An adaptive type-2 fuzzy traffic engineering method for video surveillance systems over software defined networks

Reza Mohammadi[1] · Reza Javidan[1]

Abstract Software Defined Network (SDN) is a new network technology which allows network providers to afford predefined Quality of Service (QoS) for video streaming applications. Network administrators can develop desired traffic engineering techniques over SDN and support Quality of Experience (QoE) and QoS for their customers. One of the most important issues in traffic engineering is to find favorable links for routing between source and destination. The fitness of each link in the network depends on the end users QoE and the applications they are used. In this paper to achieve optimal routes, the fitness of each link is determined by type-2 fuzzy sets. Then, an adaptive traffic engineering method is proposed to find the best routes between source cameras and monitoring center in a video surveillance system. The proposed method is based on Constraint Shortest Path (CSP) problem and calculates minimum cost path which satisfies delay constraint. Due to NP-completeness of the CSP problems, LARAC algorithm is used to solve it. To the best of our knowledge, this is the first proposed traffic engineering technique which is based on type-2 fuzzy set for video streaming applications over SDN. The contribution of the proposed method regarding to the related works, is to apply type-2 and type-1 fuzzy logic for calculating the costs of network links based on QoE for providing QoS in a video surveillance system. In addition, this paper models the provisioning of QoS in a real scenario and emulates them in a network emulator. Many comparisons carried out between the proposed method and other well-known methods to show the effectiveness of the proposed method in terms of packet loss, delay and PSNR.

Keywords Software defined networks · Type-2 fuzzy · Traffic engineering · Video streaming · Quality of service

✉ Reza Javidan
javidan@sutech.ac.ir

Reza Mohammadi
r.mohammadi@sutech.ac.ir

[1] Department of Computer Engineering and Information Technology, Shiraz University of Technology, Shiraz, Iran

1 Introduction

In the last few years, Software Defined Network (SDN) is one of the most attractive architecture for enhancing network efficiency. Using SDN, network administrators can program the network dynamically and control the behavior of the network by separating control plane from the forwarding plane [15, 21]. The control plane globally controls the network via network administration in centralized manner [1]. The forwarding plane is responsible for packet forwarding functions and follows the policies of the control plane. The forwarding plane consists of multi-layer programmable switches which can be programmed dynamically. The communication between control plane and forwarding plane is made using southbound APIs such as OpenFlow [27]. The OpenFlow protocol facilitates software programs executing on OpenFlow switches to forward packets among the network switches [1]. Furthermore, various applications exist in the application layer to develop network services such as traffic engineering, security, network resource management and other network controlling techniques. In SDN architecture, when a switch receives a packet, it extracts the header of the packet and explores its relevant rule from forwarding rule table. If the header of the packet matched with any rule entry, the actions associated with the specific entry are performed. If the header does not match any rule entry, the switch forwards the header to the controller by using southbound protocols such as OpenFlow. Then the controller executes traffic engineering or routing algorithm and delivers a new forwarding entry to the relevant switches along the path between source and destination [21, 31]. All consequent packets with the same matching headers will then be forwarded according the new forwarding entry. The above mentioned schema is named *reactively*. The network providers can also program the control plane *proactively*. In proactive schema, the controller delivers forwarding entries to the relevant switches before the switches request any forwarding entry [6].

Usually provisioning Quality of Service (QoS) and Quality of Experience (QoE) for real time applications such as video streaming is one of the main important networking issues [27]. For example, in a video surveillance system, high packet loss rate and delay reduce the quality of received video data. In such systems, the network administrator can develop a suitable traffic engineering to provide QoS and reduce packet loss rate and delay time. An efficient traffic engineering technique calculates optimal path between source and destination. In a video surveillance system an optimal path is a path which is built by finding favorable links between source cameras and monitoring center. Moreover, the end to end delay of video data is very important and should not be more than 200 ms between two end points in video streaming applications [9]. In a video surveillance system, a favorable link is a link which has enough available bandwidth and short delay time. Therefore, the cost of each link should be determined based on these parameters. Since each path between two hosts consists of a set of links, choosing proper cost for links can improve path selection between hosts.

Due to dependency of each link to the end user perception (QoE) as well as network resources status, the overall concept is uncertain and therefore it can be modeled by fuzzy logic to assign a proper cost for each link regarding its available bandwidth and delay. Usually, in a computer network, there are many senders and receivers with different traffic patterns. Therefore, the available bandwidth and delay of each link may change dynamically. This dynamic nature of users' traffics in networks makes it difficult to assign a proper cost to links and this will increase uncertainty. In such situations, due to high level of uncertainty, using fuzzy logic is a suitable choice. Although type-1 fuzzy system can also be utilized to compute cost for each link, type-2 fuzzy logic allows handling more uncertainty and achieving better

results [19]. It has been shown [2, 10] that type-2 fuzzy systems are more viable than type-1 fuzzy systems in scenarios with high level of uncertainty. The main difference between type-2 fuzzy and type-1 fuzzy is associated with the nature of the membership functions. Implementing intelligent algorithm based on fuzzy logic for routing in computer networks is very popular [25].

Type-2 fuzzy logic system has been widely used in finance, communication, control, and medicine [4]. Type-2 fuzzy logic introduced by Zadeh [37] and involves procedures of fuzzification, inference, and output processing. Figure 1 illustrates the structure of type-2 fuzzy logic system [19]. In type-2 fuzzy logic system, the fuzzifier maps crisp inputs into type-2 fuzzy sets. The role of inference engine is to combine "IF-THEN" rules and give a mapping from input type-2 fuzzy sets to the output type-2 fuzzy sets. Type reduction procedure takes type-2 fuzzy set and produces a type-1 fuzzy set. Finally, defuzzifier maps type-1 fuzzy sets to crisp output.

