Journal of Engineering and Applied Sciences 11 (4): 722-732, 2016
ISSN: 1816-949X

Optimization of Throughput and Efficiency in Software-Defined Data Center

[1]Azadeh Darvazeh Ban and [2]Seyed Javad Mirabedini
[1]Department of Computer of Software Engineering, Islamic Azad University, Damavand, Iran
[2]Department of Computer Engineering, Islamic Azad University, Central Tehran Branch, Tehran, Iran

Abstract: Without doubt, the speed and efficiency of any system is one of the most important parameters of its success in supply of services; Network structure and its related settings have a direct impact on the speed and efficiency of network-based services as a result optimization of efficiency, power and energy is of great importance in infrastructure of networks. With the increasing volume of data, workload and traffic in data center networks, the networks require more dynamism to manage data centers with lower costs. However, the virtualization technologies in the cloud data centers improve energy consumption by reducing the number of physical devices and improve efficiency of data center so much, optimization of throughput and efficiency in software-defined data center is still the major concern for network administrators. Software defined networking and more specifically open flow protocol are ready to disrupt the current status and provide new ways to manage traffic and optimize network performance in data centers. SDN is a new network architecture that provides more flexible routing and better traffic control on different layers of network. An approach was proposed in the method proposed in this study with the capability for software planning of open flow controllers in controlling and managing software-defined data center, characterizing input traffics, controlling any traffic regarding its nature, optimal allocation of resources using token bucket algorithm, considering parameters such as quality of service and quality of experience to increase user's satisfaction which result in improvement of throughput and efficiency in software-defined data center.

Key words: Open flow, SDN, QoE, QoS, token bucket

INTRODUCTION

Speed and efficiency of any system are most important success parameters in supply of services; network structure and its related settings have a direct impact on the speed and efficiency of network-based services; a network without favorable infrastructure and with low throughput, packet loss, retransmission and latency raises a severe loss in efficiency and speed of software-defined data center. As a result, optimization of throughput and efficiency is of great importance in software-defined data center. Data centers are experiencing a fundamental change. Networks have become faster and virtualization has become a natural thing. However, the virtualization technologies in the cloud data centers (Wette *et al*., 2014) have improved energy consumption by reducing the number of physical devices and improved efficiency of data center so much, traditional software-defined data centers fail to meet all the growing needs of large organizations and end users, whereby optimization of throughput in software-defined data center has remained a major concern for managers. Software-based networks and more specifically open flow protocol provide novel methods for traffic control and management, more flexible routing and network performance optimization in large virtual data centers by separating control layers from transport layer. Mehrian *et al*. (2016), Nowruzpour Mehrian *et al*. (2013) and Vaziri *et al*. (2015) present a novel algorithm in optimizing structures which overcame the previous difficulties. Data centers as connectivity oriented are becoming service-oriented networks. This new paradigm brings about change in the framework and architecture of networks and their design and structure. In the new paradigm, instead of bedding switches and routers, it is tried to reduce complexity of its components and develop new capacities by homogenization, simple configurations and creation of programmability capabilities in the data center as much as possible and create new capacities. There are several programs in the context of network that are competing for fixed bandwidth on the basis of best effort; ultimately we will face network congestion on one

Corresponding Author: Azadeh Darvazeh Ban, Department of Computer of Software Engineering, Islamic Azad University, Damavand, Iran

hand and reduction of network efficiency and poor QOE on the other hand. Users tend to improve quality of specific programs yet the current network service controllers lack suitable mechanisms to satisfy the users in this context (Li *et al*., 2014). An approach was proposed in the method proposed in this study with the capability for software planning of open flow controllers in controlling and managing software-defined data center, characterizing input traffics, controlling any traffic regarding its nature, optimal allocation of resources using Token Bucket algorithm, considering parameters such as quality of service and quality of experience to increase user's satisfaction (Govindarajan *et al*., 2014) which result in improvement of throughput and efficiency in software-defined data center (Salvadori *et al*., 2011).

