IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 13, 2020, accepted May 2, 2020, date of publication May 11, 2020, date of current version May 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2993556

An Effective Hierarchical Control Plane for Software-Defined Networks Leveraging TOPSIS for End-to-End QoS Class-Mapping

JEHAD ALI, (Graduate Student Member, IEEE), AND BYEONG-HEE ROH, (Senior Member, IEEE)
Department of Computer Engineering, Ajou University, Suwon 16499, South Korea

Corresponding author: Byeong-Hee Roh (bhroh@ajou.ac.kr)

This was supported by the Ministry of Science and ICT (MSIT), South Korea, through the Information Technology Research Center (ITRC) Support Program, supervised by the Institute for Information and Communications Technology Planning and Evaluation, under Grant IITP-2020-2018-0-01431.

ABSTRACT If an end-to-end (E2E) path includes multiple domains, we need inter-domain collaboration to ensure the E2E quality-of-service (QoS) for the applications. In heterogeneous networks, an E2E path may go through domains with several QoS classes in each domain. However, the prevalent legacy network architecture and the standard software-defined networking (SDN) model lack effective mechanisms for inter-domain collaboration and QoS class mapping. In this study, we propose a hierarchical SDN control plane approach to guarantee the E2E QoS among multiple domains with various QoS classes on the E2E path. We propose a controller module for selecting the most suitable QoS class for each domain in the E2E path based on multi-criteria decision-making by using the technique for order of preference by similarity to ideal solution (TOPSIS). We map the suitable service classes in the global controller (GC) for provisioning the E2E QoS according to the application service requests. First, we propose an SDN-based inter-domain communication scheme and the message processing algorithm for E2E service delivery when multiple QoS classes exist in each domain. Next, we formulate the problem of service class selection with TOPSIS, provide an E2E mapping scheme, and demonstrate it with an example. Finally, we compare the proposed approach with the existing schemes for E2E QoS class mapping in terms of E2E delay, jitter, packet loss rate (PLR), and cost (per bandwidth unit). According to our simulation results, the proposed approach ensures the E2E QoS and guarantees the E2E delay, jitter, PLR, and cost according to the application service requests.

INDEX TERMS Quality of service, software-defined networking, TOPSIS.

I. INTRODUCTION

The rapid developments in Internet technology have resulted in the ubiquitous proliferation of network terminals. However, the traditional network architecture was not adapted to advancements in future communication and Internet technologies, which has resulted in heterogeneous networks. Therefore, the legacy network infrastructure could not maintain a constant pace with radical changes in the Internet. The main feature of the traditional network architecture is that the data and control planes are tightly coupled, which leads to several limitations: for example, if we want to change the configuration of the network, we should configure each device independently throughout the entire network. Similarly, vendors are hesitant in providing the internal details of their devices to the developers and users, because changes in the configuration of traditional networking devices can lead to a malfunction in the networks. Besides, the protocols are strongly embedded in the firmware of the network devices. These restrictions hamper innovation in the networks due to proprietary hardware and lack of testing for innovative networking solutions. This increases the administrative workload and the overall cost of network management.

On the contrary, the software-defined networking (SDN) paradigm [1]–[3] has revolutionized the management of networks by decoupling data and control planes. The separation of data and control planes has moved the network complexity from networking devices to the intelligent SDN controllers.

The associate editor coordinating the review of this manuscript and approving it for publication was Yulei Wu.

Thus, the devices can be programmed through applications running on the controller, and the underlying network is abstracted from the applications [4]. The innovative concept introduced by SDN has big advantages: flexible and efficient configuration, management, and operation of the networks. Therefore, SDN is predicted to be an excellent choice for future communication and Internet technologies. Due to these advantages, large information technology organizations such as Amazon, Facebook, and Google have implemented SDN for connecting their remote data centers [5], simultaneously contributing to the development of OpenFlow [6] switches for the standardization of SDN protocols and architecture.

The SDN has been deployed successfully in the data centers, enterprise networks, and traffic engineering among the remote data centers. However, the adoption of SDN in the modern and global Internet still poses several challenges that need to be investigated further. One of the main issues is the end-to-end (E2E) quality-of-service (QoS) fulfillment through heterogeneous network providers. In an enterprise network, the SDN controller, user applications, and forwarding devices are administered by the same provider. Therefore, the network status of the underlying devices is easily available to the upper-layer applications. However, the Internet is composed of different inter-providers where end-users, applications, and service providers are often heterogeneous. As a result, the network status is not directly available to the applications running on the upper layers.

Furthermore, an E2E path that traverses multiple domains may belong to different providers. In this case, to guarantee the E2E QoS, there is a need for an abstraction of the upper-level network. Moreover, to guarantee smooth communication between the users and different service providers, a strong collaboration is required among the involved domains. Hence, there is a need to explore mechanisms of effective interaction between the users and the service providers, as well as collaboration among these domains.

The separation of the data and control planes has escalated the importance of the SDN controller because of the shifting of the intelligence from the forwarding devices to a central point. Many controllers [7]–[12] are developed by different vendors. However, no common standard exists to ensure interoperability among these controllers. Therefore, centralized controller management will lead to issues with interoperability and E2E QoS fulfillment and may not apply to multiple providers. Accordingly, multiple domains with heterogeneous network providers would be unable to operate under the same SDN controller. Hence, the E2E service provision can be suboptimal for the applications running across different domains.

Moreover, each domain on the E2E path offers different QoS classes due to heterogeneous networks. Because there is no globally defined service class, collaboration among multiple domains for seamless E2E service delivery and the mapping of E2E QoS classes according to application service requests needs further investigation. To obtain a global view of the E2E domains, multiple domains must be administered by several controllers, which must collaborate for sharing the underlying network status. Thus, the SDN architecture with a global view of the inter-domain scenario needs to be further explored for ensuring E2E QoS in the presence of multiple service classes.[1] In this study, our motivation is to propose the SDN-based approach for E2E service class mapping with a global control of the QoS metrics, provisioning the E2E services delivery when multiple providers exist on the E2E path. As compared with previous work, the contributions of this study are as follows:

- We propose a packet processing algorithm with a hierarchical control plane solution leveraging SDN for inter-domain collaboration and E2E service provisioning for heterogeneous network providers.
- We propose a network model for service class selection in the SDN-based heterogeneous network environment by proposing a technique for order of preference by similarity to ideal solution (TOPSIS) module for the SDN controller and a mapping mechanism for QoS classes.
- We perform a demonstration and simulations for the SDN-based approach and compare the results with the previous schemes. Our method improves E2E performance for various performance metrics.
- To check the effectiveness of the proposed approach based on a TOPSIS module, the simulations are also performed for the SDN-based approach by removing the TOPSIS module from it.
- The simulations demonstrate effect of the cost per bandwidth unit on the E2E performance.

The remainder of this paper is organized as follows. In Section II, we discuss related work and challenges in the traditional and SDN-based approaches for E2E QoS class-mapping and service provisioning. The proposed SDN-based approach, a packet processing algorithm in the hierarchical control plane, the network model for E2E service class selection, and a step-by-step procedure for class selection and mapping based on the TOPSIS module is discussed in Section III (A)–(E). In Subsection III (F), we demonstrate the proposed approach with an example. Section IV evaluates the results and compares them with previous schemes. In Subsection IV (F), we show the effect of the cost weight on the E2E performance. Moreover, we evaluate the E2E cost (per bandwidth unit) in subsection IV (G) and compare it with previous schemes. Finally, Section V concludes the paper and discusses future work.

II. LITERATURE REVIEW

The E2E QoS provisioning in SDN is significant for seamless communication between the source and destination nodes. Because of a lack of a common northbound application programming interface (NB API), multiple heterogeneous network domains require collaboration, and service classes

[1] We use the terms ''service classes'' and ''QoS classes'' interchangeably.

that exist on the E2E path need mapping to guarantee QoS. Neither of the two mentioned topics is regarded as a trivial issue in the literature. Therefore, in this study, we focus on the inter-domain coordination and QoS class mapping on the E2E path leveraging SDN. In this section, we describe the existing studies that provide different schemes for collaboration among domains and QoS class mapping when multiple providers exist on the E2E path. First, we discuss such schemes and their issues in traditional networks. Then, we describe SDN-based studies.

The inter-domain collaboration on the E2E path is critical because of different QoS architectures in each domain and heterogeneity of the service providers. Previously, the bilateral model [13] was widely followed for QoS negotiation between inter-providers. However, in this model, the service performance negotiation happens only between the neighboring providers with no global view of the E2E domains for service delivery. Consequently, the model is not capable of providing the E2E service provisioning and is not scalable. Furthermore, the mapping of the service classes is a challenging task with respect to meeting the E2E service needs due to the local view of the network state information in the domains.

The QoS in the inter-provider faces many challenges [14]. A comprehensive in-depth analysis of inter-provider issues is given in [13]. Herein, we further investigate various approaches to address these issues. In the heterogeneous networks, the QoS architecture for each domain is different; therefore, the E2E path establishment would be suboptimal. Besides, the E2E path consists of various QoS classes. Therefore, provisioning the QoS on the E2E level is cumbersome for E2E application service requests. Several approaches have been presented owing to the limitations of the bilateral scheme. A solution based on the cooperation model was prevalent for content distributors [15], [16]. However, it does not apply to heterogeneous networks, because the service classes are not globally unique, whereas the applications need an E2E service guarantee. Hence, the QoS class mapping has been explored by several researchers due to its significance for the QoS guarantee on the E2E path passing through multiple domains with several service providers. A generic framework for service specification in [17] provides a description of service classes in network domains. Then, a mapping technique is applied for the selection of the unique E2E service class.

Several approaches exist for inter-domain class mapping on the E2E level in the traditional networking architecture [18]–[20]. In [18], the authors proposed a policy-dependent conformance matching scheme (P-CMS) as an extension of CMS using a third-party (3P) approach. A 3P agent in [21] is partially responsible for monitoring service-level agreements (SLAs) among different domains. The CMS maps the service classes from the source to the destination in heterogeneous domains. However, this approach does not ensure the E2E QoS of the applications and cannot adapt to the changing parameters. Moreover, the P-CMS has no global view of the service metrics and needs a tight collaboration among the domains. Additionally, this solution is not cost-aware because it only considers the technical performance metrics[2] such as delay, jitter, and packet loss rate (PLR).

The problem of service class selection was also formulated using a 3P integer programming (3PIP) approach [19]. A multi-constraint technique for mapping of service classes is introduced. However, it fails when the constraint limitation is not met. The tight collaboration among multiple domains of different providers is critical for QoS provisioning on the E2E path. To achieve this objective, authors in [20] proposed a goal programming (GP) approach deployed on a 3P agent for class mapping. As discussed in [21], the 3P scheme is a significant solution for QoS delivery among different providers. It meets various conflicting goals and constraints. However, the interaction between the domains and the third-party server, as well as the details of the message flows, are not given. Furthermore, it is based on traditional networking architecture and cannot meet certain objectives. Inspired by the 3P agent, its analogy with the central SDN controller, and the wide adoption of SDN in future technologies, we further explore the SDN for E2E service provisioning and QoS class mapping.

A comprehensive survey for inter-domain SDN is presented in [22] which discusses the applications, challenges, and evolution in the inter-domain SDN. First, the study discusses a typical approach to inter-domain communication mechanisms such as a West-East Bridge (WE-Bridge) i.e. a proposal made by Tsinghua University for inter-domain communication in SDN. Then, a mechanism is described for the establishment of connections, exchange of information, message format, the process of negotiation and the virtual network view in SDN domains. Finally, the applications and challenges are discussed.

A representative study in [23] proposed a framework based on the network as a service (NaaS) for E2E QoS provisioning in SDN. The SDN controllers have no common standard interface for interoperability among controllers of heterogeneous domains. Therefore, a service delivery platform (SDP) is proposed. Further, the scheme illustrates an analytical model for network abstraction and resource assignment. This study emphasizes the bandwidth utilization enhancement for the E2E QoS guarantee. Additionally, the numerical and analytical results show that the scheme simplifies resource management and improves bandwidth utilization considering the E2E QoS provisioning. Similarly, the study in [24] has also proposed the SDN-based approach for enhancing the utilization of network resources. Apart from enhancing the bandwidth utilization, the paper also describes network failure recovery in a multi-domain scenario. However, these studies failed to consider different QoS classes offered in each domain from the source to the destination on the E2E path.