This paper uses type-2 fuzzy system for assigning proper cost regarding delay and available bandwidth to each link of network based on a well-known QoE metric called Mean Opinion Score (MOS) [29]. After assigning costs to links, the proposed traffic engineering method solves the Constraint Shortest Path (CSP) problem of the network and calculates minimum cost path taking into account network constraints in terms of end to end delay for guaranteeing QoS in video surveillance system. Due to NP-completeness of the CSP problem [33], this paper uses LARAC algorithm [16] to solve the CSP problem. Assigning proper costs to the network links causes the links with high available bandwidth and low delay to be chosen for creating optimal path. As a result, the video data packets transfer via optimal path between two end points. As the matter of fact, streaming video data via optimal path reduces packet loss and delay while improves the quality of received video in monitoring center. This paper implements type-2 fuzzy system in traffic engineering method for guaranteeing QoS and supporting QoE in a video surveillance system, and compares the proposed method with type-1 fuzzy system traffic engineering and traditional techniques. Simulation results show that using type-2 fuzzy system for determining links costs gives better performance than type-1 fuzzy system as well as traditional methods.

The remainder of this paper is structured as follows. In Section 2, the related works of multimedia traffic engineering over SDN are reviewed. Section 3 presents the architecture of the proposed traffic engineering method. Formulating and solving the CSP problem using LARAC algorithm is described in Section 4. In Section 5 simulation details and test results are outlined. Finally, conclusions are described in Section 6.
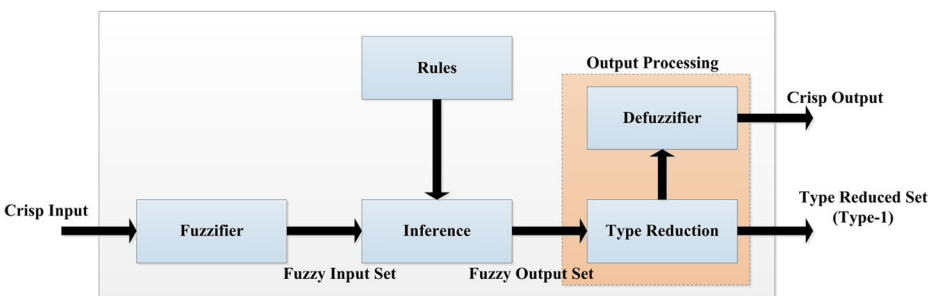

Fig. 1 Type-2 fuzzy logic system

2 Related works

In this section, some related works on multimedia and video streaming traffic engineering over SDN are surveyed. As mentioned in previous section, SDN is a new paradigm and as a result, there are few works on multimedia traffic engineering over SDN in the literatures.

Kassler et al. [20] proposed a new system for path assignment based on QoE. Their proposed system has been implemented using OpenFlow network where path cost computation and assignment is performed by conferring between end-users multimedia applications. Before transmitting streaming multimedia traffic, a session between endpoints applications is initialized based on user preferences, service requirements and terminal features. Then multimedia packets are streamed over different routes taking into account network resources via minimum cost path where costs of links are negative QoE values. In their proposed method, QoE is determined as a weighted combination of different parameters such as individual usage context, user preferences, and service content.

An optimization framework over OpenFlow networks for video streaming applications presented by Egilmez et al. [9]. This framework is based on CSP problem and network traffics are classified into three different levels where the base layers of video have been treated as a level-1 QoS flow, and the enhancement layers of video have been treated as level-2 QoS or best-effort. The authors have implemented the LARAC algorithm to solve CSP problem for computation paths. The result of the LARAC algorithm is a path with minimum cost which satisfies end to end delay constraint. In their framework, the cost of link is calculated as a weighted combination of packet loss and delay of links. Simulation results showed that their framework improves the performance of network [9]. As another work, Egilmez et al. [7] introduced a new multimedia streaming technique for large scale and multitenant SDNs. In their proposed approach, the network is divided into multi sub domains where each controller in a sub domain uses optimal QoS routing within its sub domain. Each controller shares QoS routing information with other sub domain controllers. Their approach improves received video quality, memory overhead and communication cost.

Tomovic et al. [32] presented a new control framework for QoS provisioning over SDN. In their proposed framework, controller collects network resources statistics according to the collected information. Authors have considered Dijkstra algorithm for calculating shortest path between source and destination. The cost of each link is the utilization of link. Simulation results demonstrated that their approach improves the performance under high traffic.

Another QoS routing method for video streaming over SDN has been proposed by Yu et al. [36] called ARVS. The routing uses the base and enhancement layers of video as two different levels of QoS. In their approach, delay variation has been considered as a constraint in CSP problem and the cost of each link was the weighted sum of the packet loss rate and the delay variation on a path. If the CSP problem does not satisfy the delay variation constraint, the enhancement layers of video stay on shortest path and the base layers of video reroute to another feasible path.

Egilmez et al. [8] proposed a novel approach called OpenQos for streaming video over OpenFlow networks. Their approach has a dynamic QoS routing for guaranteeing end-to-end delay QoS. Dynamic QoS routing calculates minimum cost path where the cost of each link in network is calculated as a weighted combination of link utilization and delay of link. OpenQoS reduces the packet loss and latency and can guarantee seamless video data delivery in both UDP and TCP connections.

Karl et al. [17] introduced a streaming setup which uses the dynamic routing over SDN. The streaming setup improves the quality of transmission channel for RTP/RTSP streaming applications without interaction between source and destination. In their dynamic routing packet loss rate, round-trip time (RTT) and number of hops have been considered as a single routing criterion.

The related works which have been mentioned in this section all have been proposed to provide the QoS parameters for different network applications. However, none of them has been simulated in a real specific application. Moreover, in most of the above related works, the costs of the links have been determined either as a single metric such as link utilization, packet loss and delay; or a weighted combination of multiple metrics. The main contribution of the proposed method in this paper in comparison to the related works is to apply type-2 and type-1 fuzzy logic for calculating the costs of network links based on QoE for providing QoS in a video surveillance system. In addition, this paper models the provisioning of QoS in a real scenario and emulates the model in a network emulator.

3 Architecture of the proposed method

A video surveillance system over SDN consists of multi IP cameras, OpenFlow switches, a monitoring center and a controller. The goal of developing such system is to observe and monitor an indoor or outdoor area. IP cameras capture the environment video data and then send the video to the monitoring center via network. The role of the controller in such system is to find the best routes between IP cameras and monitoring center. After that, the controller should announce the best routes to OpenFlow switches for routing video data. As mentioned before, in a video surveillance system the quality of received video in monitoring center is very vital for end user. Packet loss and delay are two important metrics in video streaming applications and affect the quality of video data. Packet loss is generally caused by congestion and occurred when a link does not have enough capacity for forwarding packets. One of the solutions for reducing packet loss is to route packets via links which have enough available bandwidth. In a video surveillance system, it is better to combine link available bandwidth and delay as a cost of link. Therefore, the controller finds the links with high available bandwidth and low delay for creating a route between an IP camera and monitoring center.