The application of Data Envelopment Analysis (DEA) method in optimizing the efficiency of the system has been investigated by Tavana *et al*. (2016), Li *et al*. (2016) and Mobin *et al*. (2015). For example as it is suggested by Tavana *et al*. (2016), Li *et al*. (2016) and Mobin *et al*. (2015) after defining the inputs and outputs of the system, the DEA method can be applied to find the optimal setting of the system. Detail description of the data envelopment analysis in provided by Tavana *et al*. (2016), Li *et al*. (2016) and Mobin *et al*. (2015). In the present research, the literature review of the proposed method has been mentioned in the first section and then the proposed method for optimization of throughput and efficiency in software-defined data center has been proposed and ultimately the results from simulation of the proposed method and its comparison with previous methods have been proposed. To sum up, the conclusion and related researchs are proposed.

Literature review: Previous methods in the context of optimization of throughput and efficiency in software-defined data center provide general solutions consisting of migration of virtual machines (Lin *et al*., 2013) in which less priority is given to SDN controller capabilities in network control and management which is considered as the strength of software-defined networks, thus we provide a new approach to manage and control the network as a software to achieve user's satisfaction and improvement in efficiency and throughput of network. The related researchs in the context of software-defined networks can be categorized in several categories:

- Hardware X box design (switches, routers, firewalls, etc.) based on idea of separation of control path and data path
- Emphasis on general applications of SDN in optimization of the mechanisms relating to data centers
- Overview of the virtualization instruments and comparison of the simulators and controllers based on open flow (Bayestehtashk *et al*., 2016; Cziva *et al*., 2014; De Oliveira *et al*., 2014)
- Emphasis on existing challenges in transition from traditional networks to SDN networks

MATERIALS AND METHODS

In this study, firstly the basic concepts of research have been examined in concise and then the proposed method for optimization of throughput and efficiency in software-defined data center has been mentioned based on these concepts. Basic concepts of research. In this study, basic and essential concepts used in the proposed method are mentioned.

Advantage of SDN networks in the proposed method: In new architecture of the modern data centers, instead of bedding switches and routers, an attempt is made to reduce complexity in the data centers by homogenization, simple configurations and planning capabilities of these centers and raise new capacities. In the proposed method, we used the software capability of SDN controller in performance of open flow switches in a software-defined data center to manage traffic levels, improve user's satisfaction, allocate resources in an optimal way and improve throughput and efficiency of network.

How open flow works: Layered Architecture for SDN is in this way that the network is controlled in a centralized way in SDN controller; open flow protocol refers to one of the most important protocols that is used as SDN controller for planning and programing behavior of switches shown in Fig. 1.

In the approach proposed in this section, function of network is planned and managed in a way to improve throughput and efficiency of network using open flow protocol, open flow switches and open flow controller. For deep understanding of the proposed method, how open flow switch works and how to use capability of management of open flow controller in control of network function have been elaborated.

How open flow switch works: Any open flow switch includes several flow tables and any flow table includes several flow entries. The open flow pipeline processing defines how packets interact with flow tables. When a packet reaches to this switch, header of this packet is compared with existing flow entry in flow table. If the information of this header is matched with values of flow