[2] We use the terms "performance metrics" and "QoS metrics" interchangeably.

In literature [25] the E2E QoS in SDN has been investigated while focusing on domain privacy. A privacy-aware framework is proposed in the inter-domain environment where the network statistics are processed in a distributed manner and then a single point collects the QoS request for making a binary decision. Further, to quantify achieved privacy a privacy index metric is proposed in the inter-domain SDN. The results show that the proposed adaptive learning technique provides 1.4x more privacy. However, the study only focus on the privacy for inter-domain SDN frameworks.

The study in [26] provided a resource reservation framework for E2E QoS fulfillment for delay bound applications. Moreover, the significance of E2E QoS in various applications (such as smart grids, healthcare, education, mobility, gaming, and manufacturing) is discussed. Further, the study discusses the mechanism for inter-domain interaction, resource reservation concept, and admission control in SDN. The simulation shows the performance results for control-plane scalability. However, it does not provide any details on QoS service classes and the communication design among the domains. Moreover, the proposed architecture follows the flat structure for inter-domain communication; therefore, it lacks global control over the performance metrics.

The authors in [27] introduced a hierarchical SDN method with the aim to reduce the flow request messages to a controller and improve the control plane scalability. The architecture is scalable irrespective of the number of connections of the network. The hierarchical architecture consists of a broker controller that acts as a coordinator between the lower-level controllers and the QoS-based routing procedure among the autonomous systems. However, the architecture needs further investigation in the case of heterogeneous network providers with multiple service classes on the E2E path from source to destination.

These previous studies were based on the traditional distributed Internet architecture. However, the distributed algorithms use border gateway protocol (BGP) [28], resource reservation protocol (RSVP) [29], and BGP link-state (LS) [30] signaling protocols. However the following political and technical aspects do not recommend the use of BGP and traditional distributing routing protocols.

- The BGP does not consider the resistances existing between the vendors and inter-providers.
- Limited length of BGP policy compliant paths among countries [31].
- The BGP has a slow convergence [32] time because its execution model is local.
- The BGP performs cross-domain routing in a decentralized way without knowledge of E2E paths, fails to provide a globally optimal path intended for E2E QoS provisioning.
- The BGP oscillation is persistent. Therefore, it does not guarantee a shortest path [33].
- The literature in [23] argues that BGP is not suitable for inter-domain SDN routing and suggests the decoupling of policy and routing control for interoperability among domains in SDN.
- Additionally, the traditional network vendors do not provide the granularity of device configurations, and the protocols are firmly embedded in the firmware.

Therefore, deploying a new service on the E2E level is cumbersome, inflexible, and complex. Moreover, the SDN-based schemes have not described the QoS class-mapping for the E2E QoS guarantee. Further, the architectures are based on the flat structure, which does not describe the global view of the inter-domains for E2E service class mapping.

SDN is considered as the technology for future Internet because of its potential benefits of (1) decoupling the control plane from the data plane (i.e., centralized management) and (2) E2E QoS guarantee according to the diverse demands of different applications. Several studies have explored the first aspect. However, leveraging the SDN paradigm in the multi-domain context with several service providers and the E2E QoS guarantee has not been investigated well, specifically, with the existence of QoS classes on the E2E path. To the best of our knowledge, our proposed SDN-based approach can provide a global view for multiple source-to-destination inter-provider domains and perform class mapping on the E2E level.

III. PROPOSED APPROACH

Our first objective is to provide an architecture for collaboration among domains that provides a global view of the E2E network. Therefore, in the first part, we are discussing our proposed architecture, the packet processing algorithm, and modules of the controller for achieving this objective. Second, we aim to select the most suitable[3] service class among a set of service classes on the E2E path and to perform a mapping of these service classes for defining a globally defined class for an E2E service provisioning. The following subsections illustrate this in detail.

A. ARCHITECTURE FOR E2E SERVICE DELIVERY

The inter-domain environment is a complex network with various components. Therefore, a suitable network model having provision for state-of-the-art technology and well-defined functional units is needed for its management, control, and design. The proposed architecture for E2E SLAs between different service providers is based on SDN. Our proposed architecture leverages the deployment of the centralized modules, deployment of new applications, and flexible interaction between forwarding devices and the SDN controller through southbound (SB) API.

Herein, we recall two types of multi-controller architectures to apply our TOPSIS module for computing the unique E2E QoS class in the presence of inter-domain heterogeneous providers. We call them the flat controller architecture (FCA) and the hierarchical controller architecture,

[3] We use the terms "suitable service class" and "ideal service class" interchangeably.

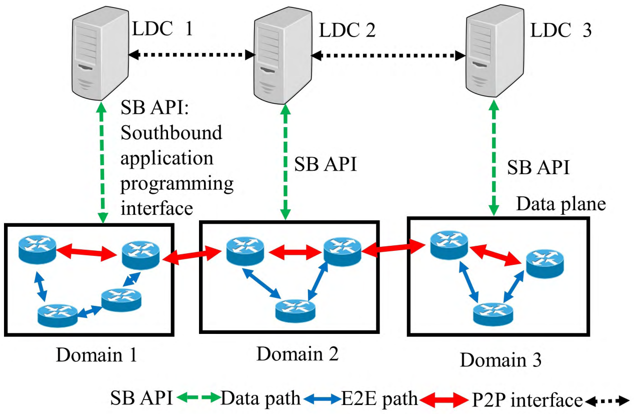


FIGURE 1. Flat controller architecture (FCA) in SDN.

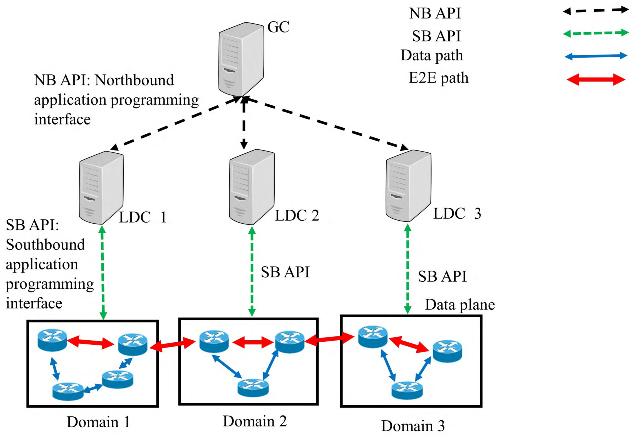

GC
NB API
SB API
Data path
E2E path
NB API: Northbound application programming interface
LDC 1
LDC 2
LDC 3
SB API: Southbound application programming interface
SB API
SB API
Data plane
Domain 1
Domain 2
Domain 3

FIGURE 2. Hierarchical controller architecture in SDN.

as shown in Figures 1 and 2, respectively. We investigated these two controller architectures for E2E service class mapping. In the FCA, the local domain controllers (LDCs) interact in a peer-to-peer (P2P) manner. However, FCA has three main problems. First, the scalability is limited as the network size grows; it worsens the processing latency of the controller [34]. Second, the lack of a common NB API restricts the interoperability among multiple vendors of controllers [23]. Third, there is no architecture to find global E2E unique service class calculation due to P2P collaboration among the domains. Further in HCA the traffic to the controller reduces 50% than the FCA [27].

Our proposed approach is based on SDN. Hence, it overcomes the issues in the BGP and the traditional distributing protocols such as the proprietry hardware, inter-operability, centralized managment which ensures the shortest path, cross-domain routing in a centralized manner and mainly the innovation in testing new technologies. Further, our proposed method leverages hierarchical architecture [35]–[37]. The hierarchical architecture of SDN controllers integrates autonomous domains with a hierarchical association. Multiple domains are integrated with a hierarchical controller architecture, where LDCs coordinate through a GC. By applying the hierarchical architecture, new services can be easily managed and deployed in the domains coexisting on the E2E path between the source and the destination [38] nodes.

The tasks handled by a controller are propagated from lower layer LDCs to the upper layer GC which decreases the computational complexity. The hierarchy control plane with a global view decreases the E2E delay as the network scales [39]. The GC in the proposed architecture acts proactively for the establishment of the E2E path, therefore, the flow setup delay (the delay for path establishment and pushing the flow entries into the switches) [40] is reduced. The hierarchical architecture enables communication among multiple LDCs with assorted equipment. The effectiveness of the hierarchical control plane for effective collaboration among heterogeneous tactical networks with a guaranteed QoS has been proved in [41].

The proposed architecture is shown in Figure 2. There are LDCs and a GC. The switches in the domains are connected with LDCs on the E2E path. The GC obtains the underlying network status dynamically from the LDCs administering the domains; consequently, it has access to the global topology. Thus, the GC provisions resources from the LDCs upon the arrival of a service request. The LDCs collaborate through GC, and SLAs are exchanged through it. Each LDC is equipped with a traffic flow template (TFT) module [42], which gathers the source and destination port numbers, Internet Protocol (IP) addresses, and QoS parameters. The collected data will be used by the E2E path computation (PCM) [43] and TOPSIS modules to find the E2E path and the ideal service class among a set of service classes on the E2E path. Finally, a GC will find the globally unique service class for an E2E service provisioning.

B. PACKET PROCESSING ALGORITHM

Figure 3 shows different modules for gathering network status, inter-domain collaboration, E2E path computation, and QoS class mapping. Each LDC runs a TFT module to obtain the service requests from the domains through its SB API protocol, i.e., OpenFlow [6], a PCM to compute a path, and a TOPSIS module to select a QoS class in the E2E path. The GC proactively gathers the path and service class information and configure the rest of LDCs. Thus, GC finds the E2E path and unique E2E service class. On the arrival of a new

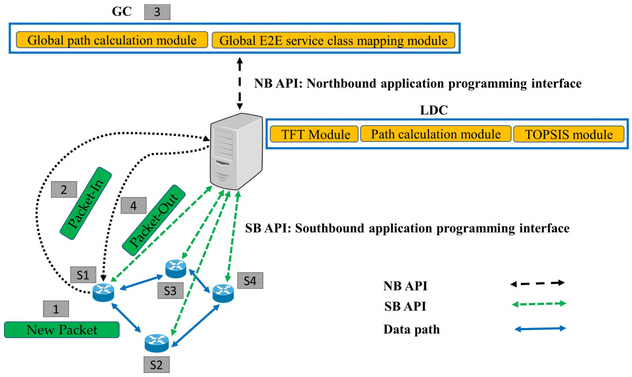

GC
3
Global path calculation module
Global E2E service class mapping module
NB API: Northbound application programming interface
LDC
TFT Module
Path calculation module
TOPSIS module
2
Packet-In
4
Packet-Out
SB API: Southbound application programming interface
S1
S3
S4
1
New Packet
S2
NB API
SB API
Data path

FIGURE 3. Packet processing in the proposed architecture.

packet, the GC maps the global service class from the source to the destination on the E2E path from the service classes computed by the TOPSIS modules of each LDC. The E2E SLAs across multiple domains is provided by the GC.

