Tierney, M. Zahn, J. Zolla, J. Ong, and A. Vahdat, "B4 and after: Managing hierarchy, partitioning, and asymmetry for availability and scale in Google's software-defined WAN," in *Proc. Conf. ACM Special Interest Group Data Commun.*, Aug. 2018, pp. 74–87.

[7] C.-Y. Hong, S. Kandula, R. Mahajan, M. Zhang, V. Gill, M. Nanduri, and R. Wattenhofer, "Achieving high utilization with software-driven WAN," in *Proc. ACM SIGCOMM Conf. (SIGCOMM)*, Aug. 2013, pp. 15–26.

[8] H. H. Liu, "Traffic engineering with forward fault correction," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 4, pp. 527–538, Aug. 2014, doi: 10.1145/2740070.2626314.

[9] A. Kumar, S. Jain, U. Naik, A. Raghuraman, N. Kasinadhuni, E. C. Zermeno, C. S. Gunn, J. Ai, B. Carlin, M. Amarandei-Stavila, M. Robin, A. Siganporia, S. Stuart, and A. Vahdat, "BwE: Flexible, hierarchical bandwidth allocation for WAN distributed computing," in *Proc. ACM Conf. Special Interest Group Data Commun.*, Aug. 2015, pp. 1–14.

[10] J. Bogle, N. Bhatia, M. Ghobadi, I. Menache, N. Bjørner, A. Valadarsky, and M. Schapira, "TEAVAR: Striking the right utilization-availability balance in WAN traffic engineering," in *Proc. ACM Special Interest Group Data Commun.*, Aug. 2019, pp. 29–43.

[11] N. Varyani, Z.-L. Zhang, and D. Dai, "QROUTE: An efficient quality of service (QoS) routing scheme for software-defined overlay networks," *IEEE Access*, vol. 8, pp. 104109–104126, 2020, doi: 10.1109/ACCESS.2020.2995558.

[12] Z. Zhong, M. Ghobadi, A. Khaddaj, J. Leach, Y. Xia, and Y. Zhang, "ARROW: Restoration-aware traffic engineering," in *Proc. ACM SIGCOMM Conf.*, Aug. 2021, pp. 560–579.

[13] D. Kreutz, F. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015, doi: 10.1109/JPROC.2014.2371999.

[14] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," *SIGCOMM Comput. Commun. Rev.*, vol. 44, pp. 87–95, Jul. 2014, doi: 10.1145/2656877.2656890.

[15] D. T. Hai, H. T. Minh, and L. H. Chau, "QoS-aware protection in elastic optical networks with distance-adaptive and reconfigurable modulation formats," *Opt. Fiber Technol.*, vol. 61, Jan. 2021, Art. no. 102364, doi: 10.1016/j.yofte.2020.102364.

[16] D. T. Hai, "On the spectrum-efficiency of QoS-aware protection in elastic optical networks," *Optik*, vol. 202, Feb. 2020, Art. no. 163563, doi: 10.1016/j.ijleo.2019.163563.

[17] T. J. Salo and Z.-L. Zhang, "Semantically aware, mission-oriented (SAMO) networks: A framework for application/network integration," in *Proc. Workshop Netw. Appl. Integr./CoDesign*, Aug. 2020, pp. 41–42.

[18] J. Case, *A Simple Network Management Protocol (SNMP)*, document IETF RFC1157, May 1990.

[19] M. Yu, "Network telemetry: Towards a top-down approach," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 49, no. 1, pp. 11–17, Feb. 2019, doi: 10.1145/3314212.3314215.

[20] C. Kim, A. Sivaraman, N. Katta, A. Bas, A. Dixit, and L. J. Wobker, "Inband network telemetry via programmable dataplanes," in *Proc. ACM SIGCOMM Conf. Posters Demos*, Aug. 2015, pp. 1–2.

[21] B. Claise, *Cisco Systems NetFlow Services Export Version 9*, document IETF RFC3954, Oct. 2004.

[22] B. Claise, *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information*, document IETF RFC7011, Sep. 2013.

