ARTICLE

A Multi-Constraint Path Optimization Scheme Based on Information Fusion in Software Defined Network

Jinlin Xu[1,2], Wansu Pan[1,*], Longle Cheng[1,2], Haibo Tan[1,2], Munan Yuan[1,*] and Xiaofeng Li[1,2]

[1]Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei, 230031, China

[2]University of Science and Technology of China, Hefei, 230026, China

*Corresponding Authors: Wansu Pan. Email: wspan@hfcas.ac.cn; Munan Yuan. Email: mnyuan@hfcas.ac.cn

ABSTRACT

The existing multipath routing in Software Defined Network (SDN) is relatively blind and inefficient, and there is a lack of cooperation between the terminal and network sides, making it difficult to achieve dynamic adaptation of service requirements and network resources. To address these issues, we propose a multi-constraint path optimization scheme based on information fusion in SDN. The proposed scheme collects network topology and network state information on the network side and computes disjoint paths between end hosts. It uses the Fuzzy Analytic Hierarchy Process (FAHP) to calculate the weight coefficients of multiple constrained parameters and constructs a composite quality evaluation function for the paths to determine the priority of the disjoint paths. The SDN controller extracts the service attributes by analyzing the packet header and selects the optimal path for flow rule forwarding. Furthermore, the service attributes are fed back to the path composite quality evaluation function, and the path priority is dynamically adjusted to achieve dynamic adaptation between service requirements and network status. By continuously monitoring and analyzing the service attributes, the scheme can ensure optimal routing decisions in response to varying network conditions and evolving service demands. The experimental results demonstrated that the proposed scheme can effectively improve average throughput and link utilization while meeting the Quality of Service (QoS) requirements of various applications.

KEYWORDS

SDN; multi-constraint path; information fusion; FAHP

Abbreviation

SDN	Software defined network
FAHP	Fuzzy analytic hierarchy process
QoS	Quality of service
kMDPC	k maximal link disjoint paths with cost
SR	Segment routing
ESR-SDN	Efficient segment routing in software-defined network
MCOP	Multiple constraint optimal paths
EARMLP	Energy-aware routing multi-level and mapping problem

SDN-MCQDP	Multi-constraint qos dual-path based on software-defined network
OSPF	Open shortest path first
ECMP	Equal-cost multi-path routing
DRL	Deep reinforcement learning
LLDP	Link layer discovery protocol
D-ITG	Distributed internet traffic generator
VR	Virtual reality

1 Introduction

In recent years, with the rapid development of Internet technology, various new network applications and services have shown explosive growth. To improve the open innovation of traditional network architecture and promote the optimization and selection of routing strategies, Software Defined Network (SDN) technology is a new network paradigm that collects the network state periodically and obtains the service demand from a global perspective [1,2]. It can also calculate the optimal path and then issue flow rules to the forwarding plane to route data flows, further improving the flexibility of network resource allocation [3,4]. With the diversity of network services, it is increasingly important to achieve the rational utilization of network resources, effectively avoid network congestion, and better improve the overall quality of network services. This puts higher requirements on SDN routing strategies. Therefore, the multipath routing strategy under SDN architecture has become one of the current research hotspots.

Currently, most multipath high-availability transmission solutions under SDN architecture tend to be designed and optimized on the terminal, mainly in multi-path management [5,6] and path selection [7,8]. These approaches face problems such as the high complexity of multi-constraint routing algorithms and lack of network global awareness when solving the problem of multi-path high availability transmission. As a result, they lead to decision lag, reduce network resource utilization, and greatly affect the implementation efficiency and Quality of Service (QoS) of upper-layer services in the network.

In this work, we focus on multi-constraint path optimization in SDN to address the issues mentioned above. A multipath routing optimization scheme based on information fusion is proposed to meet the QoS requirements of emerging applications. The proposed scheme constructs a composite quality evaluation function based on the global network state based on the global visibility of the SDN controller to the switches and other devices, and uses the Fuzzy Analytic Hierarchy Process (FAHP) [9] to determine the weight coefficients of each parameter and compute the path priority. Based on the service attributes and QoS requirements, iterative optimization of routing strategies is carried out to meet the load balancing needs within the network. The contributions of this paper are summarized as follows:

- Design an SDN transmission architecture that combines network side and terminal side optimization, including three modules: Path collection, path composite quality evaluation, and routing decision.
- Perceive the link state of the data plane layer, evaluate the composite quality of paths, and sort the paths based on FAHP. Select the route with the highest priority as the optimal route, and adjust based on the service attributes to update the iteration path priority until the route ends.
- Use Mininet to construct different network topologies for experiments. The simulation experiment results show that compared with the synchronous method, the proposed method can

improve network throughput and link utilization, and can meet the QoS requirements of different services.

The rest of the paper is structured as follows. Section 2 summarizes the related work. In Section 3, we establish a multi-constraint path optimization scheme based on information fusion, using FAHP to evaluate path priority and iterating routing based on service attributes. The experimental results are presented in Section 4. We conclude the paper in Section 5.

2 Related Work

The research on multipath transmission in SDN faces challenges such as complex multi-constraint routing algorithms and the inability to fully utilize the advantages of SDN's global network parameter acquisition [10,11]. Consequently, further investigation and improvement in these areas within the domain of SDN-based multipath transmission.

Abe et al. [12] proposed a modeling algorithm to find k Maximal link Disjoint Paths with Cost (kMDPC) to provide disjoint paths for SDN data planes and help in switch load-balancing. Lee et al. [13] designed an efficient routing algorithm with Segment Routing (SR) for SDN (ESR-SDN) to meet the bandwidth requirements of routing requests. The algorithm considers traffic load balancing and reduces the extra cost of packet header size. Diego et al. [14] proposed a solution that enables the end-to-end QoS based on the queue support in OpenFlow, allowing operators with an SDN-enabled network to efficiently allocate the network resources according to the users' demands. Forests et al. [15] introduced a classification method for different application types and corresponding network resource allocations, which is implemented by computing programmable labels in the control plane to provide differentiated QoS guarantees for various application types. Henni et al. [16] constructed a coherent network framework in SDN that improves network video transmission performance by ensuring QoS for prioritized flows. Parsaei et al. [17] used a hybrid model of type-2 fuzzy systems and a Rhododendron optimization algorithm to find the optimal and reliable path while satisfying high-quality multi-constraint QoS. Kamath et al. [18] added a component Multiple Constraint Optimal Paths (MCOP) for packet forwarding, which selects multiple optimal paths to forward packets while considering the QoS requirements of the traffic. Ibrahim et al. [19] proposed and implemented an Energy-Aware Routing Multi-Level and mapping Problem (EARMLP) algorithm to minimize the overall power consumption in SDN-based core networks. BinSahaq et al. [20] presented a new QoS-aware routing algorithm based on SDN, which uses the state information obtained from previous optimization iterations to obtain network subgraphs of small size to avoid additional useless Dijkstra calls. Kamboj et al. [21] proposed a dynamic multipath routing scheme with perceivable QoS for improving the QoS of high bandwidth applications in SDN networks. Guo et al. [22] proposed a Multi-Constraint QoS Dual-Path routing optimization algorithm based on SDN (SDN-MCQDP), using reverse links to remove redundant dual-path links that meet multi-constraint QoS and ensure data transmission after link failure.

With the development of artificial intelligence technology, routing strategies for SDN multipaths are gradually being applied [23–25]. Hu et al. [26] proposed an intelligent driving network architecture based on SDN, which optimizes network latency and throughput at the data plane. Compared with traditional routing algorithms Open Shortest Path First (OSPF) and Equal-Cost Multi-Path routing (ECMP), it effectively improves the network load balancing ability. Huang et al. [27] proposed a near-optimal flow control method for QoS optimization in SDN, which uses the Deep Reinforcement Learning (DRL) algorithm to solve the multipath routing problem in SDN. Under the architecture of the Internet of Things, Sun et al. [28] combined multiple machine-learning algorithms to identify

and classify flows and selected the optimal transmission path for different types of applications based on multi-constrained QoS. These artificial intelligence-based routing optimization methods require a large amount of label data for training during the training process, which requires high computational complexity and still has problems in terms of routing accuracy, generalization, and fault adaptability.