Algorithm 1 Packet Processing in the Network

```
A new packet ($(NP)$) arrives on the ingress switch ($(S_1)$) in the outermost domain.
Check for the ($(FR)$) in ($(S_1 - CBMQs)$). i.e. ($(FR)$) comparison according to the ($(NPH)$)
**If**
    The ($(NPH)$) does not match the ($(FR)$) of ($(S_1 - CBMQs)$), **then**
    A $Packet - In$ ($(PIN)$) message is forwarded to the ($(LDC)$), which forwards it to ($(GC)$).
    The ($(GC)$) proactively calculates the E2E path ($(P_i)$) and QoS class ($(C_i)$) using PCM and TOPSIS modules from the rest of LDCs.
    The ($(GC)$) performs the mapping according to the ($(NPH)$) service requests.
    Once, the ($(GC)$) mapping is completed, the ($(GC)$) returns the ($(P_i)$) and E2E unique service class and pushes the ($(FR)$) to the ($(S_1)$) of the outermost domain as a $Packet - Out$ ($(POUT)$) message via the ($(LDCs)$).
    The ($(FR)$) gets updated in the ingress switch ($(S_1 - CBMQs)$).
**Else**
    The ($(NPH)$) matches the ($(FR)$) on ($(S_1 - CBMQs)$) of the outermost domain, **then**
    The ($(EFR)$) in the ($(S_1 - CBMQs)$) are followed for forwarding the ($(NP)$).
```

Algorithm 1 demonstrates the packet processing in the hierarchical architecture upon the arrival of the new packet (NP). When the (NP) arrives on the ingress switch (S_1) of the outermost domain, the new packet header (NPH) is compared with the flow rules (FR) in the switch (S_1) class-based mapping queues $(CBMQs)$ i.e. $(S_1 - CBMQs)$. The (FR) for the service classes and the mapping information is implemented as the $(CBMQs)$ in the SDN switches [44]. If the (FR) on the $(S_1 - CBMQs)$ does not match with the (NPH), then, a $Packet - In$ (PIN) message is sent to the (LDC). The LDC forwards the request to the (GC). In the meanwhile, the (GC) proactively configures the E2E path (P_i) using PCM module and QoS classes (C_i) leveraging TOPSIS modules from the LDCs. In each LDC (k), the TFT module gathers the source and destination IP addresses, port numbers, and QoS requests via the SB API protocol which is utilized by the TOPSIS module. As the (GC) has a global view of the whole network and QoS classes from the TOPSIS modules, it maps it per application service requests for the E2E path (P_i). The mapping is performed with the $(CBMQs)$ on the switches, where the (FR) for the service classes are defined. Further, Equation (11) and Equation (12) shows how the service classes are mapped and computed for the (P_i). On completion of the mapping, the (GC) returns the (P_i) and E2E unique service class to the ingress switch (S_1) of the outermost domain as a $Packet - Out$ $(POUT)$ message via the $(LDCs)$. The (FR) are updated on the $(S_1 - CBMQs)$. On the contrary, if the (NPH) matches the (FR) on the $(S_1 - CBMQs)$ of the outermost domain, then, the (NP) is forwarded according to the existing flow rules (EFR) of the $(S_1 - CBMQs)$.

C. NETWORK MODEL FOR E2E SERVICE CLASS SELECTION

In this section, we describe how to select the ideal QoS class in each LDC on the E2E path with the proposed SDN architecture. A QoS class is defined as a collection of performance parameters such as delay, jitter, PLR, and cost on the E2E path in a domain with multiple inter-providers. Herein, the cost refers to cumulative costs for traffic management in terms of generation, sending, and termination. Several classes consider different QoS metrics such as delay, jitter, PLR, and cost from several providers. Therefore, QoS class selection on the E2E path can be modeled as a multi-criteria decision making (MCDM) problem. The following key problems need to be solved. First, we need to compute the most desirable QoS classes on the E2E path passing through each LDC. Second, we need to find a globally defined service class for the whole E2E path.

An SDN is represented as $G = (V, E)$, where V is a set of nodes and E is a set of links. $S = \{s_1, s_2, \ldots, s_m\}$ are the switches of the network. $K = \{k_1, k_2, \ldots, k_n\}$ are the LDCs deployed in the network, managing the domains from the source to the destination, i.e., $D = \{d_1, d_2, \ldots, d_n\}$, respectively. We assume that the LDCs are already deployed in the network [45]. The control plane layers are denoted by $L = \{L_1, L_2\}$ because we have LDCs and GC i.e. two layers control plane. The topology is partitioned into n domains, and each domain has an LDC: $D = \{d_1, d_2, \ldots, d_n\}$ are administered by $K = \{k_1, k_2, \ldots, k_n\}$, respectively. Further, there is a GC such that all the LDCs can share the status with the GC (Figure 2). The QoS classes in a domain on the path are denoted by i. The QoS parameters are denoted by j. The symbolic notations used in the method of the QoS class selection for a domain on the E2E path are defined in Table 1.

D. TOPSIS MODULE

TOPSIS is a multi-criteria decision-making (MCDM) scheme [46], [47] widely used in diverse fields [48]–[50] for MCDM problems. The effectiveness of the TOPSIS over other approaches is due to;

- Its programmable and logical behavior;
- Suitability in handling wide range of criteria and alternative metrics;
- Comparative consistency in ranking alternatives;
- Its ability in dealing with constrained subjective inputs [51], [52] and
- We can prioritize the preferences of the network users through assigning weights to the performance metrics.

TABLE 1. Summary of notations.

Notation	Explanation
V	Set of nodes
E	Set of edges
$L = \{L_1, L_2\}$	Control plane layers
M	Number of switches
N	Number of domains & LDCs
$D = \{d_n\}$	Set of domains
$S = \{s_m\}$	Set of switches
$K = \{k_n\}$	Set of LDCs
i	Service class in a domain
j	QoS performance metric in a domain
J_1	Benefit QoS performance metric
J_2	Cost QoS performance metric
$d_n^k = \{1, 0\}$	Binary variable: A value of 1 shows that k is the administrative LDC for domain d_n; otherwise, d_n^k=0.
$y_i^k = \{1, 0\}$	Binary variable: y_i^k=1 shows that service class i is selected by LDC k; otherwise, y_i^k=0.
$u_n^k = \{1, 0\}$	Binary variable: u_n^k=1 shows that domain n is administered by LDC k in the E2E path; otherwise, u_n^k=0
$A_{j,k}^*$	Positive ideal solution in the k^{th} LDC for a domain
$A_{j,k}^-$	Negative ideal solution in the k^{th} LDC for a domain
$C_{i,k}^*$	Weight of the service class i in the k^{th} LDC for a domain

The effective use of TOPSIS in heterogeneous networks has been described in [53]. Herein, we will introduce TOPSIS as an SDN controller module in the LDC for service class selection in a domain. There are two ideal solutions proposed by this technique: positive and negative. The most and least desirable service classes for a given domain in the E2E path are selected by the positive and negative ideal solutions, respectively. Using a TOPSIS module, the LDC selects the most desirable service class that has a minimum distance from the positive and maximum distance from the negative ideal solutions, respectively. The selected service classes through the TOPSIS module employed on the LDC will be used by the GC for the unique E2E service class computation on the E2E path.

Let us describe a step-by-step approach for applying TOPSIS in an LDC. A decision matrix Q_k representing the alternatives (QoS classes) and criteria (QoS performance metrics) is formed for a domain administered by LDC k in the E2E path. The general form of the decision matrix is given in (1). The service classes are represented by $i = 1$ to $i = m$ and the QoS metrics are denoted by $j = 1$ to $j = n$. Herein, $j = 1, j = 2, j = 3$, and $j = 4$ denote the delay, jitter, PLR, and cost. We are using the words "QoS metric" and "QoS performance metric" interchangeably. The row values show the service metric values for a QoS class of LDC k. Next, $q_{(i,j)}$ is the value of the service metric in the LDC k corresponding to the i^{th} service class and j^{th} QoS parameter.

$$Q_k = \begin{bmatrix} q_{11} & q_{12} & q_{13} & \cdots & q_{1n} \\ q_{21} & q_{22} & q_{23} & \cdots & q_{2n} \\ q_{31} & q_{32} & q_{33} & \cdots & q_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ q_{m1} & q_{m2} & q_{m3} & \cdots & q_{mn} \end{bmatrix} \tag{1}$$

The decision matrix is normalized by (2) using the Euclidean normalization [54]. A normalized matrix R_k is obtained for LDC k, where r_{ij} is the normalized value of the service metric corresponding to the i^{th} row and j^{th} column. These normalized values are computed for $i = 1, 2, 3, \ldots, m$ service classes and $j = 1, 2, 3, \ldots, n$ service metrics.

$$R_k = \begin{bmatrix} r_{11} & r_{12} & r_{13} & \cdots & r_{1n} \\ r_{21} & r_{22} & r_{23} & \cdots & r_{2n} \\ r_{31} & r_{32} & r_{33} & \cdots & r_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & r_{m3} & \cdots & r_{mn} \end{bmatrix} \tag{2}$$

$$r_{ij} = \frac{q_{ij}}{\sqrt{\sum_{i=1}^{m} (q_{ij})^2}} \tag{3}$$

Next, the normalized matrix is converted into a weighted decision matrix F_k by (4). Each value f_{ij} in this matrix is calculated by (5). Here, W_i denotes a weight factor for the service metric in a QoS class.

$$F_k = \begin{bmatrix} f_{11} & f_{12} & f_{13} & \cdots & f_{1n} \\ f_{21} & f_{22} & f_{23} & \cdots & f_{2n} \\ f_{31} & f_{32} & f_{33} & \cdots & f_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f_{m1} & f_{m2} & f_{m3} & \cdots & f_{mn} \end{bmatrix} \tag{4}$$

$$f_{ij} = W_i \; * r_{ij} \tag{5}$$

Then, we compute the positive and negative ideal solutions for the QoS classes of LDC k. Equations (6) and (7) show the positive ($A_{j,k}^*$) and negative ($A_{j,k}^-$) ideal solutions for the QoS classes in domain k on the E2E path. Here, J_1 and J_2 are the benefit and cost metrics in the E2E QoS classes. These equations show our objective functions: to maximize the benefit QoS metrics and minimize the cost performance metrics in a service class on the E2E path. In our case, we aim to minimize the following performance metrics on the E2E path passing from the source domain to the destination domain: delay, jitter, cost, and PLR. Hence, a smaller value for these performance metrics will be selected in the service classes.

$$A_{j,k}^* = \left\{ maxF_{ij}, \quad j \in J_1, \quad MinF_{ij}, \quad j \in J_2 \right\} \tag{6}$$

$$A_{j,k}^- = \left\{ minF_{ij}, \quad j \in J_1, maxF_{ij}, \quad j \in J_2 \right\} \tag{7}$$

The similarity index is calculated by (8) and (9) toward the positive and negative ideal solutions. These equations quantify the Euclidean distance (ED) from the positive ideal and negative ideal service classes (i.e., the most and least desirable QoS classes). Variables ($D_{i,k}^*$) and ($D_{i,k}^-$) show the best and worse values, respectively, of the E2E service computed by the TOPSIS module deployed in LDC k.

$$D_{i,k}^* = \sqrt{\sum_{j=1}^{n} (f_{ij} - A_j^*)^2} \tag{8}$$

$$D_{i,k}^- = \sqrt{\sum_{j=1}^{n} (f_{ij} - A_j^-)^2} \tag{9}$$

Finally, we obtain weights of the candidate alternatives (QoS classes) using (10) by the TOPSIS module in LDC k. Furthermore, these alternatives are ranked according to their final weights obtained using (10). The alternative with the highest weight is selected by LDC k as the ideal service class for the domain in the E2E path.

$$C^*_{i,k} = \frac{D_i^-}{D_i^* + D_i^-} \tag{10}$$

Thus, the service classes with the highest weight would be selected from each domain d by the TOPSIS modules of each LDC in the E2E path. These values are shared with the GC. The GC has a global view of the service classes from all LDCs. Thus, GC computes the unique service class for the whole E2E path. Further, it performs a mapping of these service classes per application service requests.

E. GLOBAL E2E SERVICE CLASS MAPPING

The GC checks for the degree of correspondence (DC) [55] ratio of the offered and required service classes for an application service request (i.e., whether the service classes offered in the domains on the E2E path meet the service requirements of an application). Therefore, a global view of the E2E QoS service classes is shared with the LDCs. Each LDC updates the information for the forwarding devices of the corresponding domain. To verify whether the E2E QoS classes satisfy the service requirements of an application, we calculate the E2E DC^{E2E} ratio. The formula for calculating DC^{E2E} is shown in (11). In this equation, $AS_{j,req}$ and $AS_{j,off}$ represent the required and the offered service class values on the E2E path passing from the source domain to the destination domain for a service metric j.