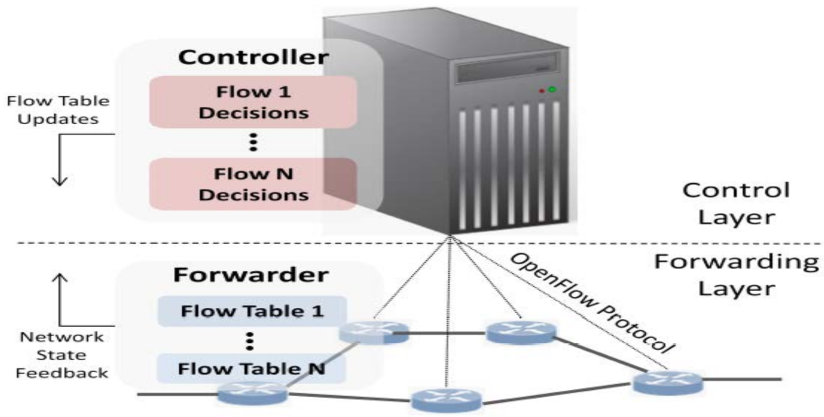

Fig. 1: Open flow architecture

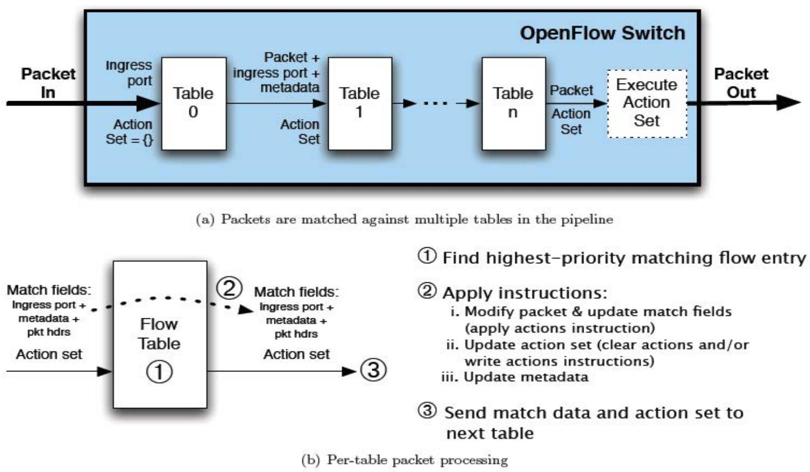

Fig. 2: Flow of packets through pipeline processing

entry, this will imply that how open flow controller has declared exposure with this packet to the switches that the tasks below can be mentioned:

- Send the packet to the output port
- Drop packet
- Send packet to the next flow table
- Send packet to controller

Further, open flow controller has this ability to declare a task to each switch at any state and time and update its flow table:

- Ingress port so
- Metadata
- Ethet src
- Ether dest
- Ether type
- VLAN id
- VLAN priority
- MPLS label
- MPLS traffic class
- IPV4 src
- IPV4 dst
- IPV4 proto/ARP opcode
- IPV4 ToS bits
- TCP/UDP/SCTP src port ICMP type
- TCP/UDP/SCTP dst port ICMP code

An open flow switch must have at least one flow table. Flow tables are numbered in turn that numbering starts from 0; the packet is firstly compared with flow entries in table 0 and other flow tables might be used regarding the output from comparison in the first table shown in Fig. 2 and 3.

Some of the management tasks of controller: Some of the management tasks of controller include topology management, path management, flow management, path calculation, traffic policy making (Egilmez *et al*., 2012).

Characterization of data flows: What causes improvement in throughput and efficiency of network is access to maximum bandwidth, minimum delay and