The above SDN routing solutions have improved network performance such as transmission delay, QoS, and packet loss rate, but there is still room for improvement in terms of routing calculation difficulty, link utilization, and QoS satisfaction with different services. Therefore, we propose a multi-constraint path optimization scheme based on information fusion, utilizing FAHP for routing evaluation and selection, and adjusting weight coefficients to update the iterative path priority based on the service attributes. Ultimately, achieving adaptive matching between requirements and network resources reduces network load pressure.

3 Scheme Architecture

Utilizing the advantages of SDN technology, design path optimization with multi-constraints, so that routing strategies can be adaptively adjusted based on service requirements and the supply-demand relationship between network resources. The overall architecture of the optimization scheme is shown in Fig. 1, which mainly includes path collection, path composite quality evaluation, and routing decisions.
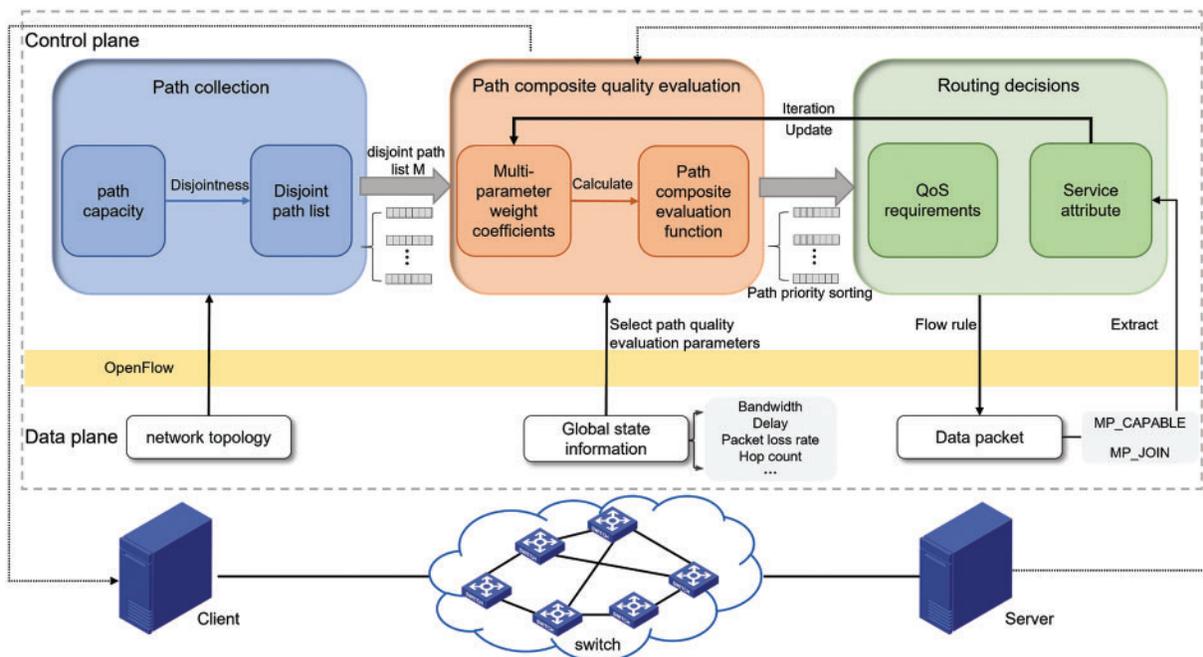
Figure 1: The framework of a multi-constraint path optimization scheme based on information fusion

The SDN control plane and data plane communicate through the OpenFlow protocol. The control plane obtains global state information by sending messages to the switches in the data plane and storing basic topology relationships. In path collection, select the optimal path that satisfies the link state, delete the current optimal path based on the jointness constraint, and iteratively update the topology relationship to obtain a disjoint path list. The path composite quality evaluation module

perceives the global state and selects multiple path quality evaluation parameters based on the actual link state. Calculate the weight of each parameter through FAHP and arrange the paths in descending priority order. Based on the above, the routing decision module extracts the packet header for analysis and matching, and then selects the route with the highest priority as the optimal route. This module provides feedback on service attributes to the path composite quality evaluation module and adjusts the weight coefficients to update the iterative path priority until the routing process is completed. Moreover, extending the SDN control function to terminals can support information exchange between the control plane and terminal devices. After parsing the IP information of terminals, the control plane can send instructions to terminal devices to adjust services, achieving dynamic adaptation of service requirements and network resources.

3.1 Path Collection

In the initial stage, SDN controller discovery topology and link state. The controller sends flow entries in the form of flooding to all nodes connected to it in the network, requesting port information. Switches send Packet_Out packet containing Link Layer Discovery Protocol (LLDP) to the underlying nodes. When a node receives an LLDP packet sent by a neighboring node, it will send a Packet_In packet to the SDN controller containing its own and neighboring node link information. Fig. 2 shows the process of topology discovery through the LLDP protocol. The SDN controller completes topology discovery and topology management based on the feedback information. Using network tools to store topology maps G (V, E), where V is the set of network nodes, and E is the set of links between these nodes.
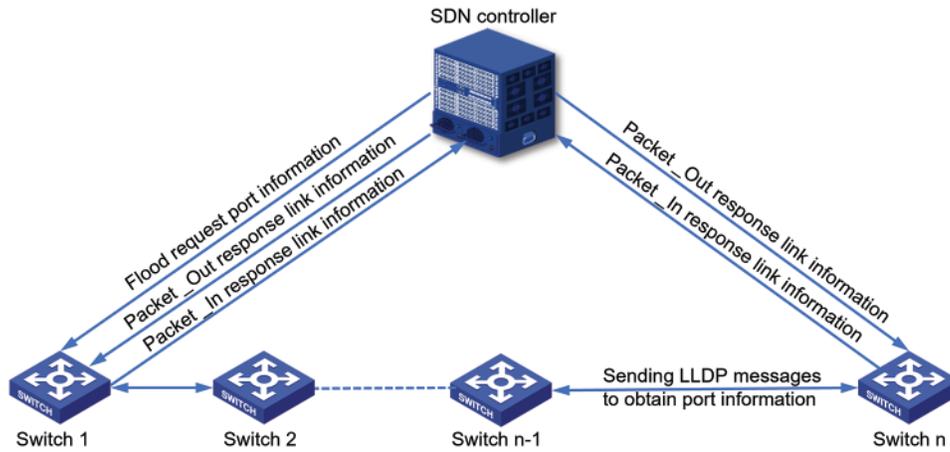
Figure 2: Topology discovery process

Firstly, the disjoint path list is selected in the basic topology relationship. Assuming that the network is composed of a set of switching devices $S = \{s_1, s_2, \ldots, s_{N-1}\}$, with flows $F = \{f_1, f_2, \ldots, f_n\}$ passing through the SDN controller for path calculation and forwarding, the objective function expression of the disjoint paths is as follows:

$$\text{Minimize} \sum_{f_i, f_j \in s_f, i < j} |IS(f_i, f_j)| \tag{1}$$

where IS represents the set of disjoint switches with different paths, S_f represents the available paths of flow. The constraint conditions for multi-objective functions are joint constraint and link state

constraint, respectively, expressed as follows:

$$\forall R \in s_f, \forall e \in R, \text{s.t } e \notin R', \exists R' \in s_f \setminus R \tag{2}$$

$$\forall r_l \in L, r_l = \begin{cases} 1, & \text{if } l \in R \\ 0, & \text{if } l \notin R \end{cases} \tag{3}$$

where R represents the path, and r_l represents the available state of link l.

To meet the real-time requirements of multi-constraint QoS for applications, it is also necessary to monitor the network status. By using timestamp counters to send data flow detection between nodes, regularly polling and storing the number of bytes sent by the port, network status information such as bandwidth, latency, and jitter is calculated. The SDN controller calculates the remaining bandwidth Br based on switch port statistics. Firstly, obtain the capacity C of each port of each switch, and then calculate the port transfer rate Tr based on traffic statistics. The remaining bandwidth Br is represented as:

$$Br = C - Tr \tag{4}$$

For each path, the available bandwidth is the minimum remaining bandwidth value of all switch ports. Therefore, the available bandwidth Br_k for each path is:

$$Br_k = \min \{Br_l | 1 \leq l \leq k\} \tag{5}$$

where k represents the k-th path between two terminals.