The DC_j^{E2E} ratio denotes whether the offered service classes on the E2E path computed by the TOPSIS module guarantee the service class requests of an application for a service metric j. For example, $DC_{j=1}^{E2E} = 1$ means that the ratio of the QoS class requested by an application has a perfect match with the QoS classes offered in each domain on the E2E path for the performance metric delay. Herein, $j = 1, 2, 3, 4$ are the performance metrics of delay, jitter, PLR, and cost, respectively. If DC_j^{E2E}>1, it indicates that the offered service classes ($AS_{j,off}$) provide better service class values than the service class values required for an application ($AS_{j,req}$). In contrast, if DC_j^{E2E}< 1, the offered service classes in the E2E path returned by the TOPSIS modules from E2E LDCs do not satisfy the required service class requests for an application.

$$DC_j^{E2E} = \frac{AS_{j,req}}{AS_{j,off}} \tag{11}$$

Assume that the E2E domains are managed by n LDCs such that $K = \{k_1, k_2, k_3, \ldots, k_n\}$. The offered service classes on the E2E path are summed by (12). We introduce two binary variables: $u_n^k = \{1, 0\}$ for a domain n under LDC k on the path and $y_i^k = \{1, 0\}$ for class selection in a domain i.e., $u_n^k = \{1\}$ if a domain n in the E2E path is selected under LDC k and $y_i^k = \{1\}$ if a service class i is selected by LDC k in a domain; otherwise, $y_i^k = \{0\}$. Let $S^k{}_{j,i}$ be the value of performance metric j in LDC k for a service class i computed by the TOPSIS module. Then, $AS_{j,off}$ will be determined by the application according to its service requirements.

$$AS_{j,off} = \sum_{k=1}^{n} \sum_{i=1}^{m} y_i^k \times u_n^k \times S^k{}_{j,i} \tag{12}$$

F. DEMONSTRATION OF THE PROPOSED APPROACH

In this section, we demonstrate the proposed approach for QoS class mapping. Figure 4 shows an E2E view of five domains with five LDCs and a GC. The path from the source to the destination (returned by PCM) has five autonomous domains (d_1, d_2, d_3, d_4, and d_5) administered by five LDCs (k_1, k_2, k_3, k_4, and k_5). On the arrival of a service request, Algorithm 1 performs message processing. On the E2E path, each domain has a set of service classes ($i = 1, 2, 3, \ldots, m$). The TOPSIS module in each LDC finds the service class among available classes in each domain and shares it with the GC. The GC obtains the status of each LDC proactively and performs class mapping and SLA's among LDCs. H_1 and H_2 are the source and destination hosts. Tables 2 and 3 show the E2E the service requests and the service classes offered on the E2E path in five domains. Specifically, Table 3 shows the offered service class values in each domain on the E2E path, which are taken from the specifications in [19]. Herein, the cost is represented in monetary units (MU), which is the cost for the bandwidth transfer unit within the domain. Table 2 shows the service requests for the E2E path. These QoS values have been adopted from the standard requirements for the service requests [56].

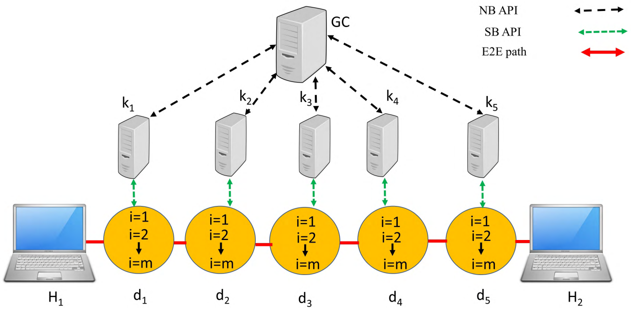


FIGURE 4. Service class selection on the E2E path.

TABLE 2. Application service requests.

Metric	Service requests		
	1	2	3
Delay	150	200	400
Jitter	60	60	80
PLR	10^{-4}	10^{-3}	10^{-3}

Each LDC has a TOPSIS module that finds a service class among a set of service classes. The service classes for d_1 on the path are available to LDC (k_1) through the TFT module

TABLE 3. QoS classes offered in five domains.

Domain D	QoS class (i)	Delay (j=1)	Jitter (j=2)	PLR (j=3)	Cost (j=4)
d_1	1	40	10	10^{-5}	40
	2	80	30	10^{-4}	20
	3	120	0	10^{-4}	15
d_2	1	20	15	10^{-6}	55
	2	50	20	10^{-5}	30
	3	70	30	5×10^{-5}	20
	4	120	0	10^{-4}	17
d_3	1	15	10	10^{-6}	75
	2	50	30	10^{-5}	60
d_4	1	12	6	10^{-5}	80
	2	0	0	10^{-4}	70
d_5	1	45	5	10^{-5}	90
	2	100	15	10^{-4}	65
	3	120	40	10^{-4}	55

using SB API protocol. The service classes are shown in (13) as a matrix, where each row represents the QoS classes on the path in a domain for a performance metric. We form a decision matrix according to (1). The result is shown in (13) by adding the values for d_1 from Table 3. Next, we apply Euclidean normalization to (13) using (3), as discussed in the section about the TOPSIS module. The result is shown in (14).

$$Q_{k=1} = \begin{bmatrix} 40 & 10 & 10^{-5} & 40 \\ 80 & 30 & 10^{-4} & 20 \\ 120 & 0 & 10^{-4} & 50 \end{bmatrix} \tag{13}$$

$$R_{k=1} = \begin{bmatrix} 0.267 & 0.316 & 0.071 & 0.848 \\ 0.535 & 0.949 & 0.705 & 0.424 \\ 0.802 & 0.000 & 0.705 & 0.318 \end{bmatrix} \tag{14}$$

Further, we use (5) to assign weights to the QoS metrics of a service class. Initially, we assign the default weights (equal to 1) to these service metrics, as shown in (15) and (16). As we do not change the weights initially, the result of (5) will not alter the values of the service metrics of (14).

$$W = [W_1, \quad W_2, \quad W_3, \quad W_4] \tag{15}$$

$$W = [1, \quad 1, \quad 1, \quad 1] \tag{16}$$

Next, we calculate positive and negative ideal solutions for the service classes by (6) and (7). Moreover, we find the deviation from the positive and negative ideal solutions of the service classes by (8) and (9). Table 4 shows the results. Finally, we obtain the deviation from the positive and negative ideal service classes by taking the square root of the summation, as shown in Table 5. Table 6 shows the priority of the service classes that are obtained by (10). The highest weight indicates the ideal service class for the domain. From the weights, we can see that the priority of the classes is $C_1 > C_3 > C_2$. Therefore, C_1 will be the desirable service class, followed by C_3 and C_2. Further, we calculate ideal classes (i.e., classes with a high priority) for other domains (i.e., d_2, d_3, d_4, and d_5) by the LDCs $k = 2$, $k = 3$, $k = 4$, and $k = 5$ on the E2E path using similar steps.

TABLE 4. Maximize the benefit criteria and minimize the cost parameter.

Equation	Delay (j=1)	Jitter (j=2)	PLR (j=3)	Cost (j=4)
$A^*_{j,k=1}$	0.267	0.000	0.071	0.318
$A^-_{i,k=1}$	0.802	0.949	0.705	0.848
$(f_{ij} - A^*_j)^2$	0	0.1	0	0.280
	0.071	0.9	0.402	0.011
	0.285	0	0.402	0
$(f_{ij} - A^-_j)^2$	0.285	0.4	0.402	0
	0.071	0	0	0.179
	0	0.9	0	0.280

TABLE 5. ED calculation for the positive and negative ideal solutions.

Equation	Delay (j=1)	Jitter (j=2)	PLR (j=3)	Cost (j=4)	Sum
$\sum_{j=1}^{n} (f_{ij} - A^*_j)^2$	0	0.1	0	0.280	0.380
	0.071	0.9	0.402	0.011	1.385
	0.285	0	0.402	0	0.688
$\sum_{j=1}^{n} (f_{ij} - A^-_j)^2$	0.285	0.4	0.402	0	1.088
	0.071	0	0	0.179	0.251
	0	0.9	0	0.280	1.180

TABLE 6. Deviation from the positive and negative ideal solutions.

Equation	Classes		
	i=1	i=2	i=3
$D^*_{i,k=1}$	0.617	1.177	0.830
$D^-_{i,k=1}$	1.043	0.501	1.087
$C^*_{i,k=1} = \frac{D^-_i}{D^*_i + D^-_i}$	0.628	0.299	0.567

The LDCs share the high-ideal service classes computed by the LDCs with the GC. Hence, the GC has the ideal QoS class for each domain on the E2E path. For example, the E2E values for the delay in these high-priority classes passing through k_1, $k = 2$, $k = 3$, $k = 4$, and $k = 5$ are 40, 50, 15, 0, and 45. The GC obtains the ideal service class from each domain, which is mapped with the service requirement by (11) and (12). The E2E service classes are computed and compared using the DC^{E2E} ratio. Equations (17), (18), and (19) calculate the DC^{E2E} ratio for $j = 1$, $j = 2$, and $j = 3$. The ideal service class values for jitter and PLR are calculated using similar procedure as we did for the delay. Finally, the flow rules for the unique E2E service class are pushed to the underlying network devices.

$$DC^{E2E}_{j=1} = \frac{150}{40 + 50 + 15 + 0 + 45} \tag{17}$$

$$DC^{E2E}_{j=2} = \frac{60}{10 + 20 + 10 + 0 + 5} \tag{18}$$

$$DC^{E2E}_{j=3} = \frac{4}{3.6} \tag{19}$$

In (17)–(19), the numerator shows the service request value required for the application from Table 2 of application service request 1. The denominator depicts the offered service calculated on the E2E path by the GC from the TOPSIS module and mapping. Equation (17) shows the service conformance ratio for E2E delay, where the $DC^{E2E}_{j=1} = 1$ shows the perfect match; therefore, the offered service class guarantees the E2E delay. Equation (18) shows that the $DC^{E2E}_{j=2}$ ratio for jitter is 1.33, which means that the offered service is better than the required service. The $DC_{j=3}^{E2E}$ for PLR is 1.11, as shown in (19). The PLR exponential values are first normalized according to the logarithmic scale and then the DC_{j}^{E2E} ratio is computed.

Next, to check the influence of the weight on the DC_{j}^{E2E} ratio of each performance metric j, we have changed the weight of the delay metric to 5 (i.e., $W = 5$) from the default value of 1. The service classes and values of the performance metrics of (13) were used for observing the impact of weight. Equation (20) shows the change in weight. The resulting weighted values are shown in Table 7. Then, positive and negative ideal service classes are determined, and the ED is calculated (as shown in Table 8 and Table 9) according to the new weight. Further, Table 10 shows the deviation from the positive and negative service classes and the final weights of the service classes according to the new weight. Table 10 shows that the priority of alternative classes changes to $C_1 > C_2 > C_3$ from $C_1 > C_3 > C_2$ with changing the weight. The ranking of the service classes changes with increasing the weight value.

$$W = [5,\ 1,\ 1,\ 1] \tag{20}$$

TABLE 7. Weight vector with changing the weight for delay.

Equation	Delay (j=1)	Jitter (j=2)	PLR (j=3)	Cost (j=4)
$f_{ij} = W_i * r_{ij}$	1.33	0.316	0.071	0.848
	2.67	0.949	0.705	0.424
	4.00	0.00	0.705	0.318

TABLE 8. Maximize the benefit criteria and minimize the cost parameter.

Equation	Delay (j=1)	Jitter (j=2)	PLR (j=3)	Cost (j=4)
$A_{j,k=1}^{*}$	1.336	0.000	0.071	0.318
$A_{j,k=1}^{-}$	4.009	0.949	0.705	0.848
$(f_{ij} - A_j^{*})^2$	0	0.1	0	0.280
	1.78	0.9	0.402	0.011
	7.14	0	0.402	0
$(f_{ij} - A_j^{-})^2$	7.14	0.4	0.402	0
	1.78	0	0	1.79
	0	0.9	0	0.280

TABLE 9. ED calculation for the positive and negative ideal solution.