As mentioned in Section 2, most of literatures have been considered these parameters either as a single metric or as a weighted combination for cost of a link. This paper is the first paper which implements type-2 fuzzy system for assigning a proper cost to each link based on QoE. To make better comparison, this paper also implements type-1 fuzzy system. In this paper, an interval type-2 Mamdani fuzzy logic system (T2FLS) is established. The input variables of the system include link delay and link available bandwidth; and the output variable is the cost of the link. In order to simplify the calculation, singleton fuzzifier is used to fuzzify crisp input values. The domains of input variables are taken to be interval between 0 to maximum value. To reduce the computation complexity, both the two inputs are divided into three levels, that is, the available bandwidth: *low, medium, high*; the link delay: *low, medium, high*. The output value (Link Cost) is divided based on a well-known QoE metric called Mean Opinion Score (MOS) [27] into five levels: *excellent, good, fair, poor, bad*. Figures 2, 3 and 4 show the inputs and output variable and their membership functions respectively.

As mentioned so far, in a computer network with different traffic patterns, the available bandwidth and delay of each link have a dynamic nature which makes difficult to assign a
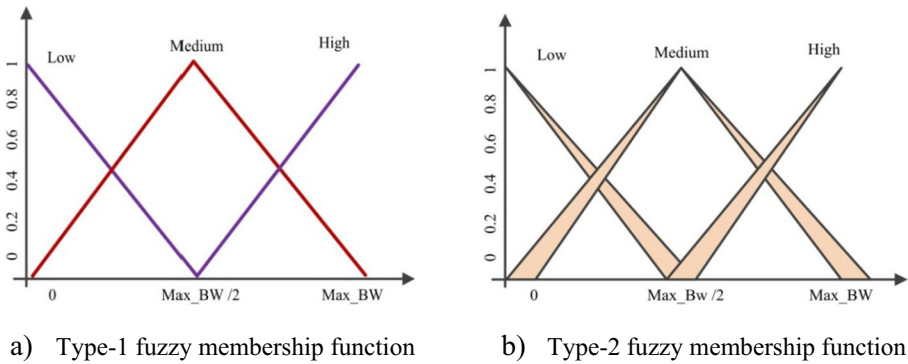
a)　Type-1 fuzzy membership function　　　b)　Type-2 fuzzy membership function

Fig. 2 Membership functions of input variable 'Available Bandwidth'

crisp cost to links and have inherited uncertainty. Due to existence of such uncertainty, type-2 fuzzy logic which allows handling more uncertainty than type-1 fuzzy can give better results [19]. As a result, in this paper for improving the performance of fuzzy system and achieving better results [3], type-2 fuzzy system as extension of type-1 fuzzy system is used by adding uncertainties and spreading the membership functions' values by ±1. The antecedents of fuzzy "IF-THEN" rules is represented by 3*3 matrix with the column representing "low", "medium" and "high" delay of link, and the rows representing "low", "medium" and "high" available bandwidth of link as shown in Fig. 5.

In this paper the Center Of Sets (COS) type-reducer [23] which combines the process of type reduction and inferencing is used in which each type-2 set is replaced by its centroid and a weighted average of centroids is computed. Finally, fuzzification process is performed by using average of the results of type reducer procedure. After assigning costs to the links of the network, the controller can execute the desired algorithm to find the best path between an IP camera and monitoring center. Figure 6 illustrates the architecture of the proposed traffic engineering for video surveillance system. As demonstrated in Fig. 6, the proposed architecture is modular and new modules and services can be added. The proposed traffic engineering schema over SDN is composed of the six following modules:

Links available bandwidth collector: Frequently acquires information about available bandwidth of each link in a network and stores them in available bandwidth matrix.
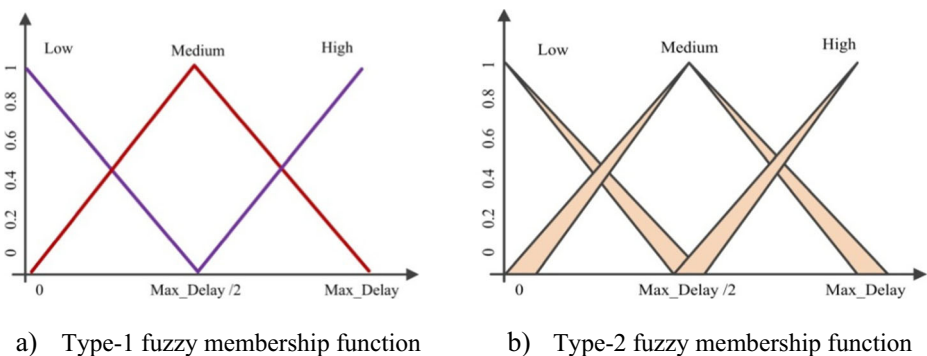

a)　Type-1 fuzzy membership function　　　b)　Type-2 fuzzy membership function

Fig. 3 Membership functions of input variable 'Link Delay'

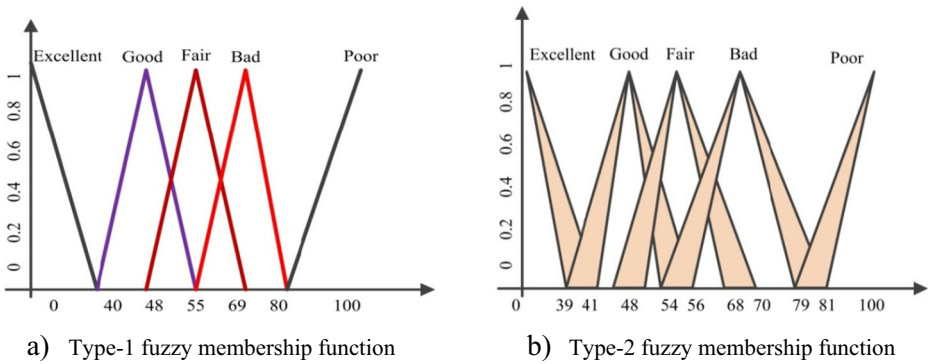a) Type-1 fuzzy membership function   b) Type-2 fuzzy membership function

Fig. 4 Membership functions of output variable 'Link Cost'

Links delay collector: Frequently acquires information regarding delay of each link of the network and stores them in delay matrix.