The delay of each link mainly includes two parts. Taking the SDN controller, switch A, and switch B as examples, the first part is the delay of T_{AB} (controller A-B controller) and T_{BA} (controller B-A controller) calculated by the difference in timestamps between two data packets. Another delay is T_A (controller A controller) and T_B (controller B controller) between the controller and switch, which use the timestamp of the echo message defined in the OpenFlow protocol to measure. The delay between A and B is:

$$D_{AB} = \frac{T_{AB} + T_{BA} - T_A - T_B}{2} \tag{6}$$

Then, the delay of each link is:

$$D_l = \sum_{i=1}^{n-1} D_i \tag{7}$$

The SDN controller utilizes the Dijkstra algorithm to calculate the optimal path between the source node S and the destination node D, considering bandwidth as the weight. To obtain disjoint paths, the current calculated optimal path is deleted from the original graph, and a new topology is obtained. It is iterated sequentially until there are no available paths, and all disjoint paths are saved in the path list. Algorithm 1 describes the procedure for collecting a disjoint path list based on bandwidth.

Algorithm 1: Path collection based on available bandwidth

Input: Network topology G = (V, E), source node S, target node D, number of paths N
Output: Disjoint path list M
1:      $Br = C - Tr$

(Continued)

Algorithm 1 (continued)

2:     $\delta = Br/C$ //   *Using Br as the calculation weight for each link*
3:     **Function** Dijkstra (S, D, G)
4:       selected path list M = Ø;
5:     **for** i from 1 to N **do**
6:      path list Θ = Ø;
7:      current node O←S;
8:     **while** O ≠ D **do**
9:        nodes list Λ = sort nodes of O by δ;   //   *Node deletion and selection*
10:       Choose a node O′ ∈ Λ, which satisfies that (O, O′) is not in M;
11:       save link (O, O′) in Θ and M;
12:       G(V, E)←G(V, E′;);
13:      **if** G (V, E′;) has routing node *v* with no link connection **then**
14:        Delete routing node *v*   in G (V, E′;);
15:        G(V′;, E′;)←G(V, E′;);   //   *Update topology*
16:       **end if**
17:     **end while**
18:       save Θ in M;
19:     **end for**
20: **Return** disjoint path list M

If the total available bandwidth of all paths is less than the bandwidth required by the service, the network will make every effort to provide the service. However, in general, the bandwidth provided by the network is greater than the service demand bandwidth. Therefore, under the premise of meeting the bandwidth demand, it is necessary to evaluate the composite quality of available paths and prioritize selecting paths with better quality.

3.2 Path Composite Quality Evaluation

When the bandwidth of all transmission paths exceeds the bandwidth required by the service, network resources are not fully utilized; When the bandwidth of all transmission paths is less than the bandwidth required by the service, the network will experience congestion, which affects the QoS. Dijkstra is a classic algorithm for finding the shortest path, but it is only applicable to paths that have already met the constraint conditions, so it has significant limitations. How to find the optimal transmission path that meets the constraints of flows in a shorter time has become a challenge.

Considering the influence of various factors on the transmission path, there exists a competitive relationship among the optimization objectives and a lack of unified calculation standards. To address this issue, we introduce the FAHP to analyze the quality of paths. FAHP is a multi-criteria decision analysis method that combines fuzzy mathematics and the analytic hierarchy process, enabling the handling of decision-making problems characterized by uncertain information and strong fuzziness. In FAHP, the decision problem is decomposed into multiple sub-objectives or criteria to construct a hierarchical network model. Then, using expert judgment to fuzzily the judgment matrices at each level, a fuzzy hierarchical structure model is obtained, and the weight vectors of each level are calculated. Finally, the ranking and priority of each decision scheme in the entire hierarchical structure network are solved.

By employing the FAHP method, we can effectively address the mutual influence and dependency relationships between decision factors, leading to more objective, scientific, and reliable decision outcomes. The usage of FAHP enhances the robustness and accuracy of the path optimization process, enabling a composite analysis of the path quality in SDN environments.

Based on FAHP, determine decision objectives and associated factors. the evaluation criteria for path quality are established by considering the essential parameters associated with various service attributes. In this study, four evaluation parameters, namely bandwidth, delay, hop count, and packet loss rate, are selected and represented as a set U, where $U = (U1, U2, U3, U4)$. Fig. 3 illustrates the implemented hierarchical model.
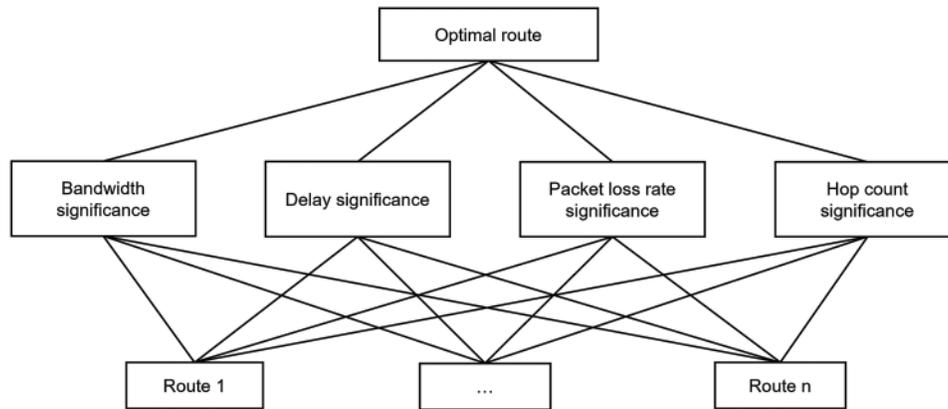
Figure 3: A hierarchical model for path optimization

Assume that the normalization value of the j-th (j = 1,2,3,4) evaluation parameter for the i-th (i = 1,2,3…n) path is denoted as x_{ij}, representing a paired selection matrix. This matrix signifies the relative importance assigned to each evaluation parameter for paths, as shown in Eq. (8):

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ \cdots & \cdots & \cdots & \cdots \\ x_{n1} & x_{n2} & x_{n3} & x_{n4} \end{bmatrix} \tag{8}$$

The fuzzy matrix $X = (x_{ij})_{n \times n}$ is constructed by comparing the relative importance of evaluation parameters pairwise, where $0 \leq x_{ij} \leq 1$ and $x_{ij} + x_{ji} = 1$. The scale and definition are shown in Table 1.

Table 1: The value and its definition and explanation

Value	Definition	Explanation
0.5	Identical	Parameter a is equally important as parameter b.
0.6	Slightly important	Parameter a is slightly more important than parameter b.
0.7	Clearly important	Parameter a is clearly more important than parameter b.

(Continued)

Table 1 (continued)

Value	Definition	Explanation
0.8	More important	Parameter a is much more important than parameter b.
0.9	Extremely important	Parameter a is extremely important than parameter b.
0.1~0.4	Inverse comparison	The importance of parameter b relative to parameter a, and the sum of the importance of parameter b relative to parameter a is 1.

If the fuzzy matrix $X = (x_{ij})_{n \times n}$ satisfies the condition:

$$\forall i, j, k, x_{ij} = x_{ik} - x_{jk} + 0.5 \tag{9}$$

Then, the resulting fuzzy matrix X is referred to as a fuzzy consistent matrix, where $x_{ij} = 0.5$, represents that parameter i and parameter j are considered equally important. If $0 \leq x_{ij} \leq 0.5$, represents that parameter j is more important than parameter i, the smaller the value of x_{ij}, the greater the importance of parameter j relative to parameter i. Conversely, if $0.5 \leq x_{ij} \leq 1$, indicates that parameter i is more important than parameter j, the larger the value of x_{ij}, the greater the importance of parameter i relative to parameter j.