Equation	Delay (j=1)	Jitter (j=2)	PLR (j=3)	Cost (j=4)	Sum
$\sum_{j=1}^{n} (f_{ij} - A_j^{*})^2$	0	0.1	0	0.280	0.380
	1.78	0.9	0.402	0.011	3.09
	7.14	0	0.402	0	7.54
$\sum_{j=1}^{n} (f_{ij} - A_j^{-})^2$	7.14	0.4	0.402	0	7.94
	1.78	0	0	0.179	1.96
	0	0.9	0	0.280	1.180

TABLE 10. Deviation from the positive and negative ideal solutions using ED and priority of the alternative classes.

Equation	Classes		
	i=1	i=2	i=3
$D_{i,k=1}^{*}$	0.617	1.761	2.747
$D_{i,k=1}^{-}$	2.819	1.402	1.087
$C_{i,k=1}^{*} = \frac{D_i^{-}}{D_i^{*} + D_i^{-}}$	0.820	0.443	0.283

Next, to check whether the new weight influences the E2E performance, the $DC_{j=1}^{E2E}$ ratio is found by (21). The $DC_{j=1}^{E2E}$ ratio for the increased weight changes to $DC_{j=1}^{E2E} = 1.25$ from 1, i.e., the offered delay service for the application further reduces with $DC_{j=1}^{E2E} > 1$. Initially the $DC_{j=1}^{E2E} = 1$ for the $W = 1$ shows the E2E delay is equal to the requested value of the application, i.e., 150 milliseconds (ms). However, when the weight is increased to $W = 5$, the offered service class values further improve, and the E2E delay becomes smaller, i.e., 120 ms as compared with 150 ms.

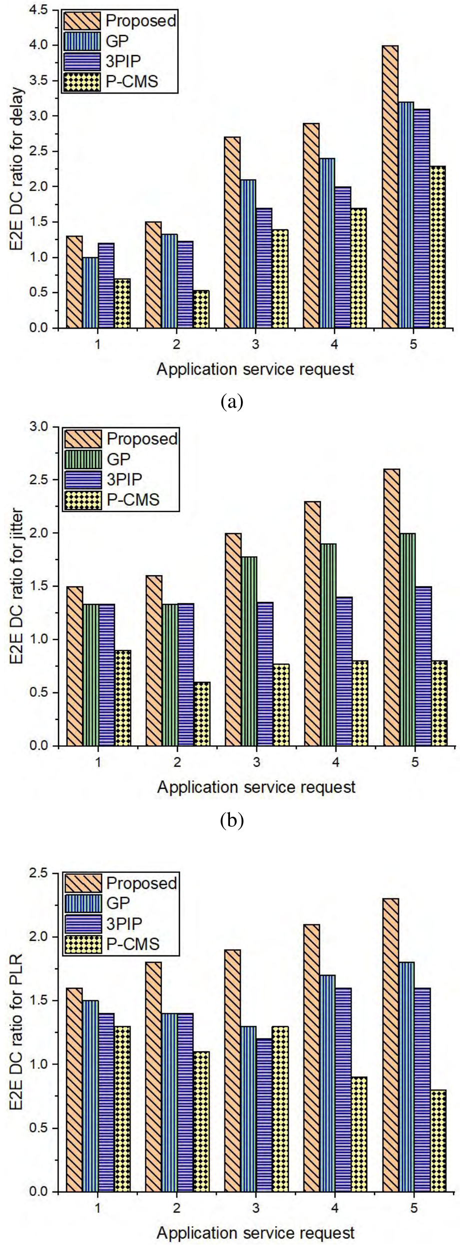
$$DC_{j=1}^{E2E} = \frac{150}{40 + 20 + 15 + 0 + 45} = \frac{150}{120} \tag{21}$$

IV. SIMULATION RESULTS AND DISCUSSION

Results for the proposed approach are compared with existing ones, such as P-CMS [18], 3PIP [19], and GP [20]. The E2E path consists of five LDCs (k_1, k_2, k_3, k_4, and k_5). Delay, jitter, PLR, and cost (per bandwidth unit) are the main QoS service class parameters in each domain. The SDN network with 50 nodes created in Mininet [57] is divided into 5 domains and the controllers are placed according to the placement defined in [45]. Thus, the total number of domains is 5 and the number of nodes assigned to each controller are 5 to 20 [58]. Each domain is administered by one LDC. We use the OpenDaylight [11] controller for the implementation of the proposed architecture. The nodes in Mininet are connected with OpenDaylight controllers (LDCs and GC) by calling their IP addresses.

Both Mininet and OpenDaylight run in virtual machines on a server. The configuration of the server consists of Intel Corei7, 3.4 GHz and 16 GB RAM. The operating system is Linux i.e. Ubuntu 16.04. The TFT, PCM, and TOPSIS modules are loaded as applications on the OpenDaylight controller. For an incoming service request, if the flow rules on the switches do not exist then the request is sent to LDC, which sends it to GC. In each LDC the TFF module obtains the source and destination information, which is used by the E2E path calculation (PCM) and TOPSIS modules. Further, the TOPSIS module finds the ideal QoS class among a set of service classes on the E2E path. A GC proactively configures the resources such as the E2E path, QoS classes and performs mapping. Each service class values for the delay, jitter, and PLR are calculated using the distributed Internet traffic generator (D-ITG) [59]. The cost (per bandwidth unit) is computed from the bandwidth computed with iperf [60]. The mapping is performed by defining flow rules (FR) for the service classes in the $CBMQ$ on the SDN switches.

We compute the DC_{j}^{E2E} ratio for $j = 1, j = 2$, and $j = 3$ for five application service requests. The E2E delay, jitter, PLR, and cost are also analyzed by changing the weight of these metrics. Moreover, the DC_{j}^{E2E} ratio is also assessed with a change in the weight for each performance metric j. Moreover, the DC_{j}^{E2E} ratio of the proposed approach is compared with FCA and the proposed scheme without a TOPSIS module (WTM). More and more, we evaluate the E2E delay,



(a)

(b)

(c)

FIGURE 5. Comparison of DC_j^{E2E} ratio for QoS metrics. (a) Delay ($j = 1$), (b) jitter ($j = 2$), and (c) PLR ($j = 3$).

jitter, PLR, and cost on the E2E path passing through five domains with the proposed scheme, FCA, and WTM. Finally, we demonstrate the effect of the cost (per bandwidth unit) on the E2E performance according to the customer priorities and compare the E2E cost with GP, 3PIP, and P-CMS.

A. ANALYSIS OF DC_j^{E2E} RATIO

Figure 5 compares the results of DC_j^{E2E} ratio for the QoS metrics such as delay ($j = 1$), jitter ($j = 2$), and PLR ($j = 3$) of our proposed SDN-based scheme leveraging TOPSIS with P-CMS, 3PIP, and GP. The DC_j^{E2E} ratios for the delay, jitter, and PLR are calculated on the E2E path. Because DC_j^{E2E} shows a ratio between the required and offered service classes, the values greater than 1 indicate that a scheme is offering a service metric value that is greater than the required value. For example, the delay ratio with the proposed approach is either equal to 1 or it is greater than 1 for all the application service requests. Figure 5 shows that the $DC_{j=1}^{E2E}$ ratios for the delay ($j = 1$) with P-CMS are lower than all other schemes. Consequently, E2E application service requests cannot be fulfilled, which is evident from the $DC_{j=1}^{E2E}$ ratio of P-CMS, which is less than 1 for application service requests 1 and 2. Similarly, the $DC_{j=1}^{E2E}$ ratio for other schemes are also lower than in the proposed scheme.

Moreover, Figure 5 shows that the DC_j^{E2E} ratio for the delay, jitter, and PLR ($j = 1, 2, 3$) for the SDN-based scheme is greater than other schemes. Figure 5 also shows that the 3P-based schemes such as GP and 3PIP $DC_{j=1}^{E2E}$ ratio and the proposed approach performs well as their values are either 1 or greater than 1; however, our proposed scheme surpasses the 3P schemes for most application service requests. The reason is that the proposed scheme computes a unique ideal service class for the E2E path in the GC according to the E2E application service requests. For P-CMS, the mapping is performed on the local domain level. Therefore, $DC_{j=1}^{E2E} < 1$ for service requests 1 and 2.

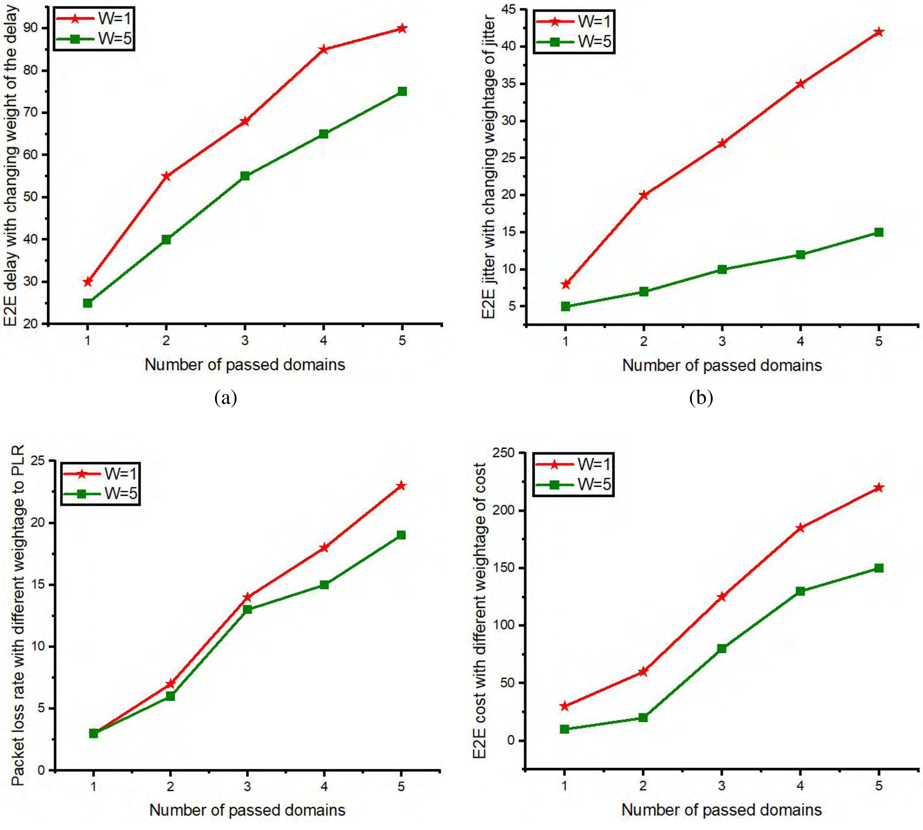
B. ANALYSIS OF E2E QoS METRICS WITH CHANGING WEIGHT

In this experiment, we have changed the weights of delay, jitter, PLR, and cost to see the impact on the E2E performance for our proposed scheme. Figure 6 compares E2E values for the QoS metrics (delay, jitter, PLR, and cost) with $W = 1$ and $W = 5$. According to Figure 6, if the weight of the delay is changed to 5, the E2E delay (ms) significantly decreases. Likewise, Figures 6(B)–(D) show a decrease in the E2E jitter (ms), PLR, and cost with a change in the weight ($W = 5$). Moreover, we have used (22) to check the percentage of decrease (PD) of every performance parameter when the weight is increased. Values of each parameter before the increasing weight ($PVBIW$) and after increasing weight ($PVAIW$) were recorded. The PD is substantial with a change in the weight from 1 to 5 for the E2E delay, jitter, PLR, and cost passing through five domains. The PD for the E2E delay is 33.76% according to this formula. Similarly, the change in weight significantly influences the jitter: the PD in jitter is 71.32%. Likewise, the PD for PLR is 11.76%. For the cost, the PD is 25.11%. The PD in delay, jitter, PLR and cost from (22) shows the effectiveness of the weight factor in the TOPSIS module.

$$PD = \frac{(PVBIW - PVAIW)}{PVBIW} \times 100 \tag{22}$$

C. ANALYSIS OF DC_j^{E2E} RATIO WITH CHANGING WEIGHT

In this experiment, we evaluated the DC_j^{E2E} ratio to see whether the QoS satisfies the service requirements of