[23] *sFlow*. Accessed: Apr. 2022. [Online]. Available: http://www.sflow.org/

[24] S. Kashima, *Information Elements for Data Link Layer Traffic Measurement*, document IETF RFC7133, May 2014.

[25] Y. Li, R. Miao, C. Kim, and M. Yu, "FlowRadar: A better NetFlow for data centers," in *Proc. NSDI*, 2016, pp. 311–324.

[26] Z. Liu, A. Manousis, G. Vorsanger, V. Sekar, and V. Braverman, "One sketch to rule them all: Rethinking network flow monitoring with UnivMon," in *Proc. ACM SIGCOMM Conf.*, Aug. 2016, pp. 101–114.

[27] Q. Huang, X. Jin, P. P. C. Lee, R. Li, L. Tang, Y.-C. Chen, and G. Zhang, "SketchVisor: Robust network measurement for software packet processing," in *Proc. Conf. ACM Special Interest Group Data Commun.*, Aug. 2017, pp. 113–126.

[28] T. Yang, J. Jiang, P. Liu, Q. Huang, J. Gong, Y. Zhou, R. Miao, X. Li, and S. Uhlig, "Elastic sketch: Adaptive and fast network-wide measurements," in *Proc. Conf. ACM Special Interest Group Data Commun.*, Aug. 2018, pp. 561–575.

[29] Z. Zhao, X. Shi, X. Yin, Z. Wang, and Q. Li, "HashFlow for better flow record collection," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 1416–1425, doi: 10.1109/ICDCS.2019.00141.

[30] L. Tang, Q. Huang, and P. P. C. Lee, "A fast and compact invertible sketch for network-wide heavy flow detection," *IEEE/ACM Trans. Netw.*, vol. 28, no. 5, pp. 2350–2363, Oct. 2020, doi: 10.1109/TNET.2020.3011798.

[31] Q. Zhao, Z. Ge, J. Wang, and J. Xu, "Robust traffic matrix estimation with imperfect information: Making use of multiple data sources," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 34, no. 1, pp. 133–144, Jun. 2006, doi: 10.1145/1140103.1140294.

[32] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale IP traffic matrices from link loads," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 31, no. 1, pp. 206–217, Jun. 2003, doi: 10.1145/885651.781053.

[33] M. T. Goodrich and M. Mitzenmacher, "Invertible Bloom lookup tables," 2011, *arXiv:1101.2245*.

[34] G. Cormode and S. Muthukrishnan, "An improved data stream summary: The count-min sketch and its applications," *J. Algorithms*, vol. 55, no. 1, pp. 58–75, Apr. 2005, doi: 10.1016/j.jalgor.2003.12.001.

[35] C. Filsfils, *Segment Routing Architecture*, document IETF RFC8402, Jul. 2018.

[36] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An approach to universal topology generation," in *Proc. MASCOTS, Proc. 9th Int. Symp. Modeling, Anal. Simulation Comput. Telecommun. Syst.*, Aug. 2001, pp. 346–353.

[37] D. T. Hai, "Photonic network coding and partial protection in optical-processing-enabled network: Two for a tango," *Opt. Quantum Electron.*, vol. 54, p. 282, Apr. 2022, doi: 10.1007/s11082-022-03628-5.

SHOHEI KAMAMURA (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Waseda University, Tokyo, Japan, in 2004, 2006, and 2013, respectively. He joined the Nippon Telegraph and Telephone Corporation (NTT), in 2006. From 2017 to 2020, he was engaged in the commercial development of SDN controller and speech recognition for web conferencing at NTT Communications. He has been an Associate Professor at the Department of Computer and Information Science, Seikei University, since 2021. Since then, his research interests have included IP and optical network design, control, optimization, and network data mining. He is a member of The Institute of Electronics, Information and Communication Engineers (IEICE) of Japan. He received the Young Engineer Award from IEICE, in 2010, the ICM English Session Encouragement Award from IEICE, in 2010, the Best Paper Awards of ICC from IEEE, in 2014, the TELECOM System Technology Award from the Telecommunication Advancement Foundation, in 2017, and the Distinguished Contributions Award from IEICE, in 2017.