If the fuzzy matrix does not meet Eq. (9), it needs to be transformed into a fuzzy consistent matrix. The matrix $X = (x_{ij})_{n \times n}$ needs to be summed by rows:

$$X_i = \sum_{k=1}^{n} x_{ik}; \ i = 1, 2, \cdots, n \tag{10}$$

The elements in the fuzzy matrix are mathematically transformed to obtain the fuzzy consistent matrix:

$$x_{ij}' = (x_i - x_j)/2n + 0.5 \tag{11}$$

Based on the judgment matrix, calculate the priority weight of the importance of the parameters in this level for a certain parameter in the previous layer. The weight coefficient w of parameter i is:

$$w_i = \frac{1}{n} - \frac{1}{2\alpha} + \frac{\sum_{j=1}^{n} x_{ij}'}{n\alpha}, i = 1, 2 \ldots n \tag{12}$$

where $\alpha = (n-1)/2$, the weight coefficient w should satisfy the constraint conditions shown in Eq. (13):

$$\begin{cases} 0 \leqslant w_j \leqslant 1, (j = 1, 2, 3, 4) \\ \sum_{j=1}^{4} w_j = 1 \end{cases} \tag{13}$$

The composite evaluation function of the path is obtained by summing up the weights of the evaluation parameters and the normalized values of the parameters.

$$P(i) = \sum_{j=1}^{4} w_j x_{ij}, \ (i = 1, 2, \cdots, n) \tag{14}$$

This function integrates the importance weights assigned to each evaluation parameter with their normalized values, thereby providing a comprehensive assessment of the path's quality. By adjusting the weight coefficients, it is possible to modify the relative importance of different evaluation parameters in the composite quality evaluation function of the path. This allows for meeting the specific link requirements of different services.

Based on the above, calculate the initialization weights of various evaluation parameters. Establish an initial weight table for path service quality evaluation based on Table 1, as shown in Table 2.

Table 2: The analysis of path composite quality weight

Path composite quality evaluation	Bandwidth	Delay	Packet loss rate	Hop count
Bandwidth	0.5	0.55	0.6	0.7
Delay	0.45	0.5	0.55	0.6
Packet loss rate	0.4	0.45	0.5	0.55
Hop count	0.3	0.4	0.45	0.5

The matrix corresponding to the path composite quality evaluation can be obtained from Table 2:

$$X = \begin{bmatrix} 0.5 & 0.55 & 0.6 & 0.7 \\ 0.45 & 0.5 & 0.55 & 0.6 \\ 0.4 & 0.45 & 0.5 & 0.55 \\ 0.3 & 0.4 & 0.45 & 0.5 \end{bmatrix} \tag{15}$$

According to Eqs. (10) and (11), calculate the conversion to a fuzzy consistent matrix:

$$X = \begin{bmatrix} 0.50 & 0.542 & 0.575 & 0.616 \\ 0.458 & 0.50 & 0.533 & 0.575 \\ 0.425 & 0.467 & 0.50 & 0.542 \\ 0.384 & 0.425 & 0.468 & 0.50 \end{bmatrix} \tag{16}$$

Further, calculate the initialization priority weights for each parameter according to Eq. (12) as follows: $w_1 = 0.288$, $w_2 = 0.261$, $w_3 = 0.239$, $w_4 = 0.212$. From this, the importance of different indicators for evaluating the quality of path composition can be determined.

3.3 Routing Decision

The routing decision module adjusts the path evaluation parameters and corresponding parameter weights based on the service attributes and adjusts the paths for different flows through global optimization iterations. Algorithm 2 demonstrates the process of SDN controllers assigning deterministic paths. When the client sends the first packet to the switch, the switch uses the Packet_In in the OpenFlow protocol message to send the message to the SDN controller. The SDN Controller extracts

packet header options, if it is MP_ CAPABLE, select the path with the highest ranking from the path list, and send the flow table of that path to the corresponding switch. If the header option for extracting data packets is MP_JOIN, the determination of whether it is a new sub-flow can be based on the token. If the sub-flow belongs to the same connection as the previous sub-flow, the paths in the path list can be distributed sequentially. Otherwise, update the path list and allocate paths in order. For paths that require re-routing, trace back to the path composite quality evaluation module based on the original forwarding path and reevaluate the path.

Algorithm 2: Routing decision

Input: MP_CAPABLE, MP_JOIN, Disjoint path list M;
Output: Forwarding path
1:     **if** Packet header = MP_ CAPABLE **then**
2:     Select the path with the highest ranking from the path set M and assign it to the sub-flow;
3:      **end if**
4:      **if** Packet header = MP_JOIN **then**
5:      **if** the token $\neq$ the existing token **then**
6:         Create a new sub-flow connection;
7:         Select the next path from M;
8:     **else**
9:          Select the next path from M;
10:    **end if**
11: **end if**

Based on the aforementioned algorithm, the overall flowchart of the proposed scheme is depicted in Fig. 4. The scheme iteratively updates the routing table based on various service attributes until the routing process is completed. In the case of paths requiring re-routing, pertinent path information is extracted from the path monitoring list. Considering the current forwarding conditions within the network, the weight coefficients of the evaluation parameters are recalculated. Subsequently, the available path list is updated and transmitted to the routing decision module. This ensures that the routing process incorporates the most recent information, allowing for efficient and effective path selection based on the evaluation criteria.

At the end of each routing iteration, the QoS parameters for each route are updated based on the utilization of the initial parameters. This ensures that the impact of the previous round is considered and applied to the subsequent rounds. Through iterative routing updates, the path availability is enhanced, while simultaneously achieving adaptive alignment between service requirements and network resources. This scheme facilitates the continuous improvement of the routing process, as the QoS parameters are dynamically adjusted based on the evolving network conditions and the changing demands of the service.

4 Simulation Experiments and Result Analysis

4.1 Simulation Setup

To evaluate the performance of the proposed scheme, we conduct experiments on Mininet. The network controller uses Ryu v4.3, and the OpenFlow protocol version is 1.4. The OpenFlow switches are used to set up multiple paths, and bandwidth is controlled with the help of their meter function. We used the Iperf and Distributed Internet Traffic Generator (D-ITG) to model the flow requests in the network and generate the experimental packets using the uniform distribution method. Each host