Link cost calculator: This module works both on available bandwidth and delay matrix and computes the costs of the links using fuzzy logic and stores the link costs in link cost matrix. This module can be replaced and reprogrammed with the desired module by network administrator for assigning link costs using another method such as weighted sum of metrics.

Network topology mapper: This module periodically collects information about network topology including nodes and links and maps them into a structure called network connectivity matrix.

Route finder: Implements the CSP model which is described in Section 4. This module uses the network connectivity matrix and link cost matrix to solve the CSP problem using LARAC algorithm and finds the optimal route regarding the costs of the links.

Route inserter: This module injects the forwarding rule entries into the relevant switches along the optimal route.

All of these six modules respectively execute periodically and cause the video data packets to be transferred via optimal path.

4 Problem formulation

In this paper, computation of an optimal path between IP cameras and monitoring center is considered as a CSP problem. The goal of CSP problem is to find the minimum cost path between two endpoints regarding the situation where the end to end delay should be equal or less than a specified value [23]. A communication network in CSP problem is considered as directed graph $G(V, E)$, where E is a set of network links and V is a set of network nodes. Each link (a, b) in the graph has two weights $c_{ab} > 0$ and $d_{ab} > 0$ which denote link cost and link delay, respectively. In addition, s and t are source and destination nodes and P_{st} denote the set of all feasible path from node s to node t. Let P be a feasible route in Pst. For each P is P_{st}, the

Fig. 5 The fuzzy rule matrix

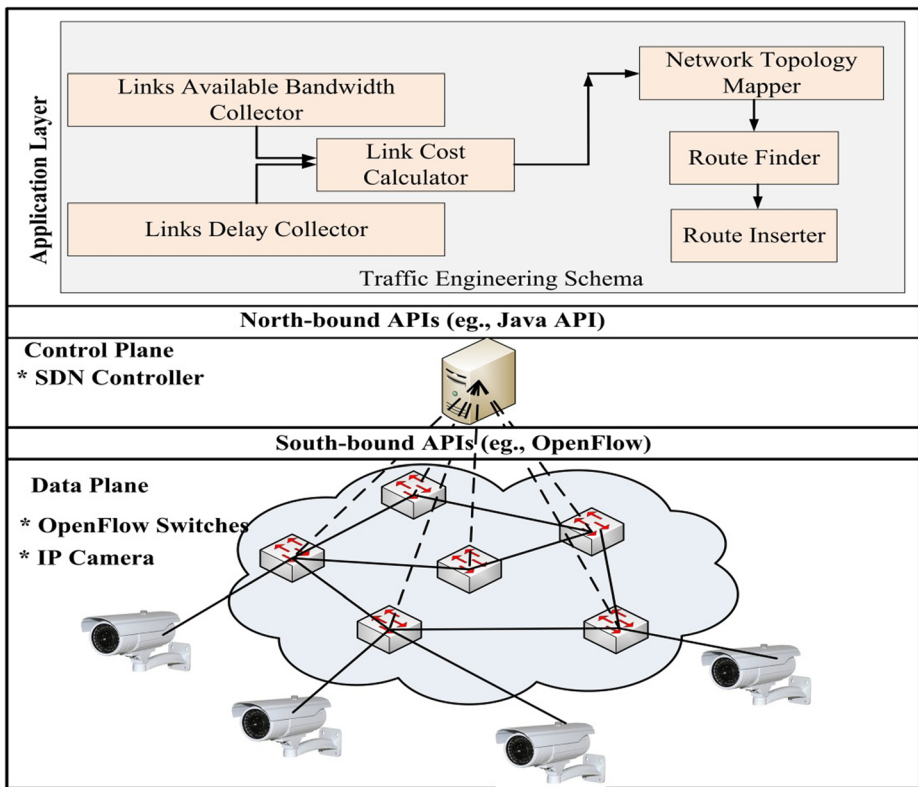|	Low	Medium	High
Low	Poor	Poor	Bad
Medium	Good	Fair	Poor
High	Excellent	Good	Poor

Fig. 6 Architecture of the proposed traffic engineering for video surveillance system

cost of path, $c(P)$ and delay of path, $d(P)$ is defined as follows:

$$c(P) = \sum_{(a,b)\in P} c_{ab}, d(P) = \sum_{(a,b)\in P} d_{ab} \qquad (1)$$

Consider $T > 0$ as a threshold for end to end delay of path P. The CSP problem can be formulated as the following integer linear program:

$$\min \sum_{(a,b)\in E} c_{ab}x_{ab}$$
$$\text{s.t.} \forall a \in V$$

$$\sum_{\{b\backslash(a,b)\in E\}} x_{ab} - \sum_{\{b\backslash(b,a)\in E\}} x_{ba} = \begin{cases} 1 \text{ if } a = s \\ -1 \text{ if } a = t \\ 0 \text{ otherwise} \end{cases} \qquad (2)$$

$$\sum_{(a,b)\in E} d_{ab} \leq T, \quad T \geq 0$$
$$x_{ab} = 0 \text{ or } 1, \quad \forall(a,b)\in E$$

Where x_{ab} and c_{ab} are decision variable and the link cost of (a, b), respectively. The objective function minimizes the path cost from node s to node t regarding to problem constraints [35]. As reviewed in previous sections, the link cost of (a, b) in the most literatures

is either the combination of multiple metrics or a single metric of QoS. In this paper the link cost is determined by using fuzzy logic.