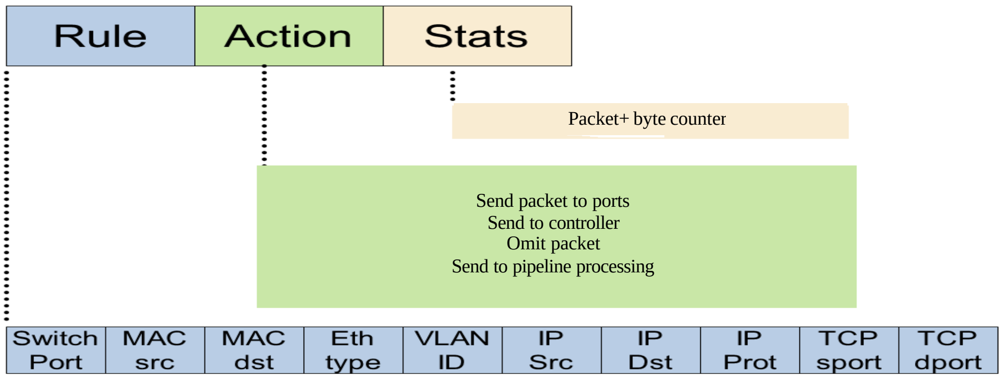

Fig. 3: Inputs of open flow table

Table 1: The requirements for quality of service for different applications

Application	Bandwidth	Delay	Jitter	Loss of packets
VIOP	Low	High	High	High
Video conference	High	High	High	Average
VoD	High	Average	Average	Average
Streaming audio	Low	Average	Average	Average
Email	Low	Low	Low	High
File transmission	Average	Low	Low	High

minimum loss of packet with the least cost. In the proposed approach, certain traffic levels are raised to optimize throughput and efficiency of network, improve quality of service in network and reduce complexity rate of network so as to provide service for each traffic regarding its nature. The common parameters of quality of service which are used to define efficiency of network include shown in Table 1. Bandwidth (throughput): the volume of traffic in bits that is transmitted from a point of the network within a specified time (in sec) or a specified period through the network. Delay: the amount of time which takes for a data packet to reach from the source to destination. Jitter (delay variance). Delay variance that a data packet faces it equals to the difference between minimum and maximum delay at packet. Packet loss: a percent of transmitting packets does not reach to destination. In the proposed method for optimal control and management of flows, we characterize a variety of flows as follow:

- Real-time flows
- Non real-time flows
- Best effort flows

Routing per flow: The current Internet does not allow routing on per-flow. When a packet reaches to a router, it checks both source and destination address of packet with inputs in routing table and forwards it based on the predefined rules which are configured via network operator. In the proposed method, open flow provides the flexibility of defining different types of flows to which a set of actions and rules can be associated. For example, one type of flow may be forwarded using a shortest path routing algorithm while the other flows may follow manually configured routes over the network. In order to avoid complex flow table lookups, flow definitions should be cleverly set and aggregated if possible. In open flow, network devices store the flows and their associated rules in flow tables which are processed as a pipeline shown above. The goal of the pipelined processing is to reduce the packet processing time (Li *et al*., 2014).

Token bucket algorithm: Suitable allocation of network resources is one of the components affecting improvement in throughput and efficiency of network; in the proposed method, we used token Bucket algorithm for suitable allocation of resources to services.

Token bucket algorithm: in this algorithm, there is a token bucket with limited capacity to parameter “burst size” that the tokens are poured to this bucket with the speed equivalents to average speed of packets. If this bucket is filled, additional tokens which are entered later are overloaded and are not entered into the bucket. When a packet is entered provided that size of packet is less than number of existing tokens in the bucket, the tokens equivalent to the size of packet are omitted from bucket. In this state, packet follows the policy and conform is applied on packet. Yet if the sufficient token does not exist in the bucket, bucket will exceed from policy on which exceed action will be applied. Result from exceed action can be as follow: transmission, omission, mark and transmission, mark and continuity.

In the proposed approach, token bucket algorithm to allocate resources to the transmitted requests has been used and service has been provided to each traffic well suited with its nature so as to use resources of network in an optimal way and improve throughput and efficiency of network.

How to assess the user's satisfaction with data: In the proposed method, user's satisfaction with received data is an important parameter to improve throughput and efficiency of network so that improvement of throughput and efficiency of network results in user's satisfaction. As mentioned above some data flows are susceptible to loss, e.g., 'file download' in which increase of loss of packet results in increase of user's dissatisfaction; real time flows are susceptible to delay in case delay increases to a certain extent, resulting in user's dissatisfaction. Best Effort flows have not specific susceptibility, e.g., opening a normal page in the web browser. Video data are susceptible to both loss of packet and delay; the user's satisfaction with a video data is calculated using PSNR formula; in following the procedure and algorithm related to this parameter are proposed.