resource sends data packets to the destination, sets the controller to update the network status every 25 s, and issues a new flow table to the nodes. The capacity of each available communication link is set to 1 Gpbs, and the static buffer of the switch port is set to 300 KB. Each experiment lasts for 100 s and is repeated 10–20 times. Simulate the topology diagram using four real network topology cases from the Internet Topology Zoo dataset [29], where the Global Center topology consists of 9 nodes and 36 links; The ATT North America topology has a total of 25 nodes and 57 links; TaTa topology has a total of 145 nodes and 194 links; The Cogent topology consists of 197 nodes and 245 links.
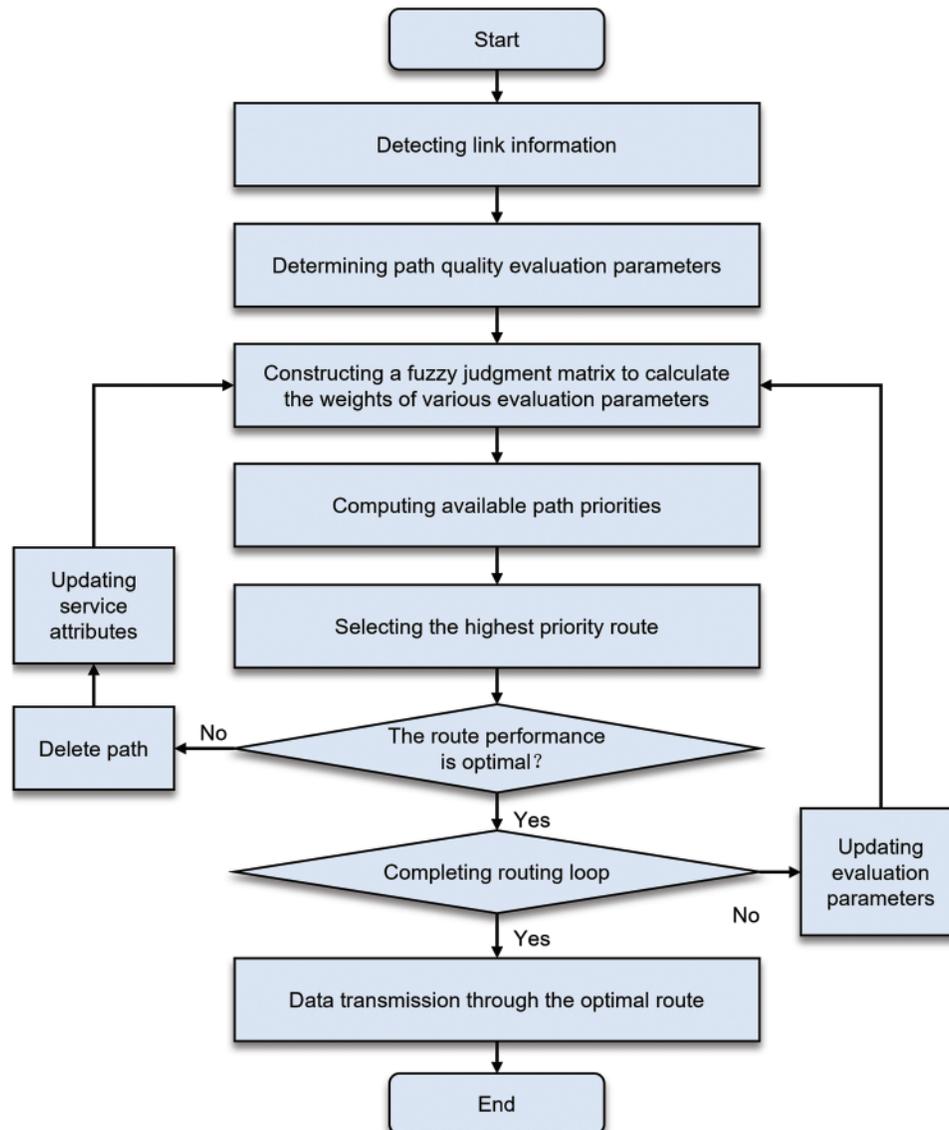
Figure 4: The overall process of the path optimization scheme

The performance of the proposed scheme is validated through a comprehensive analysis of throughput, link utilization, latency, and QoS satisfaction. Simulation-based comparative experiments

are conducted, comparing the proposed scheme with routing optimization schemes ESR-SDN [13], MCOP [18], and SDN-MCQDP [22], all based on the SDN architecture.

4.2 Throughput and Link Utilization

The throughput of schemes under different loads was compared, with experimental settings ranging from 100 to 500 requests. As depicted in Fig. 5, the proposed scheme demonstrates the superiority of its path planning in achieving globally optimal transmission performance across different numbers of requests. When the number of requests is 100, there is minimal disparity in average throughput among the four schemes. However, as the number of requests increases, the average throughput increases significantly. Notably, the proposed scheme exhibits a slightly higher average throughput compared to the benchmark schemes. This can be attributed to the relatively low dependency between different paths in the proposed scheme, allowing for routing selection based on individual requests. Such a characteristic enables the proposed scheme to optimize path planning and enhance overall transmission performance as the number of requests escalates.
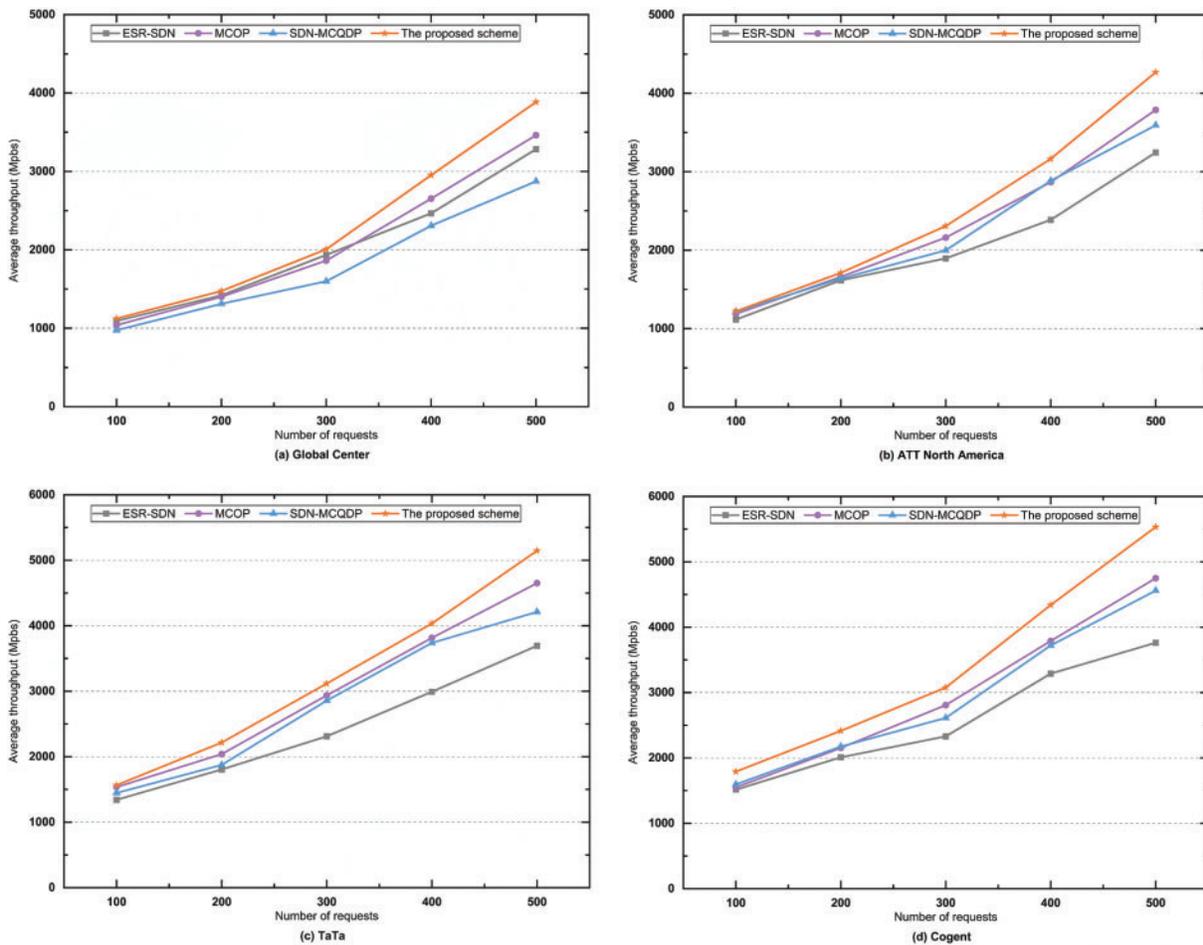
Figure 5: Average throughput for different network scales

The amount of link utilization is equal to the total amount of consumed bandwidth by each link to the total available capacity in the network. The link utilization under different topologies