According to [22], it has been proved that the CSP problem is NP-complete and some approximation and heuristic algorithms have been proposed in many literatures [16, 22]. The Lagrangian Relaxation Based Aggregated Cost (LARAC) algorithm is one of the most promising and polynomial-time algorithms in which the lower bound for optimal value of the CSP problem is provided. Figure 7 shows the LARAC algorithm [16]. As shown in Fig. 7, *Dijkstra(src,dest,cost)* finds the shortest path using link costs, *Dijkstra(src,dest,delay)* finds the shortest path using link delays and *Dijkstra(src,dest,c$_\lambda$)* finds the shortest path using combined link weights regarding to λ multiplier. The functions *d(r), c(r)* and *c$_\lambda$(r)* are used to calculate the delay, cost and aggregated cost of path, respectively. In the first step, the LARAC algorithm computes the shortest path (p_c) using link costs; If p_c satisfies the delay constraint (be equal or less than T threshold value), this will be the optimal solution. Otherwise, LARAC computes the shortest path using link delays denoted as p_d. If the path p_d does not meet the delay constraint (be equal or less than T threshold value), there is no solution to this instance. Otherwise, LARAC computes the optimal solution by searching for optimal λ, iteratively. It has been proved that LARAC finds a good solution in $O\left([nlogn + m]^2\right)$ time [33] where n and m are the number of nodes and links, respectively.

5 Performance evaluation

In this section, a comprehensive simulation study is investigated and the results have been analyzed to evaluate the performance of the proposed type-2 fuzzy system traffic engineering method for video surveillance system. The performance of the proposed method is evaluated against three following benchmarks:

Benchmark-1: the cost of each link in the test-bed network is computed based on type-1 fuzzy system according to Fig.2, Fig.3, Fig.4 and Fig. 5. Then the LARAC algorithm is used for calculating the minimum path cost between source IP cameras and monitoring center.

Benchmark-2: the cost of each link in the test-bed network is determined as the available bandwidth ratio of the links. In this approach the cost of link *(a, b)* is defined as Eq. (3):

$$c_{ab} = \frac{B_{ab} - G_{ab}}{B_{ab}} \tag{3}$$

Procedure *LARAC(G, src, dest , cost, delay, T)*
      $Pc=Dijkstra(src,dest,cost);$
      If $d(p_c) \leq T$ **then return** p_c.
      $P_d=Dijkstra(src,dest,delay);$
      If $d(p_d) > T$ **then return** *"there is no feasible solution"*
      Repeat
            $\lambda = \frac{c(p_c)-c(p_d)}{d(p_d)-d(p_c)}$
            $r= Dijkstar(src,dest,c_\lambda);$
            if $c_\lambda(r)=c_\lambda(p_c)$ **then return** p_d;
            else if $d(r) \leq T$ **then** $p_d=r$ **else** $p_c=r$
      end repeat
End procedure

Fig. 7 Pseudo-code of LARAC algorithm

Where B_{ab} and G_{ab} are the total bandwidth and the occupied bandwidth of link *(a, b)*, respectively. After assigning the costs of links, this approach uses the LARAC algorithm for finding the shortest path between source IP cameras and monitoring center.

Benchmark-3: OSPF routing is the one of the most popular routing protocols in IP based networks. OSPF protocol has been widely used in the literatures for comparing and evaluating the performance of new routing schemes [18, 24, 28]. The cost of a link in OSPF is inversely proportional to the bandwidth of that link. OSPF uses Dijkstra routing algorithm to find the shortest path between two desired source and destination node [5]. In this approach OSPF routing protocol is used for finding the shortest path between source IP cameras and monitoring center.

5.1 Configuration

To make a fair comparison, the proposed method and benchmarks are evaluated in two different test-bed scenarios as follow:

Scenario-1: in this scenario, the test-bed network consists of 10 OpenFlow switches, 5 IP cameras and a monitoring center. There are 24 links in the network for connecting OpenFlow switches together which have different bandwidth and delays.

Scenario-2: the test-bed network of this scenario consists of 5 IP cameras, 28 OpenFlow switches and a monitoring center. OpenFlow switches are connected together with 50 links which have different bandwidth and delays.

The bandwidth and delay of links in both scenarios are randomly chosen between 1 Mbps to 15 Mbps, and 1 ms to 30 ms, respectively. In this paper *Opendaylight* controller is used for implementing the traffic engineering technique. Opendaylight is a popular open source controller which is used for enabling SDN network to be implemented using programs [34]. This controller is installed on a machine with CPU Intel Core i7-4700MQ-2.4 GHz and 6GB RAM. In the test-bed scenarios, the proposed method and the benchmarks are programmed in Java programming language and each of them installed as a bundle on Opendaylight controller. Each different bundle is evaluated separately for each scenario. The controller executes the bundle every 20 s for calculating optimal paths between the IP cameras and the monitoring center. In both scenarios, the maximum tolerable end to end delay for video surveillance system is set to 200 ms [9]. This value is the *T* threshold value in the CSP problem which is used for modeling the network.

The both test-bed scenarios are emulated in *mininet* environment [11]. To achieve accurate simulation results, the hosts in mininet are configured according to the specification of Cisco video surveillance IP camera 6030. This camera is a high-definition video endpoint which supports multiple resolutions and video encoding and compression methods [13]. Each host in the test-bed scenarios uses VLC player [14] for streaming video to the monitoring center over SDN. VLC player is an open source multimedia player which can stream multimedia over HTTP or RTP protocol [26]. The performance of each traffic engineering method is evaluated under three different workloads. Therefore, each host using VLC player streams three different video resolutions: 1920*1080 (H.264), 1280*720 (H.264) and 640*480 (H.264) video at 30 fps over RTSP protocol for 180 s (3 min). RTSP is an application layer protocol for streaming real time data such as video and audio [30]. The video content which is streamed by each host is the *Big Buck Bunny* animation [12].

5.2 Performance metrics

The performance of the proposed method and three benchmarks is evaluated using the following QoS parameters in both scenarios:

PSNR (Peak Signal-To-Noise Ratio): PSNR is a metric which is useful to measure video quality. This metric is used to calculate the error between the original streamed video and the received video.

Average end to end delay: this metric is the average time interval from the IP camera to the monitoring center for each successfully delivered video packet.

Packet loss ratio: packet loss ratio is the total number of video packets which haven't been delivered at the monitoring center to the total number of packets streamed at the IP cameras.