(a)

(b)

(c)

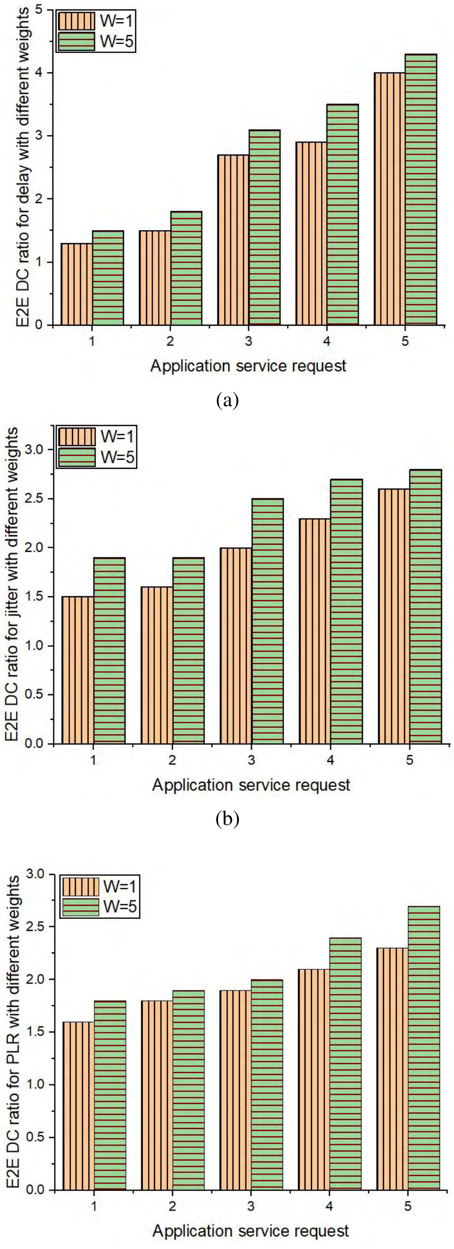
(d)

FIGURE 6. Comparison of performance metrics with different weight assignment ($W = 1$ and $W = 5$) on the path passing through five domains. (a) Delay ($j = 1$), (b) jitter ($j = 2$), (c) PLR ($j = 3$), and (d) cost ($j = 4$).

the applications with newly assigned weights. Therefore, we changed the default weights of delay, jitter, and PLR from $W = 1$ to $W = 5$ in the proposed scheme and checked the DC_j^{E2E} ratio for ($j = 1$, 2, and 3). Figure 7 compares the DC_j^{E2E} ratio with $W = 1$ and $W = 5$ for the delay, jitter, and PLR for the application service requests 1, 2, 3, 4, and 5, respectively. According to Figures 7(A)–(C), if the weight of the performance metric is changed, the DC_j^{E2E} ratio changes. Figure 7 (A) shows an impact of the higher weight on the performance metric delay; a greater weight means that the resulting $DC_{j=1}^{E2E}$ ratio is greater than 1: $DC_{j=1}^{E2E} > 1$. As a result, we can see a decrease in the E2E delay when passing through five domains. Therefore, the performance of the application will improve because of the lower E2E delay for application service requests, which is obvious from the higher $DC_{j=1}^{E2E}$ ratio. Consequently, the offered service class values in the domains will guarantee the service requests 1, 2, 3, 4, and 5 for the delay. Similarly, Figure 7(B) and Figure 7(C) show a higher DC_j^{E2E} ratio with an increase in weight for $j = 2$ and $j = 3$. Therefore, the jitter and PLR on the E2E path will further reduce with the higher weights of these performance metrics.

D. ANALYSIS OF DC_j^{E2E} RATIO WITH DIFFERENT SDN ARCHITECTURES

To check the effectiveness of the hierarchical SDN architecture equipped with a TOPSIS module (the proposed approach) on the E2E performance metrics, we have computed the DC_j^{E2E} for the delay, jitter, and PLR with the proposed method. Then, we repeated the experiment by removing the TOPSIS module from the LDCs, i.e., executing the simulations without a TOPSIS module (WTM) and, finally, in the FCA without the GC and TOPSIS module. Figure 8 compares the DC_j^{E2E} for delay ($j = 1$), jitter ($j = 2$), and PLR ($j = 3$) in the three scenarios. According to Figure 8, $DC_{j=1}^{E2E} < 1$ for the architecture without the TOPSIS module for some application service requests, because the ideal service classes are not selected in the absence of a TOPSIS module. The FCA $DC_{j=1}^{E2E} < 1$ for most service requests, because it lacks a global view and partial inter-domain collaboration, which influence the E2E performance metrics. Similarly, the same effect on the DC_j^{E2E} is evident for PLR ($j = 2$) and jitter ($j = 3$). Hence, the proposed approach guarantees the E2E QoS service for different application service requests whereas preserving the SDN programma-

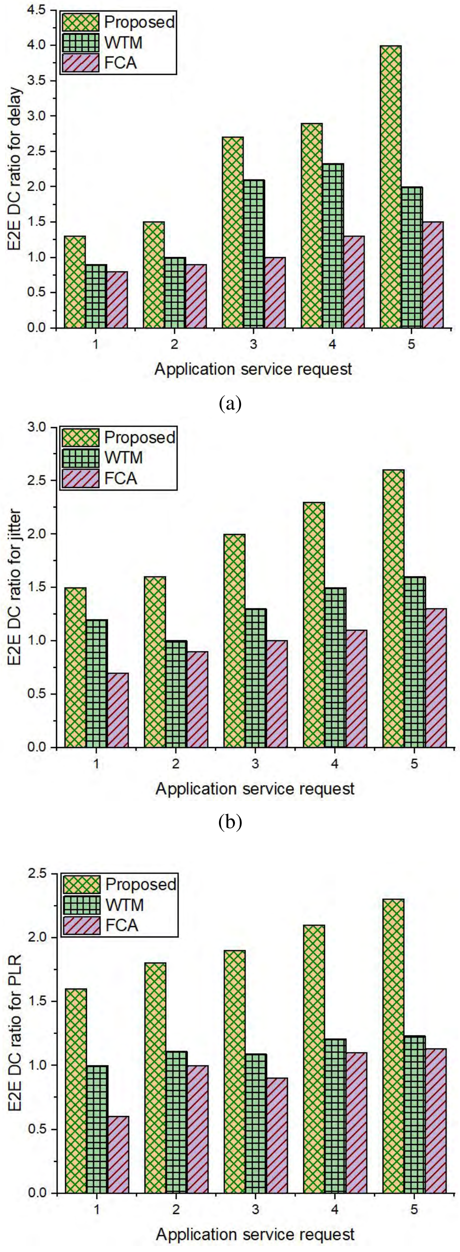


(a)

(b)

(c)

FIGURE 7. Comparison of DC_j^{E2E} ratio of with $W = 1$ and $W = 5$ for performance metrics. (a) Delay ($j = 1$), (b) jitter ($j = 2$), and (c) PLR ($j = 3$).



(a)

(b)

(c)

FIGURE 8. Comparison of DC_j^{E2E} ratio of the proposed approach with WTM and FCA for QoS metrics. (a) Delay ($j = 1$), (b) jitter ($j = 2$), and (c) PLR ($j = 3$).

bility, flexibility, and weight-adjusting capability for further reducing the E2E delay, jitter, PLR, and cost.

E. ANALYSIS OF E2E QoS METRICS WITH DIFFERENT SDN ARCHITECTURES

Figure 9 compares E2E values for the QoS metrics (delay, jitter, PLR, and cost) with three architectures: the proposed scheme, WTM, and FCA. According to Figure 9, when the TOPSIS module is removed from the LDCs, there is a significant increase in the E2E delay (ms). Likewise, Figures 9 (B)–(D) show an increase in the E2E jitter (ms), PLR, and cost in the absence of a TOPSIS module and GC. Moreover, we have used (23) to check the percentage of increase (PI) of each QoS metric when the architecture is changed. For each QoS metric, we have recorded its value before any change in the architecture ($PVBC$) and after the change in the architecture ($PVAC$). By (23), the PI for E2E delay is 35.78% with WTM compared with our proposed scheme. However, the PI for E2E delay with FCA is substantial (60.23%), because the FCA lacks a global view of the service classes and does not have a TOPSIS module for ideal

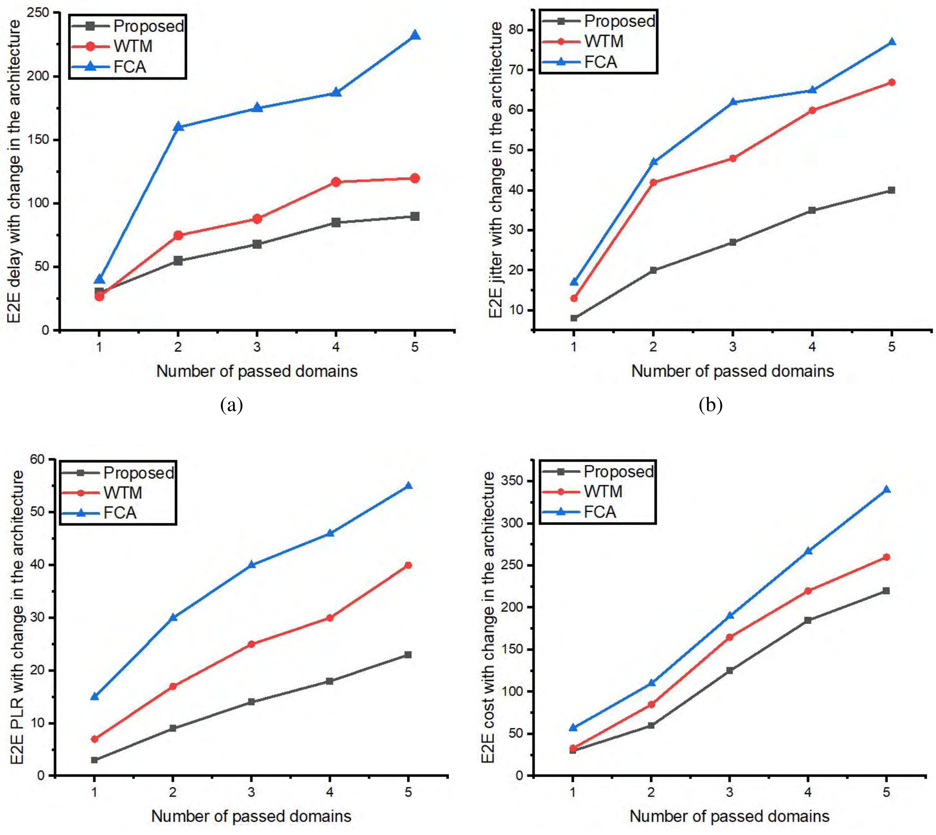


FIGURE 9. Comparison of performance metrics with the proposed scheme, WTM, and FCA for the path passing through five domains. (a) Delay ($j = 1$), (b) jitter ($j = 2$), (c) PLR ($j = 3$), and (d) cost ($j = 4$).

service class selection on the E2E path. Similarly, the change in the architecture significantly influences the jitter as well, where the PI in jitter is 55.77% for WTM; for FCA, PI jumps to 75.28%. Likewise, the PI for PLR is 55.73% with WTM and, for FCA, the PI in PLR further increases to 92.23%. Finally, the PI reported cost is 7.87% and 14.66% for WTM and FCA architectures, respectively. Hence, according to the results in Figure 9, the hierarchical control plane with a GC and TOPSIS module effectively reduces the E2E delay, jitter, PLR, and cost.