is shown in Figs. 6a and 6b, corresponding to 100 and 500 requests, respectively. As evident from Fig. 6, the proposed scheme exhibits a higher average link utilization compared to the other three schemes, highlighting its advantage in achieving global path resource utilization. ESR-SDN, due to its limitations in returning the optimal path and errors in calculating the optimal path, exhibits lower link utilization. SDN-MCQDP partially mitigates link congestion and enhances link utilization to some extent through dual-path routing. When the number of requests is 500, as shown in Fig. 6b, the average link utilization of MCOP surpasses that of SDN-MCQDP due to the capability of selecting multiple optimal paths simultaneously for data forwarding. The proposed scheme prioritizes the links through multi-parameter analysis, ensuring maximum transmission efficiency and reducing resource waste of available bandwidth. Thus, while satisfying the bandwidth requirements of network applications, the proposed scheme effectively improves link utilization. These results highlight the performance advantage of the proposed scheme in terms of throughput, providing evidence of its ability to enhance overall network transmission performance through optimized allocation of network resources.
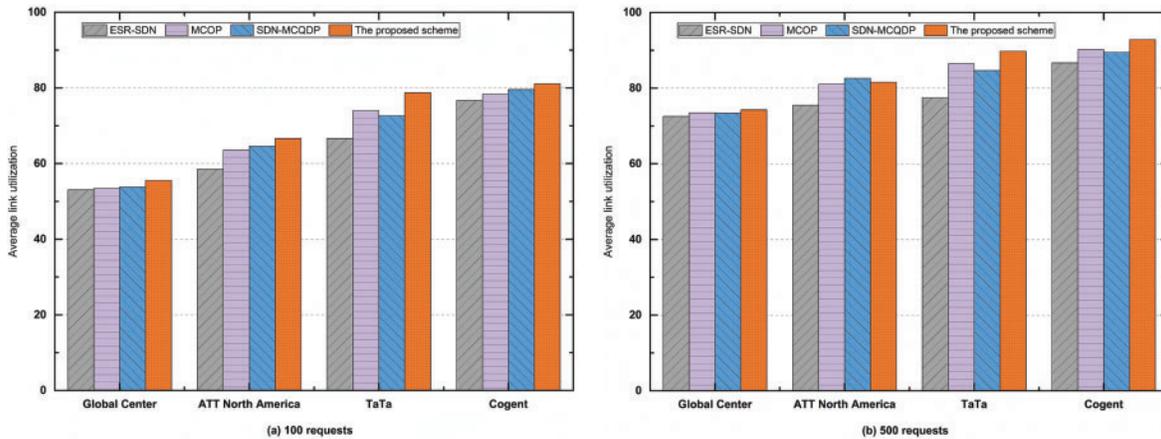
Figure 6: Comparison of average link utilization

4.3 Path Costs

The average routing cost in a network is estimated by the utilization of all paths for the corresponding traffic requests. Fig. 7 shows the routing costs of these schemes for requests, with Figs. 7a–7d representing different network scales. When the number of requests is less than 50, all four schemes choose the path with the shortest distance between the controller and the switch. Therefore, the routing costs for different network scales are almost equal. In the same network scale, the routing cost increases as the number of requests increases. Similarly, as the network scale increases, the traffic demand also increases and the routing costs also increase. In Figs. 7c and 7d, the routing cost of the MCOP scheme exceeds that of SDN-MCQDP and is marginally lower than that of our proposed scheme. The proposed scheme incurs a higher total routing cost as it necessitates the controller to generate path priorities based on the network topology to enhance link utilization. However, the differences in routing costs are 8.3%, 3.7%, and 4.6% in comparison to the ESR-SDN, MCOP, and SDN-MCQDP, respectively. If the requirement is to promptly select the optimal path for data forwarding under multi-constraint QoS conditions, and a slight increase in routing cost is acceptable, the proposed scheme remains the optimal choice.
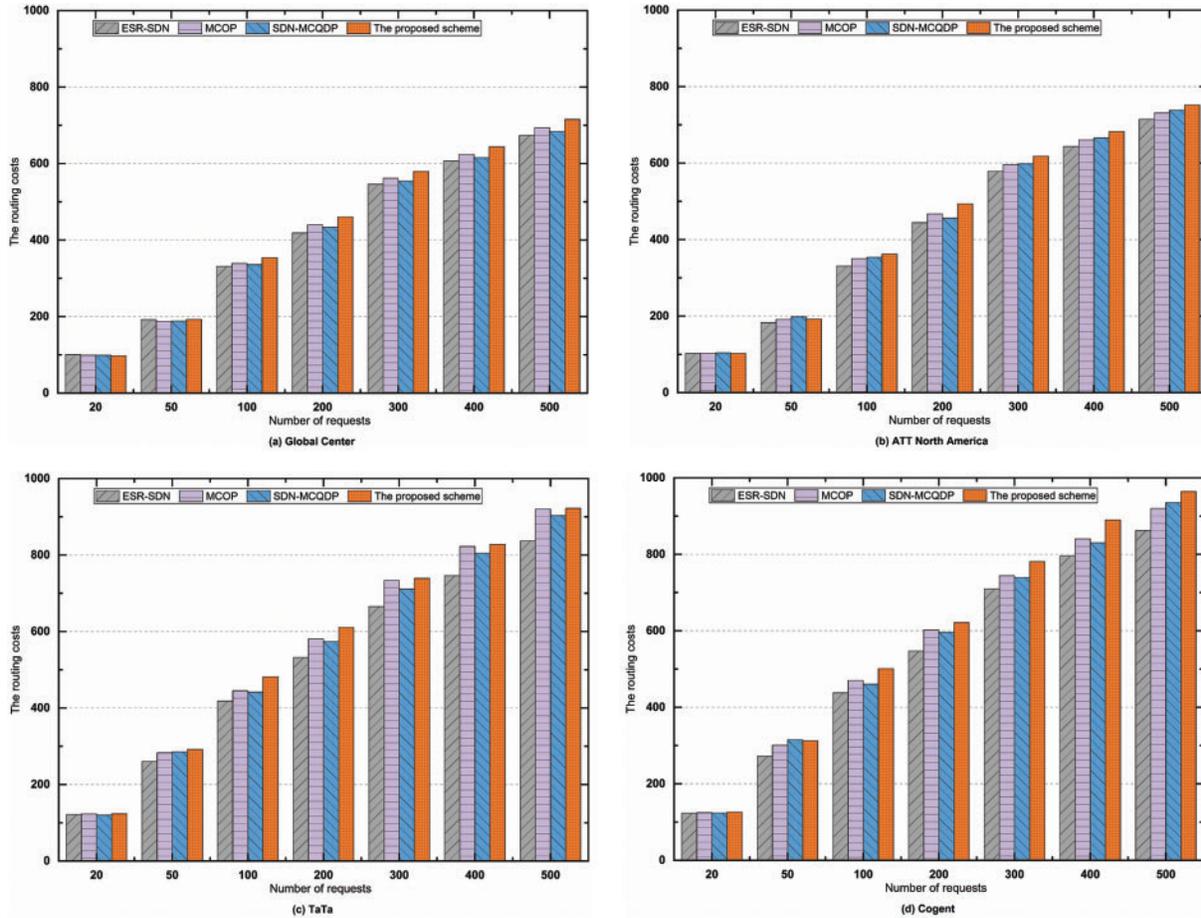
Figure 7: All path costs for different network scales

4.4 Latency

To test the trade-off between delay and routing cost, we further observe the delay of the path. Compare the performance of the proposed scheme in terms of latency at different network scales based on the relative difference percentage of latency in the link. The relative difference percentage of delay η_t is calculated as follows:

$$\eta_t = \frac{d}{D} \tag{17}$$

where d represents delay constraint. The smaller the value of η_t, the better the latency performance.

From Fig. 8, it can be seen that the proposed scheme has advantages in terms of delay compared to the other three schemes. Fig. 8a depicts the proposed scheme demonstrating a notable advantage in terms of latency, exhibiting relative percentage improvements of 2.9%, 1.2%, and 1.4% compared to ESR-SDN, MCOP, and SDN-MCQDP schemes, respectively. In Fig. 8b, a marginal difference between MCOP and SDN-MCQDP, with their performance reversing as the number of nodes increases. the proposed scheme maintains its dominance, showcasing a relative percentage improvement of 3.7% over ESR-SDN. The ESR-SDN algorithm has the lowest relative percentage