5.3 Simulation results

In this section, simulation results have been analyzed for different scenarios and workloads in terms of QoS metrics. Figure 8 illustrates the average end to end delay under different workloads for the test-bed scenarios. Figure 9 illustrates the packet loss under different workload for test-bed scenarios. As shown in Fig. 8 and Fig. 9, it is clear that type-2 fuzzy system traffic engineering performs better than type-1 fuzzy system traffic engineering and other methods in all of the trials. This is because type-2 fuzzy system handles more uncertainty and in the proposed method it assigns proper link cost, and the LARAC algorithms also finds low cost links for calculating the minimum cost routes. Moreover, OSPF routing protocol (benchmark-3) performs worst in all trials. This is because the OSPF protocol only considers the total bandwidth of the link as a metric for link cost, and Dijkstra algorithm finds the shortest path without considering end to end delay constraint. Since benchmark-2 uses the available bandwidth ratio metric for assigning link cost and the link with higher available bandwidth has lower cost, it performs better results than OSPF routing.

Figure 10 illustrates the PSNR under different workloads for the test-bed scenario-1 where the horizontal axes denotes the number of video frames and the vertical axes denotes average PSNR. Figure 11 illustrates the PSNR under different workloads for
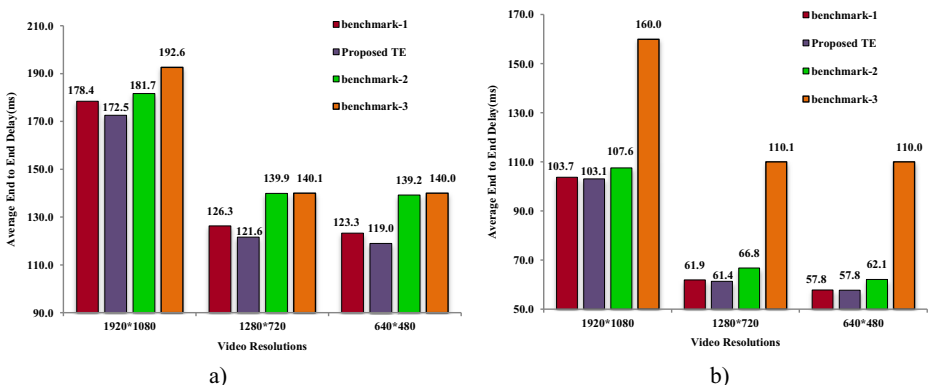

Fig. 8 Average end to end delay: **a** test-bed scenario-1, **b** test-bed scenario-2
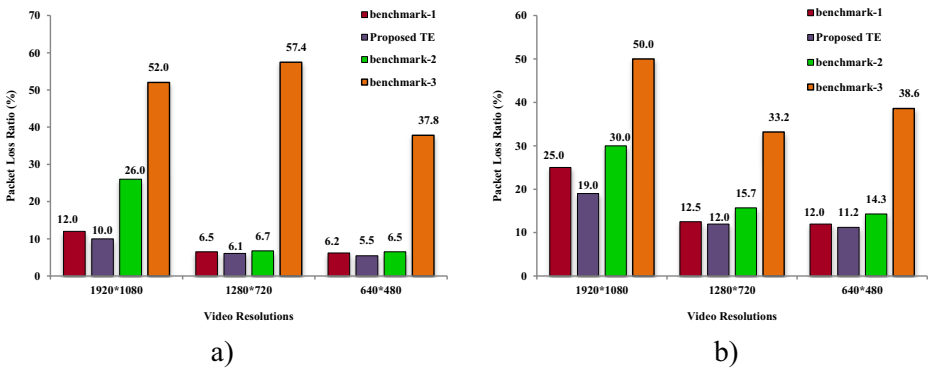
Fig. 9 Packet loss ratio: **a** test-bed scenario-1, **b** test-bed scenario-2

the test-bed scenario-2 where the horizontal axes denotes the number of video frames and the vertical axes denotes average PSNR. As depicted in Fig.10 and Fig. 11, both type-1 and type-2 fuzzy system traffic engineering show better performance than two other methods in all experiments in terms of PSNR. Because by using type-1 and type-2 fuzzy system traffic engineering, the end to end delay and packet loss have been reduced, as a result the PSNR of received video has been increased. Moreover, type-2 fuzzy system traffic engineering outperforms other methods in high workloads.
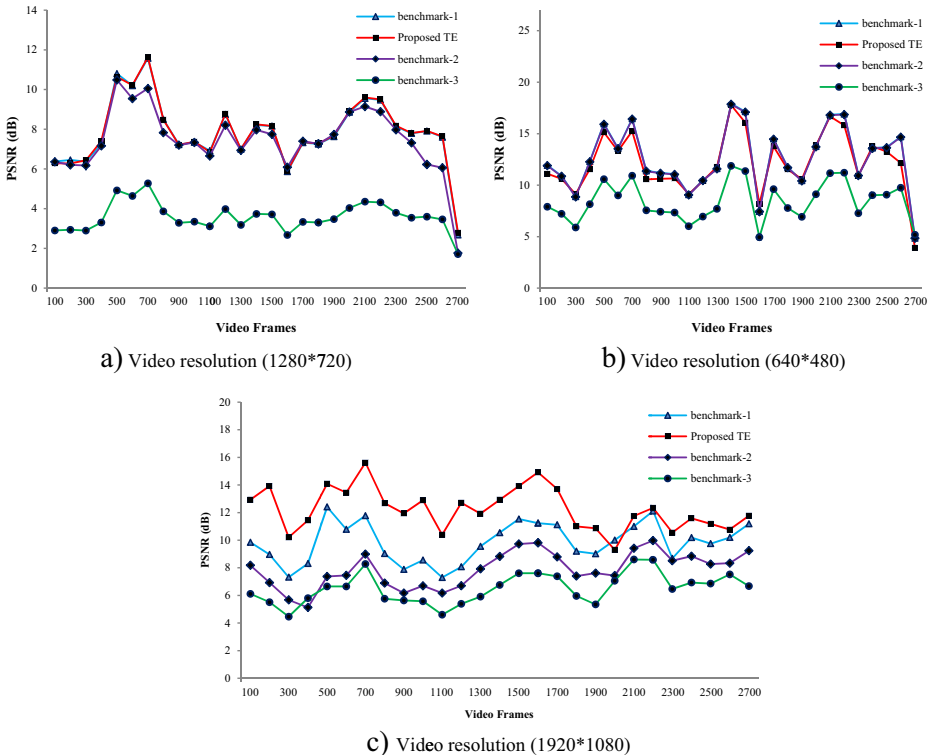