$$PI = \frac{(PVAC - PVBC)}{PVBC} \times 100 \tag{23}$$

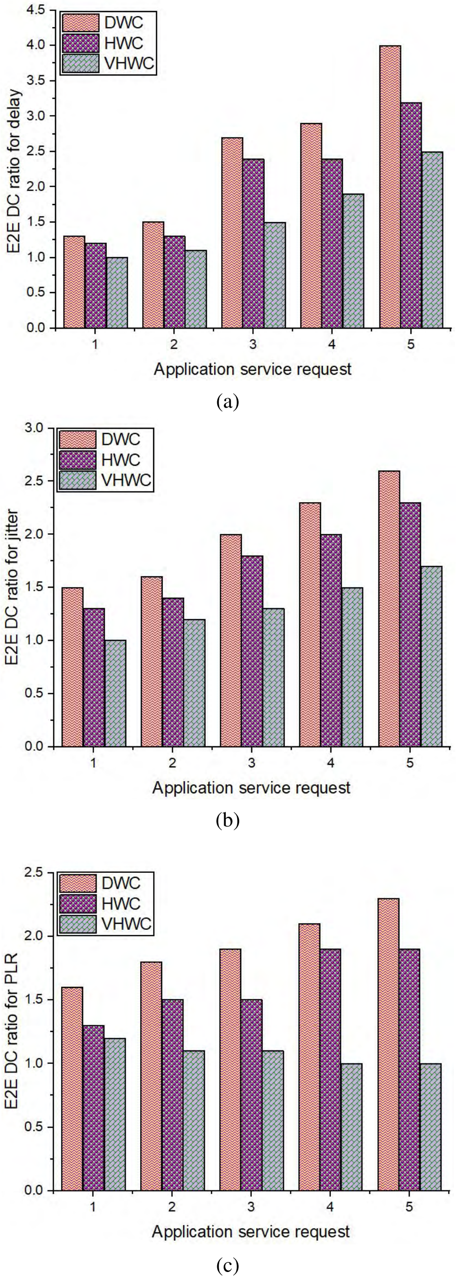
F. EFFECT OF THE COST (PER BANDWIDTH UNIT) ON THE DC_j^{E2E} RATIO

In this experiment, we evaluate the impact of the change in the weight of the cost and observe the effect on the on the DC_j^{E2E} ratio for ($j = 1, 2, 3$) i.e. delay, jitter and PLR. The customer choices are prioritized through the weight associated with the cost provided by the TOPSIS module in the SDN controller. Figures 10 (A)–(C) show the effect of change in the weight of the cost on the DC_j^{E2E} ratio in three scenarios i.e. (1) when the customer wants the services with default weight of the cost (DWC) which is equal to 1, (2) if the customer wants a cheaper service, then we assign a high weight to the cost (HWC) i.e. 5, and (3) when the customer wants more cheaper service i.e. very high weight of the cost (VHWC) which is to 10 because after this value the DC_j^{E2E} for ($j = 1, 2, 3$) is negligible. To see the impact of these three priorities, we calculate the DC_j^{E2E} ratio for ($j = 1, 2, 3$) and also calculate the average E2E cost.

Figures 10 (A)–(C) shows when the weight of the cost changes then the results of DC_j^{E2E} ratio for the delay, jitter, and PLR from the TOPSIS mapping also change. We can see from Figure 10 (A) that when the customer priority for the cost increases. i.e. the customer wants a cheaper service (HWC and VHWC), then a decrease in the $DC_{j=1}^{E2E}$ ratio of delay is evident from the graph. However, the $DC_{j=1}^{E2E}$ ratio of the proposed method is still equal to or greater than 1 for all of the service requests. Hence, the QoS is provisioned for the service requests but the delay is a little high for the cheaper services. i.e. with the cost plans of HWC and VHWC. Figures 10 (B)–(C) shows similar trends in the graphs for the DC_j^{E2E} ratio of jitter and PLR ($j = 2, 3$).

G. EVALUATION OF E2E COST (PER BANDWIDTH UNIT)

In this subsection, we evaluate the E2E cost (per bandwidth unit) for DWC, HWC, and VHWC. Further, we compare



(a)

(b)

(c)

FIGURE 10. Comparison of DC_j^{E2E} ratio of the proposed approach with different cost priorities for (a) Delay ($j = 1$), (b) jitter ($j = 2$), and (c) PLR ($j = 3$).

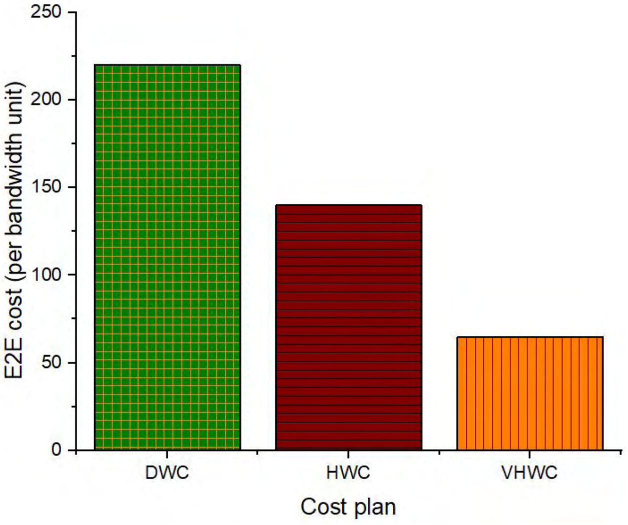


FIGURE 11. Comparison of Average E2E Cost (per bandwidth unit) on the E2E path passing through five domains with DWC ($W = 1$), HWC ($W = 5$) and VHWC ($W = 10$).

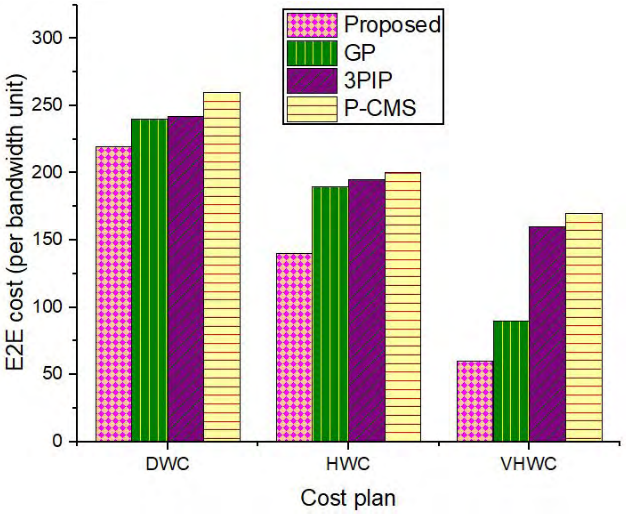


FIGURE 12. Comparison of Average E2E Cost (per bandwidth unit) on the E2E path passing through five domains with DWC ($W = 1$), HWC ($W = 5$) and VHWC ($W = 10$) of the proposed scheme with P-CMS, 3PIP, and GP.

it with the previous schemes i.e. P-CMS, 3PIP, and GP. Figure 11 shows the average E2E cost for these three scenarios (DWC, HWC, and VHWC) according to the cost priorities of the customer or the cost plans. We can see a decrease in the average E2E cost as the priority of the cost increases for the customer. The cost is more with the default cost plan. However, if the customer compromises on the E2E performance such as delay, jitter, and PLR, then there is a reduction in the E2E cost because we increase the weight of the cost in the TOPSIS module of the SDN controller. Figure 11 shows that as the weight of the cost increase (VHWC) then the average E2E cost is minimum among the three cost plans.

Figure 12 depicts the comparison of the E2E cost for the three cost scenarios (DWC, HWC, and VHWC) with P-CMS, 3PIP, and GP. We can see that the E2E cost is smaller for the proposed scheme. The GC computes the ideal service classes for the E2E path utilizing the TOPSIS module due to which the cost of the proposed solution is small as compared to P-CMS, 3PIP, and GP. We can see that P-CMS has a high cost due to its lack of a global view of the E2E service classes. Moreover, we can see that the E2E cost is small even with increasing the weight of the cost i.e. with HWC and VHWC as compared to other mapping schemes.

V. CONCLUSION

The distributed management and the tight coupling of data and control planes in the traditional networks restrict the global view and control of the network resources. With the emergence of centralized management through SDN technology, the underlying networks are managed globally from the central controller. However, a single controller can lead to a single point of failure and E2E service delivery issues in the presence of multiple providers with varying QoS classes. In this study, we have investigated the applicability of the hierarchical SDN architecture to the E2E service class mapping in a multi-domain environment using a multi-objective decision-making scheme TOPSIS. The concept of hierarchical architecture enables the management and collaboration of autonomous domains on the E2E path for service class mapping. Traditional Internet architecture and prevalent SDN schemes lack an effective QoS class mapping mechanism to guarantee the E2E QoS. In this paper, we first proposed an architecture for E2E collaboration among the autonomous domains with a packet processing algorithm. Next, we provided a TOPSIS-based ideal service class mapping scheme in SDN. Our proposed approach is flexible and adaptable to various technical and non-technical service parameters. A demonstration of the proposed approach was made with standard QoS classes offered in five domains.

The simulation results demonstrate the superiority of the proposed approach for the E2E QoS fulfillment in a multi-domain environment with different QoS classes in each domain. Additionally, the results verify that the degree of correspondence for the proposed approach is 1 or above 1 for most of the application service requests; hence, the offered service is far better than the required service. Furthermore, we analyzed the delay, jitter, PLR, and cost for E2E domains according to different weights allocated to these metrics. The percentage of reduction according to the changing weights and the DC_j^{E2E} was investigated for the QoS guarantee according to the service demands of the application. To check the effectiveness of the proposed approach, the PI in delay, jitter, PLR and cost was also compared with FCA and WTM SDN architectures. Finally, we evaluated the cost (per bandwidth unit) and compared it with the previous schemes. In future work, we will investigate the resources utilization in a multi-domain environment. Further, qualitative parameters will also be considered on the E2E path that influences the E2E QoS.

REFERENCES

[1] B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, “A survey of software-defined networking: Past, present, and future of programmable networks,” *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1617–1634, 3rd Quart., 2014.

[2] H. Farhady, H. Lee, and A. Nakao, “Software-defined networking: A survey,” *Comput. Netw., Int. J. Comput. Telecommun. Netw.*, vol. 81, pp. 79–95, Apr. 2015.

[3] J. H. Cox, J. Chung, S. Donovan, J. Ivey, R. J. Clark, G. Riley, and H. L. Owen, “Advancing software-defined networks: A survey,” *IEEE Access*, vol. 5, pp. 25487–25526, 2017.

[4] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling innovation in campus networks,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

[5] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, J. Zolla, U. Hölzle, S. Stuart, and A. Vahdat, “B4: Experience with a globally-deployed software defined WAN,” in *Proc. ACM SIGCOMM Conf.*, 2013, pp. 3–14.

[6] S. J. Vaughan-Nichols, “OpenFlow: The next generation of the network?” *Computer*, vol. 44, no. 8, pp. 13–15, Aug. 2011.

[7] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker, “NOX: Towards an operating system for networks,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 3, pp. 105–110, Jul. 2008.

[8] *NOX Controller*. Accessed: Sep. 2019. [Online]. Available: https://github.com/noxrepo/nox

[9] *RYU Controller*. Accessed: Oct. 2019. [Online]. Available: https://osrg.github.io/ryu/

[10] *Floodlight Controller*. Accessed: Nov. 2019. [Online]. Available: https://github.com/floodlight/floodlight

[11] *OpenDayLight (ODL) Controller*. Accessed: Oct. 2019. [Online]. Available: http://www.opendaylight.org/

[12] *Open Network Operating System*. Accessed: Nov. 2019. [Online]. Available: https://github.com/opennetworkinglab/onos

[13] P. Levis and M. Boucadair, *Considerations of Provider-To-Provider Agreements for Internet-Scale Quality of Service (QoS)*, document RFC 5160, IETF, 2008.

[14] P. Jacob and B. Davie, “Technical challenges in the delivery of interprovider QoS,” *IEEE Commun. Mag.*, vol. 43, no. 6, pp. 112–118, Jun. 2005.

[15] J. Wulf, F. Limbach, and R. Zarnekow, “Analysis of QoS platform cooperation strategies,” in *Proc. 21st Eur. Regional ITS Conf.*, 2010, pp. 1–17.

[16] J. Zhao, J. Guan, C. Xu, W. Su, and H. Zhang, “Peering strategic game models for interdependent ISPs in content centric Internet,” *Sci. World J.*, vol. 2013, no. 10, pp. 1–10, 2013.