of delay because it does not directly return the optimal path, requiring additional calculations to determine the path that satisfies multiple constraints. The SDN-MCQDP reduces the repeated node calls to some extent by generating directed acyclic graphs based on the destination node. In contrast, the proposed scheme calculates all feasible paths from intermediate nodes to destination nodes during the path composite quality evaluation process and prioritizes them, effectively reducing link latency.
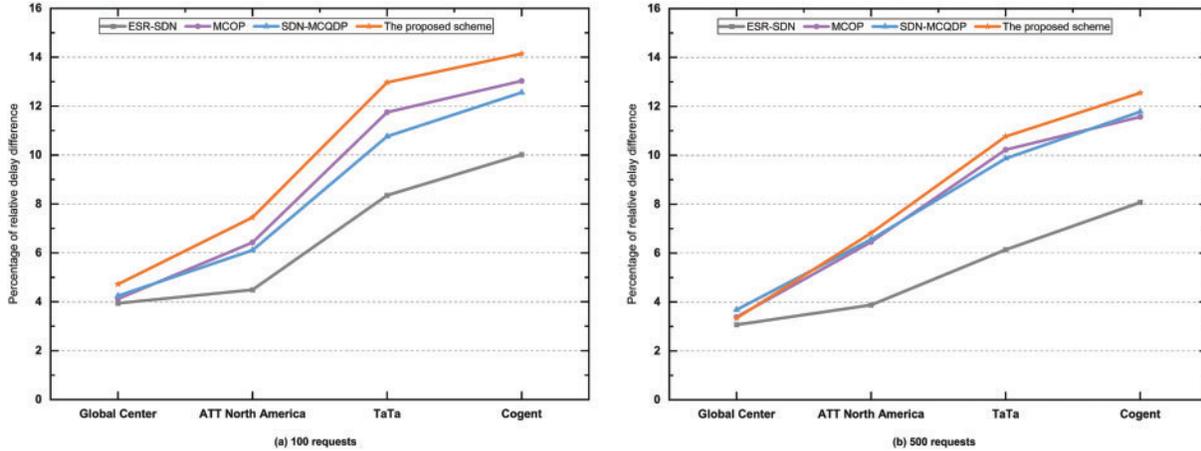
Figure 8: The percentage of relative delay difference

4.5 QoS Satisfaction

We conducted QoS satisfaction testing on a range of network applications, aiming to assess their performance under different QoS requirements. Table 3 provides an overview of the QoS requirements for five commonly encountered real network applications that were included in our study.

Table 3: The QoS requirements for different applications

Application type	Bandwidth/Mbps	Delay/ms	Packet loss rate/%	Jitter/ms
VoIP	0.1	140	1.0	30
Online game	3.0	100	0.5	20
Video conference	0.5	120	1.0	30
Online VR	25.0	70	1.0	20
Remote control	0.2	150	0.5	30

The experimental results are presented in Fig. 9, illustrating the percentage of QoS satisfaction achieved by the four schemes for various network sizes. The proposed scheme consistently achieves the highest QoS satisfaction rate, surpassing 91% for diverse application requirements. We observed that the QoS satisfaction of different applications exhibits a slight decrease with an increase in network nodes. However, there are also minor variations in some applications due to their varying sensitivity towards different parameters.
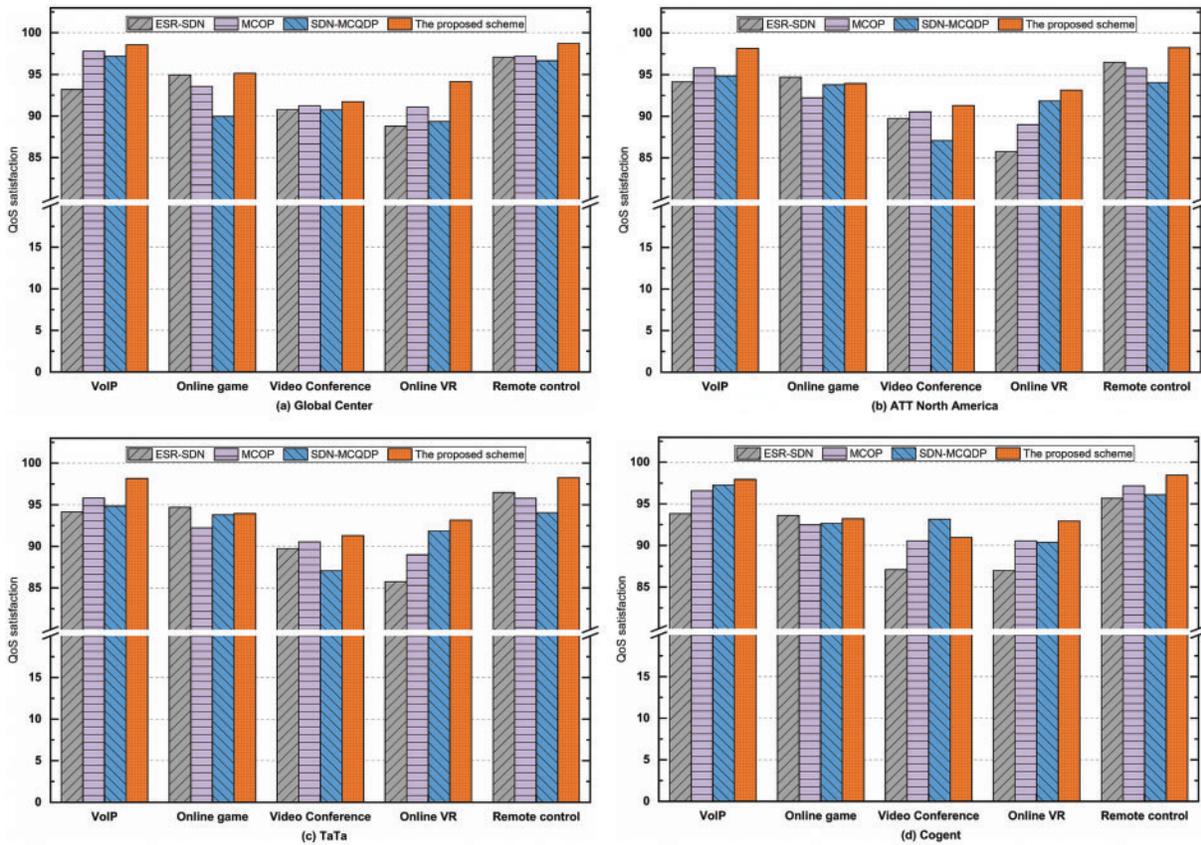
Figure 9: The QoS satisfaction of different applications

In Fig. 9a, the MCOP scheme outperforms the ESR-SDN and SDN-MCQDP schemes, providing better QoS satisfaction. However, its QoS satisfaction slightly diminishes as the number of links increases, as shown in Figs. 9b–9d. This can be attributed to the calculation of weights along the path in the MCOP scheme, which aims to select the path with the maximum available bandwidth, thereby improving the quality of service. The SDN-MCQDP scheme exhibits different trends for different applications in various network topologies. This is due to the utilization of a reverse link pruning algorithm in the SDN-MCQDP scheme, which enables the acquisition of two disjoint optimal paths, providing backup options to meet the requirements of different applications. However, the quality of service for applications can be influenced by the routing computation time, leading to fluctuations. The proposed scheme calculates and analyzes network state information, taking into account various QoS constraints such as delay, hop count, bandwidth, and packet loss rate. By comparing service requirements with the available network capacity within the SDN controller, the optimal path is computed, thus meeting the satisfaction of a majority of network applications. Through comparisons, the proposed solution demonstrates its effectiveness in satisfying QoS demands and meeting the diverse requirements of network services, while also providing insights into the performance characteristics of different schemes in different network scenarios.

5 Conclusion and Future Work

We have developed and implemented a multi-constraint path optimization scheme that utilizes information fusion to address the high complexity issues related to multi-constraint QoS routing algorithms in SDN architecture. By linking state awareness at the data plane, the FAHP is used to integrate multi-constraint path quality evaluation parameters, generate a path composite quality evaluation model, and prioritize the optimal path list. Additionally, we dynamically adjust the weight of path quality evaluation parameters based on the service attributes. Through global optimization, we iteratively fine-tune paths for different flows, striving to achieve adaptive alignment between business requirements and network resources. The simulation results demonstrate that the proposed optimization scheme effectively enhances the average throughput of data streams, improves link utilization, and exhibits adaptability to various network topology types. These findings highlight the capability of the proposed scheme to meet the QoS requirements of a majority of network applications.