Fig. 10 PSNR values for test-bed scenario-1 under three different video resolutions
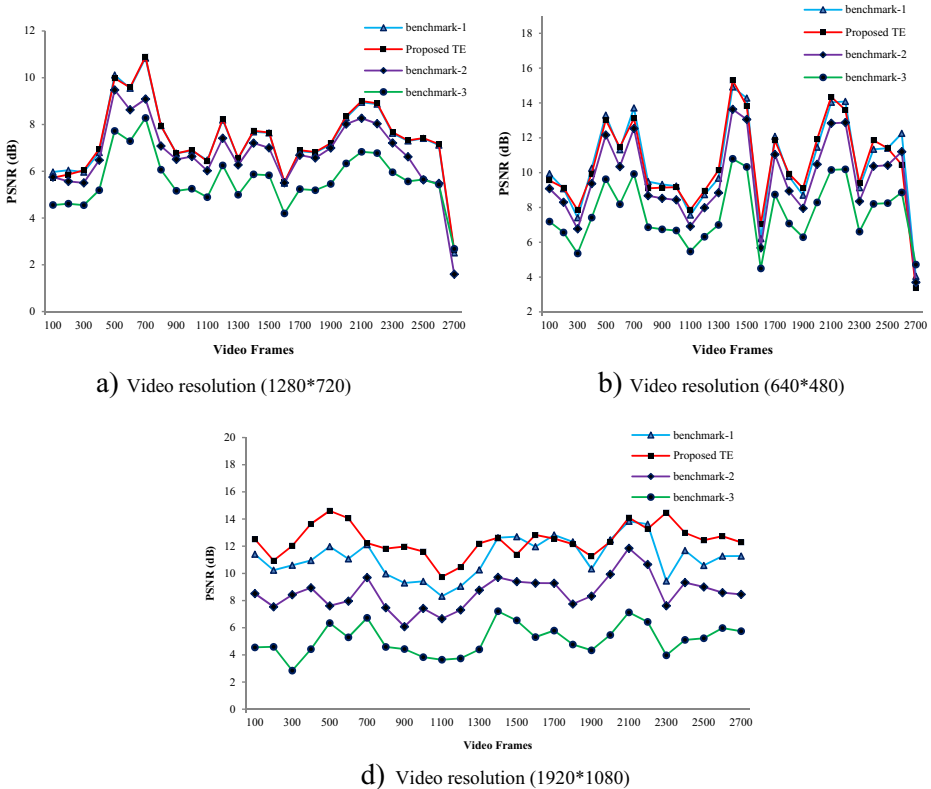
a) Video resolution (1280*720)

b) Video resolution (640*480)

d) Video resolution (1920*1080)

Fig. 11 PSNR values for test-bed scenario-2 under three different video resolutions

6 Conclusions

In this paper an adaptive traffic engineering method based on type-2 fuzzy logic for video surveillance system over SDN is proposed. Type-2 fuzzy system has been used for assigning costs to the network links based on QoE; and adaptive traffic engineering calculates the best routes to guarantee QoS in a video surveillance system. Finding the optimal routes between streamer cameras and monitoring center in video surveillance system has been modeled as a CSP problem which is solved by using LARAC algorithm. The produced forwarding rules are send to the associated OpenFlow switches for routing manipulation. The Performance of the proposed method on the proposed video surveillance system has been evaluated against the type-1 fuzzy system traffic engineering, available bandwidth based traffic engineering and OSPF routing protocol. To measure the benefits of the proposed method, different scenarios and workloads have been investigated. Because PSNR, end to end delay and packet loss ratio are the most important metrics for measuring QoS in real time video streaming applications, this paper evaluates these metrics for evaluating the proposed method. Simulation results demonstrated that the proposed traffic engineering based on type-2 fuzzy system is better than the other methods in terms of end to end delay, PSNR and packet loss ratio.