[17] S. I. Maniatis, E. G. Nikolouzou, and I. S. Venieris, “End-to-end QoS specification issues in the converged all-IP wired and wireless environment,” *IEEE Commun. Mag.*, vol. 42, no. 6, pp. 80–86, Jun. 2004.

[18] M. D. Stojanovic and S. V. B. Rakas, “Policies for allocating performance impairment budgets among multiple IP providers,” *AEU-Int. J. Electron. Commun.*, vol. 67, no. 3, pp. 206–216, Mar. 2013.

[19] B. J. Mali, N. M. Ninkovic, M. D. Stojanovic, and G. I. Savic, “Service class mapping based on integer programming algorithm in the third party agent,” in *Proc. 22nd Telecommun. Forum (TELFOR)*, Belgrade, Serbia, Nov. 2014, pp. 170–173.

[20] N. M. Ninkovic, B. J. Mali, M. D. Stojanovic, and G. I. Savic, “Multi-objective third-party approach for service class mapping among multiple providers in the Internet,” *Elektronika ir Elektrotechnika*, vol. 21, no. 2, pp. 80–84, 2015.

[21] E. Boschi, M. Bossardt, and T. Dubendorfer, “Validating inter-domain SLAs with a programmable traffic control system,” in *Active and Programmable Networks* (Lecture Notes in Computer Science), vol. 4388. Berlin, Germany: Springer-Verlag, 2005, pp. 1–12. [Online]. Available: https://dl.acm.org/doi/10.1007/978-3-642-00972-3_1

[22] Y. Wang and J. Bi, “Survey of mechanisms for inter- domain SDN,” *ZTE Commun.*, vol. 15, no. 3, p. 1, 2017.

[23] Q. Duan, “End-to-end service delivery with QoS guarantee in software-defined networks,” *Trans. Netw. Commun.*, vol. 6, no. 2, p. 10, 2018.

[24] A. Mendiola, J. Astorga, E. Jacob, and K. Stamos, “Enhancing network resources utilization and resiliency in multi-domain bandwidth on demand service provisioning using SDN,” *Telecommun. Syst.*, vol. 71, no. 3, pp. 505–515, Jul. 2019.

[25] K. D. Joshi and K. Kataoka, “PRIME-Q: Privacy aware End-to-end QoS framework in multi-domain SDN,” in *Proc. IEEE Conf. Netw. Softwarization (NetSoft)*, Jun. 2019, pp. 169–177.

[26] F. Lucrezia, G. Marchetto, F. Risso, M. Santuari, and M. Gerola, “A proposal for End-to-end QoS provisioning in software-defined networks,” *Int. J. Electr. Comput. Eng. (IJECE)*, vol. 7, no. 4, pp. 2261–2277, 2017.

[27] M. Karakus and A. Durresi, “A scalable inter-AS QoS routing architecture in software defined network (SDN),” in *Proc. IEEE 29th Int. Conf. Adv. Inf. Netw. Appl.*, Gwangju, South Korea, Mar. 2015, pp. 148–154.

[28] Y. Rekhter, T. Li, and S. Hares, *A Border Gateway Protocol 4 (BGP-4)*, document RFC 4271, 2006.

[29] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, *Resource Reservation Protocol (RSVP)—Version 1 Functional Specification*, document RFC 2205, 1997.

[30] H. Gredler, J. Medved, S. Previdi, A. Farrel, and S. Ray, *North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP*, document RFC 7752, 2016.

[31] R. Singh, H. Koo, N. Miramirkhani, F. Mirhaj, P. Gill, and L. Akoglu, "The politics of routing: Investigating the relationship between AS connectivity and Internet freedom," in *Proc. 6th USENIX Workshop Free Open Commun. Internet*, 2016, pp. 1–7.

[32] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed Internet routing convergence," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 30, no. 4, pp. 175–187, 2000.

[33] T. G. Griffin and G. Wilfong, "Analysis of the MED oscillation problem in BGP," in *Proc. 10th IEEE Int. Conf. Netw. Protocols*, Nov. 2002, pp. 90–99.

[34] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, and T. Hama, "Onix: A distributed control platform for large-scale production networks," in *Proc. OSDI*, 2010, pp. 1–6.

[35] *SDN Architecture*, document ONF TR-502, Jun. 2014, no. 1.

[36] *Openflow-Enable Transport SDN*, Open Netw. Found., Menlo Park, CA, USA, 2014.

[37] *SDN Architecture*, document ONF TR-521, Jan. 2016, no. 1.

[38] L. E. Li, Z. M. Mao, and J. Rexford, "Toward software-defined cellular networks," in *Proc. Eur. Workshop Softw. Defined Netw.*, Oct. 2012, pp. 7–12.

[39] I. Elgendi, K. S. Munasinghe, and A. Jamalipour, "A three-tier SDN architecture for DenseNets," in *Proc. 9th Int. Conf. Signal Process. Commun. Syst. (ICSPCS)*, Cairns, QLD, Australia, Dec. 2015, pp. 1–7.

[40] R. Khalili, Z. Despotovic, and A. Hecker, "Flow setup latency in SDN networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 12, pp. 2631–2639, Dec. 2018.

[41] I. Elgendi, K. S. Munasinghe, and B. Mcgrath, "A heterogeneous software defined networking architecture for the tactical edge," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, Nov. 2016, pp. 1–7.

[42] *Technical Specification Group Services and System Aspects; Policy and Charging Control Architecture*, document TS 23.203, 3GPP, Jun. 2010.

[43] S.-C. Lin, I. F. Akyildiz, P. Wang, and M. Luo, "QoS-aware adaptive routing in multi-layer hierarchical software defined networks: A reinforcement learning approach," in *Proc. IEEE Int. Conf. Services Comput. (SCC)*, Jun. 2016, pp. 25–33.

[44] O. Chato and W. E. S. Yu, "An exploration of various quality of service mechanisms in an OpenFlow and software defined networking environment in terms of latency performance," in *Proc. Int. Conf. Inf. Sci. Secur. (ICISS)*, Dec. 2016, pp. 1–7.

[45] B. Heller, R. Sherwood, and N. McKeown, "The controller placement problem," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 4, pp. 473–478, Sep. 2012.

[46] K. Yoon and C. L. Hwang, *Multiple Attribute Decision Making: Methods and Applications*. Berlin, Germany: Springer-Verlag, 1981.

[47] E. Triantaphyllou, *Multi-Criteria Decision Making Methods: A Comparative Study* (Applied Optimization), vol. 44. Norwell, MA, USA: Kluwer, 2002.

[48] N. A. Nabeeh, F. Smarandache, M. Abdel-Basset, H. A. El-Ghareeb, and A. Aboelfetouh, "An integrated neutrosophic-TOPSIS approach and its application to personnel selection: A new trend in brain processing and analysis," *IEEE Access*, vol. 7, pp. 29734–29744, 2019.

[49] M. Alaa, I. S. M. A. Albakri, C. K. S. Singh, H. Hammed, A. A. Zaidan, B. B. Zaidan, O. S. Albahri, M. A. Alsalem, M. M. Salih, E. M. Almahdi, M. J. Baqer, N. S. Jalood, S. Nidhal, A. H. Shareef, and A. N. Jasim, "Assessment and ranking framework for the English skills of pre-service teachers based on fuzzy Delphi and TOPSIS methods," *IEEE Access*, vol. 7, pp. 126201–126223, 2019.

[50] E. K. Zavadskas, Z. Turskis, and V. Bagočius, "Multi-criteria selection of a deep-water port in the eastern Baltic sea," *Appl. Soft Comput.*, vol. 26, pp. 180–192, Jan. 2015.

[51] P. P. Kalbar, S. Karmakar, and S. R. Asolekar, "Selection of an appropriate wastewater treatment technology: A scenario-based multiple-attribute decision-making approach," *J. Environ. Manage.*, vol. 113, pp. 158–169, Dec. 2012.

[52] V. Yadav, S. Karmakar, P. P. Kalbar, and A. K. Dikshit, "PyTOPS: A Python based tool for TOPSIS," *SoftwareX*, vol. 9, pp. 217–222, Jan. 2019.

[53] M. Lahby and A. Sekkaki, "Optimal vertical handover based on TOPSIS algorithm and utility function in heterogeneous wireless networks," in *Proc. Int. Symp. Netw., Comput. Commun. (ISNCC)*, May 2017, pp. 1–6.

[54] S. Chakraborty and C.-H. Yeh, "A simulation comparison of normalization procedures for TOPSIS," in *Proc. Int. Conf. Comput. Ind. Eng.*, Jul. 2009, pp. 1815–1820.

[55] M. D. Stojanovic, S. V. Bostjancic Rakas, and V. S. Acimovic-Raspopovic, "End-to-end quality of service specification and mapping: The third party approach," *Comput. Commun.*, vol. 33, no. 11, pp. 1354–1368, Jul. 2010.

[56] *Network Performance Objectives for IP-Based Services*, Standard ITU-T Recommendation Y.1541, 2011.

[57] R. L. S. de Oliveira, C. M. Schweitzer, A. A. Shinoda, and L. R. Prete, "Using mininet for emulation and prototyping software-defined networks," in *Proc. IEEE Colombian Conf. Commun. Comput. (COLCOM)*, Jun. 2014, pp. 1–6.

[58] G. Cheng, H. Chen, Z. Wang, and S. Chen, "DHA: Distributed decisions on the switch migration toward a scalable SDN control plane," in *Proc. IFIP Netw. Conf. (IFIP Networking)*, May 2015, pp. 471–477.

[59] A. Botta, A. Dainotti, and A. Pescapé, "A tool for the generation of realistic network workload for emerging networking scenarios," *Comput. Netw.*, vol. 56, no. 15, pp. 3531–3547, Oct. 2012.

[60] *IPerf Speed Test Tool*. Accessed: Feb. 2020. [Online]. Available: https://www.iperf.fr/

JEHAD ALI (Graduate Student Member, IEEE) received the B.S. and M.S. degrees from the Department of Computer Systems Engineering, University of Engineering and Technology, Peshawar, Pakistan, in 2008 and 2016, respectively, and the Ph.D. degree from Ajou University, Suwon, South Korea, in 2019. From 2011 to 2016, he has worked as a Computer Engineer with the Ghulam Ishaq Khan Institute of Engineering Science and Technology, Pakistan. He is currently working as a Postdoctoral Researcher with the Department of Computer Engineering, Ajou University. He has contributed and presented papers at the IEEE IEMCON 2019 held at UBC Canada, ICNC held at Honolulu, HI, and USA, and MobiSys 2019, Seoul, South Korea. His research interests include QoS improvements for software defined networking (SDN), the optimization of SDN controllers placements, and the benchmarking of SDN controllers. His research interests also include wireless sensor networks (WSNs), the Internet-of-Things (IoT), and the applications of machine learning for SDN. He is also reviewing articles for several reputable SCIE and Springer journals such as the IEEE Comunication Letters and Wireless personal communications.

BYEONG-HEE ROH (Senior Member, IEEE) received the B.S. degree in electronics engineering from Hanyang University, Seoul, South Korea, in 1987, and the M.S. and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 1989 and 1998, respectively. From 1989 to 1994, he was with the Telecommunication Networks Laboratory, Korea Telecom, as a Researcher. From February 1998 to March 2000, he worked with Samsung Electronics Company, Ltd., South Korea, as a Senior Engineer. Since March 2000, he has been with the Department of Software and Computer Engineering, Ajou University, Suwon, South Korea, where he is currently a Professor. In 2005, he was a Visiting Associate Professor with the Department of Computer Science, State University of New York, Stony Brook, NY, USA. In 2014, he was an Adjunct Researcher with the Agency for Defense Development (ADD), South Korea. Since 2016, he has been the Director of the MR-IoT Convergence Disaster Response AI Research Center supported by the Ministry of Science and ICT (MSIT), South Korea, through the Information Technology Research Center (ITRC) Support Program. His research interests include mobile multimedia networking, future Internet, military communications, the Internet of Things (IoT) platform and services, augmented reality (AR), and mixed reality (MR).