In future work, one of the main directions for improvement is to consider additional QoS indicators as path quality evaluation parameters to further optimize the proposed scheme. Moreover, it is essential to conduct testing in more complex simulation scenarios and network platforms to expand node scale and make network topology more complex, to study its scalability in real situations.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Jinlin Xu, Wansu Pan; data collection: Jinlin Xu; analysis and interpretation of results: Longle Cheng, Wansu Pan; draft manuscript preparation: Jinlin Xu, Munan Yuan; project administration: Haibo Tan, Xiaofeng Li. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data used to support the findings of this study are included within the article.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] L. S. Wang, P. Yi, T. Hu, Y. M. Jiang, J. P. Hu and Z. K. Hu, "Adaptive flow balancing algorithm based on global topology awareness in SDN," *Acta Elect. Sinica*, vol. 49, no. 5, pp. 964–974, Aug. 2021. doi: 10.12263/DZXB.20190989.

[2] 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. Surv. Tutorials*, vol. 16, no. 3, pp. 1617–1634, Jan. 2014. doi: 10.1109/SURV.2014.012214.00180.

[3] S. Bhardwaj and S. N. Panda, "Performance evaluation using ryu sdn controller in software-defined networking environment," *Wireless Pers. Commun.*, vol. 122, no. 1, pp. 701–723, Aug. 2022. doi: 10.1007/s11277-021-08920-3.

[4] A. Kumar, D. Anand, S. Jha, G. P. Joshi, and W. Cho, "Optimized load balancing technique for software defined network," *Comput. Mater. Contin.*, vol. 72, no. 1, pp. 1409–1426, Mar. 2022. doi: 10.32604/cmc.2022.024970.

[5]   J. P. Duan, Z. Wang, and C. Wu, "Responsive multipath TCP in SDN-based datacenters," in *Proc. ICC*, London, England, Apr. 2015, pp. 5296–5301.

[6]   S. Zannettou, M. Sirivianos, and F. Papadopoulos, "Exploiting path diversity in datacenters using MPTCP-aware SDN," in *Proc. ISCC*, New York, NY, USA, Nov. 2016, pp. 539–546.

[7]   A. Hussein, I. H. Elhajj, A. Chehab, and A. Kayssi, "SDN for MPTCP: An enhanced architecture for large data transfers in datacenters," in *Proc. ICC*, Paris, France, May 21–25, 2017.

[8]   K. Gao, C. Q. Xu, and P. Zhang, "GCH-MV: Game enhanced compensation handover scheme for multipath TCP in 6G software defined vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 16142–16154, Dec. 2020. doi: 10.1109/TVT.2020.3042987.

[9]   T. L. Saaty, *What is the analytic hierarchy process?* Berlin Heidelberg: Springer, 1988.

[10] M. Karakus and A. Durresi, "Quality of service (QoS) in software defined networking (SDN): A survey," *J. Netw Comput. Appl.*, vol. 80, no. 2015, pp. 200–218, Feb. 2017. doi: 10.1016/j.jnca.2016.12.019.

[11] S. Chattopadhyay, S. Shailendra, S. Nandi, and S. Chakraborty, "Improving MPTCP performance by enabling sub-flow selection over a SDN supported network," in *Proc. WiMOB*, Limassol, Cyprus, Oct. 15–17, 2018.

[12] J. O. Abe, H. A. Mantar, and A. G. Yayimli, "k-maximally disjoint path routing algorithms for SDN," in *Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discovery*, New York, NY, USA, Sep. 2016, pp. 499–508.

[13] M. C. Lee and J. P. Sheu, "An efficient routing algorithm based on segment routing in software-defined networking," *Comput. Netw.*, vol. 103, no. 2, pp. 44–55, Jul. 2016. doi: 10.1016/j.comnet.2016.03.017.

[14] D. L. C. Dutra, M. Bagaa, T. Taleb, and K. Samdanis, "Ensuring end-to-end QoS based on multi-paths routing using SDN Technology," in *Proc. GLOBECOM*, New York, NY, USA, Dec. 2017.

[15] W. Froes, L. Santos, and L. N. Sampaio, "ProgLab: Programmable labels for QoS provisioning on software-defined-networks," *Comput. Commun.*, vol. 161, no. 1, pp. 99–108, Sep. 2020. doi: 10.1016/j.comcom.2020.07.026.

[16] D. E. Henni, A. Ghomari, and Y. Hadjadj-Aoul, "A consistent QoS routing strategy for video streaming services in SDN networks," *Int. J. Commun. Syst.*, vol. 33, no. 10, pp. 1–15, Oct. 2020. doi: 10.1002/dac.4177.

[17] M. R. Parsaei, H. R. Boveiri, R. Javidan, and R. Khayami, "Telesurgery QoS improvement over SDN based on a Type-2 fuzzy system and enhanced cuckoo optimization algorithm," *Int. J. Commun. Syst.*, vol. 34, no. 11, pp. e4426, Jul. 2020. doi: 10.1002/dac.4426.

[18] S. Kamath, A. Srivastava, and P. Kamath, "Application aware multiple constraint optimal paths for transport network using SDN," *IEEE Trans. Netw. Serv. Manage.*, vol. 18, no. 4, pp. 4376–4390, Dec. 2021. doi: 10.1109/TNSM.2021.3103600.

[19] A. A. Z. Ibrahim, F. Hashim, A. Sali, N. K. Noordin, and S. M. E. Fadul, "A multi-objective routing mechanism for energy management optimization in SDN multi-control architecture," *IEEE Access*, vol. 10, pp. 20312–20327, Mar. 2022. doi: 10.1109/ACCESS.2022.3149795.

[20] A. BinSahaq, T. Sheltami, A. Mahmoud, and N. Nasser, "Fast and efficient algorithm for delay-sensitive QoS provisioning in SDN networks," *Wirel. Netw.*, vol. 6, no. 10, pp. 1–22, Aug. 2022. doi: 10.1007/s11276-022-03028-3.

[21] P. Kamboj, S. Pal, and S. Bera, "QoS-aware multipath routing in software-defined networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 2, pp. 723–732, Apr. 2022. doi: 10.1109/TNSE.2022.3219417.

[22] P. Z. Gou, L. Ma, B. Y. Guo, and C. Yuan, "A multi-constrained QoS dual-path routing optimization algorithm based on software defined network," *Comput. Eng. Sci.*, vol. 45, no. 1, pp. 46–56, Jun. 2023. doi: 10.3969/j.issn.1007-130X.2023.01.006.

[23] Z. R. Zhang, J. Y. Wang, Q. Qi, H. F. Sun, and J. X. Liao, "Graph-aware deep learning based intelligent routing strategy," in *Proc. LCN*, Chicago, IL, USA, Oct. 01–04, 2018.

[24] T. M. Modi and P. Swain, "Intelligent routing using convolutional neural network in software-defined data center network," *J. Supercomput.*, vol. 78, no. 11, pp. 13373–13392, Mar. 2022. doi: 10.1007/s11227-022-04348-z.

[25] R. J. Ding, Y. D. Xu, F. F. Gao, X. M. Shen, and W. Wu, "Deep reinforcement learning for router selection in network with heavy traffic," *IEEE Access*, vol. 7, pp. 37109–37120, 2019. doi: 10.1109/AC-CESS.2019.2904539.

[26] Y. X. Hu, Z. Y. Li, J. L. Lan, J. X. Xu, and L. Yao, "EARS: Intelligence-driven experiential network architecture for automatic routing in software-defined networking," *China Commun.*, vol. 17, no. 2, pp. 149–162, Feb. 2020. doi: 10.23919/JCC.2020.02.013.

[27] X. Huang, M. Zeng, and K. Xie, "Intelligent traffic control for QoS optimization in hybrid SDNs," *Comput. Netw.*, vol. 189, no. 2, pp. 107877–107887, Feb. 2021. doi: 10.1016/j.comnet.2021.107877.

[28] W. F. Sun, Z. Wang, and G. H. Zhang, "A QoS-guaranteed intelligent routing mechanism in software-defined-networks," *Comput. Netw.*, vol. 185, no. 4, pp. 107709, Feb. 2021. doi: 10.1016/j.comnet.2020.107709.

[29] S. Knight, X. N. Hung, N. Falkner, R. Bowden, and M. Roughan, "The Internet topology zoo," *IEEE J. Select. Areas Commun.*, vol. 29, no. 9, pp. 1765–1775, Oct. 2011. doi: 10.1109/JSAC.2011.111002.