References

1. Akyildiz IF, Lee A, Wang P, Luo M, Wu C (2014) A roadmap for traffic engineering in SDN-OpenFlow networks. Comput Netw 71:1–30
2. Aladi JH, Wagner C, Garibaldi JM (2014) Type-1 or interval type-2 fuzzy logic systems—on the relationship of the amount of uncertainty and FOU size. 2014 I.E. International Conference on Fuzzy Systems (FUZZ-IEEE). IEEE
3. Baklouti N, John R, Alimi AM (2012) Interval type-2 fuzzy logic control of mobile robots. J Intell Learn Syst Appl 4(04):291
4. Castillo O, Melin P, Kacprzyk J, Pedrycz W (2007) Type-2 fuzzy logic: theory and applications. In Granular Computing, 2007. GRC 2007. IEEE international conference on, pp. 145–145. IEEE
5. Dijkstra E W (1959) A note on two problems in connexion with graphs. Numerischemathematik 1, no. 1: 269–271
6. Dusi M, Bifulco R, Gringoli F, Schneider F (2014) Reactive logic in software-defined networking: Measuring flow-table requirements. In Wireless Communications and Mobile Computing Conference (IWCMC), 2014 International, pp. 340–345. IEEE
7. Egilmez H, Tekalp M (2014) Distributed QoS architectures for multimedia streaming over software defined networks. Multimedia, IEEE Transactions on 16(6):1597–1609
8. Egilmez HE, Dane ST, Bagci KT (2012) OpenQoS: An OpenFlow controller design for multi-media delivery with end-to-end Quality of Service over Software-Defined Networks. In Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific, pp. 1–8. IEEE
9. Egilmez HE, Civanlar S, Tekalp AM (2013) An optimization framework for QoS-enabled adaptive video streaming over OpenFlow networks. Multimedia, IEEE Transactions on 15, no. 3: 710–715
10. Hagras H, Wagner C (2012) Towards the wide spread use of type-2 fuzzy logic systems in real world applications. IEEE Comput Intell Mag 7(3):14–24
11. http://mininet.org/ [Accessed 06 February 16]
12. http://www.bigbuckbunny.org/ [Accessed 17 March 16]
13. http://www.cisco.com/c/en/us/products/collateral/physical-security/video-surveillance-6000-series-ip-cameras/data_sheet_c78-714746.html [Accessed 06 February 16]
14. http://www.videolan.org/vlc/index.html [Accessed 06 February 16]
15. Jammal M, Singh T, Shami A, Asal R, Li Y (2014) Software defined networking: state of the art and research challenges. Comput Netw 72:74–98
16. Juttner A, BalazsSzviatovski, IldikóMécs, ZsoltRajkó (2001) Lagrange relaxation based meth-od for the QoS routing problem. In INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 2, pp. 859–868. IEEE
17. Karl M, Gruen J, Herfet T (2013) Multimedia optimized routing in OpenFlow networks. In Networks (ICON), 2013 19th IEEE International Conference on, pp. 1–6. IEEE
18. Karl M, Gruen J, Herfet T (2013) Multimedia optimized routing in OpenFlow networks. In *Networks (ICON), 2013 19th IEEE International Conference on*, pp. 1–6. IEEE, 2013.
19. Karnik NN, Mendel JM, Liang Q (1999) Type-2 fuzzy logic systems. Fuzzy Systems, IEEE Transactions on 7(6):643–658
20. Kassler A, et al. (2012) Towards QoE-driven multimedia service negotiation and path optimization with software defined networking. Software, Telecommunications and Computer Networks (SoftCOM), 2012 20th International Conference on. IEEE
21. Kreutz D, Ramos FMV, Verissimo PE, Rothenberg CE, Azodolmolky S, and Uhlig S (2015) Software-defined networking: A comprehensive survey. Proceedings of the IEEE 103, no. 1: 14–76
22. Liu G, and Ramakrishnan KG (2001) A* Prune: an algorithm for finding K shortest paths subject to multiple constraints. In INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 2, pp. 743–749. IEEE
23. Mendel JM. (2001) Uncertain rule-based fuzzy logic system: introduction and new directions.
24. Mirzahossein K, Nguyen M, and Elmasry S. (2013) Analysis of RIP, OSPF, and EIGRP Routing Protocols using OPNET. Simon Fraser University, School of Engineering Final Year Project, ENCS 427: Communication Networks

25. Mohammadi R, Javidan R, Jalili A (2015) Fuzzy Depth Based Routing Protocol for Underwater Acoustic Wireless Sensor Networks. Journal of Telecommunication, Electronic and Computer Engineering (JTEC) 7, no. 1: 81–86
26. Müller C, Timmerer C (2011) A VLC media player plugin enabling dynamic adaptive streaming over HTTP. In Proceedings of the 19th ACM international conference on Multimedia, pp. 723–726. ACM
27. Ongaro F, Cerqueira E, Foschini L, Corradi A, Gerla M (2015) Enhancing the quality level support for real-time multimedia applications in software-defined networks. In Computing, Networking and Communications (ICNC), 2015 International Conference on, pp. 505–509. IEEE
28. Rakheja P et al (2012) Performance analysis of RIP, OSPF, IGRP and EIGRP routing protocols in a network. International Journal of Computer Applications 48(18):6–11
29. Rec I. (1996) T. U. T. P. 800: methods for subjective determination of transmission quality. international telecommunication union, Geneva
30. Schulzrinne H. (1998) Real time streaming protocol (RTSP)
31. Sezer S, Scott-Hayward S, Pushpinder-KaurChouhan, Fraser B, Lake D, Finnegan J, NielViljoen, Miller M, and NavneetRao. (2013) Are we ready for SDN? Implementation challenges for software-defined networks. Communications Magazine, IEEE 51, no. 7: 36–43
32. Tomovic S, Prasad N, Radusinovic I. (2014) SDN control framework for QoS provisioning. In Telecommunications Forum Telfor (TELFOR), 2014 22nd, pp. 111–114. IEEE
33. Wang Z, and Crowcroft J (1996) Quality-of-service routing for supporting multimedia applications. Selected Areas in Communications, IEEE Journal on14, no. 7: 1228–1234
34. www.opendaylight.org [Accessed 06 February 16]
35. Xiao Y, KrishnaiyanThulasiraman, GuoliangXue, AlpárJüttner (2005) The constrained shortest path problem: algorithmic approaches and an algebraic study with generalization. AKCE International Journal of Graphs and Combinatorics 2, no. 2: 63–86
36. Yu T-F, Wang K, Hsu Y-H (2015) Adaptive routing for video streaming with QoS support over SDN networks. In Information Networking (ICOIN), 2015 International Conference on, pp. 318–323. IEEE
37. Zadeh LA (1975) The concept of a linguistic variable and its application to approximate reasoning—I. Information sciences 8, no. 3: 199–249

Reza Mohammadi was born in 1986. He is graduated from MSc degree in Computer Networks from Shiraz University of Technology in 2013. Now he is a PhD student on Computer Networks in Shiraz University of Technology. Mr. Mohammadi has many publications in international conferences and journals regarding Underwater Wireless Sensor Networks (UWSNs). Currently he focused on Software Defined Networks (SDN) as a new trend in computer networks.

His major fields of interest are Software Defined Networks (SDNs), heuristic algorithms, Underwater Wireless Sensor Networks (UWSNs), Ad hoc Networks and Underground Wireless Sensor Networks (UWSNs).

Reza Javidan born in 1970. He is graduated from MSc Degree in Computer Engineering (Machine Intelligence and Robotics) from Shiraz University in 1996. He received PhD degree in Computer Engineering (Artificial Intelligence) from Shiraz University in 2007. Dr. Javidan has many publications in international conferences and journals regarding Image Processing, Underwater Wireless Sensor Networks (UWSNs) and Software Defined Networks (SDNs). He has many PhD and MSc students working together as a team.

His major fields of interest are artificial intelligence, image processing, Underwater Wireless Sensor Networks (UWSNs), Software Defined Networks (SDNs), and SONAR systems. Dr. Javidan is now member of faculty and lecturer in Department of Computer Engineering and Information Technology in Shiraz University of Technology.