Signal-to-Noise Ratio (PSNR): This scale discusses on the video signal ratio to be electromagnetic noise and shows the difficulties in transmitting packets. Further to measure and examine quality, the content inside the packet is used after coding and transmitting in the network:

$$\mathrm{PSNR} = 10\log_{10}\left[\frac{255^2}{\mathrm{MSE}}\right]$$

$$\mathrm{MSE} = \frac{\sum_{p=p1}^{p2}\sum_{M=M1}^{M2}\sum_{N=N1}^{N2}\big(d(p,m,n)-o(p,m,n)\big)^2}{(P2-P1+1)(M2-M1+1)(N2-N1+1)}$$

In this equation, d (p, m, n) and o (p, m, n) represent the undermined and real values of frame p which is in row m and column n. Mehrian *et al*. (2016) suggested a method to solve tasks which contains utterances with wide range of background noise that are not well represented in the training data.

The algorithm pertaining to PSNR: To estimate quality of receiving video original video is required in this algorithm. PSNR criterion calculates maximum ratio of energy amount of signal to noise energy. The equation below displays how to calculate PSNR for nth frame of video, obtained of the difference on lightness component of transmitting frame and original video:

$$\mathrm{PSNR}_{(n)db} = 20\log_{10}\left(\frac{V_{peak}}{\sqrt{\frac{1}{N_{col}N_{row}}\sum_{i=0}^{N_{col}}\sum_{j=0}^{N_{row}}\left[Y_s(n,i,j)-Y_D(n,i,j)\right]^2}}\right)$$

Where:

$V_{peak} = 2^k-1$	=	In which K represents number of bits used for each pixel
N_{row} and N_{col}	=	Dimensions of video frame
Y_s (n, i, j)	=	Lightness rate of pixel (i, j) in nth frame of transmitted video
Y_D (n, i, j)	=	Lightness rate of this pixel in the received frame. Ranking PSNR is as follow to determine quality of received video

Now with regard to Table 2 if the obtained values are greater than 4, the user will be satisfied with the supplied service if the values are <4, user will be dissatisfied with the proposed video service.

The proposed approach: In this study, firstly the proposed architecture of network and then the proposed Workflow are proposed.

Architecture of network: In the proposed method, traffic flows have been categorized to optimize throughput and efficiency in software-defined data center using capability of SDN networks in network management and control. Any type of traffic is examined in its specific queue and the priority is given to the traffic based on user's satisfaction or dissatisfaction with received data, whereby allocation capability of bandwidth, delay, jitter is managed and network efficiency is improved.

The proposed architecture with SDN includes Open Flow switches and one server; a SDN controller is the socket server as application layer to receive a request from the user shown in Fig. 4.

As mentioned previously, control and data transmission part have been separated from each other such that it can consider this network as follow shown in Fig. 5.

In the proposed method, the capability of network controller planning has been used to manage and control switches to optimize throughput and efficiency in software-defined data center. It should pay a particular attention to this point that client-side programs are mentioned in application layer, thus daemon must exist to translate the name of program to a word or network language associated to it, e.g., it can used socket technology that the client socket in user's device must

Table 2: User's satisfaction in video data

PSNR[db]	MOS[1]
>37	5 (very good-excellent)
31-37	4 (good)
25-31	3 (average)
20-25	2 (poor)
<20	1 (bad)

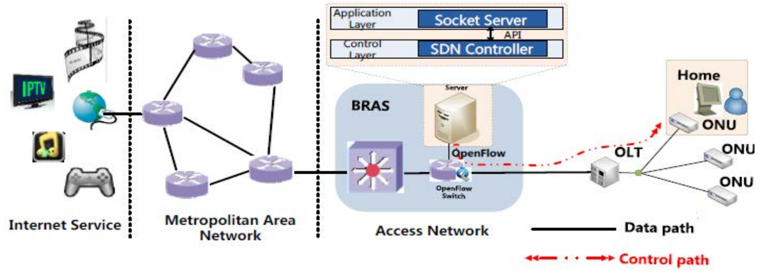

Fig. 4: Network architecture based on SDN

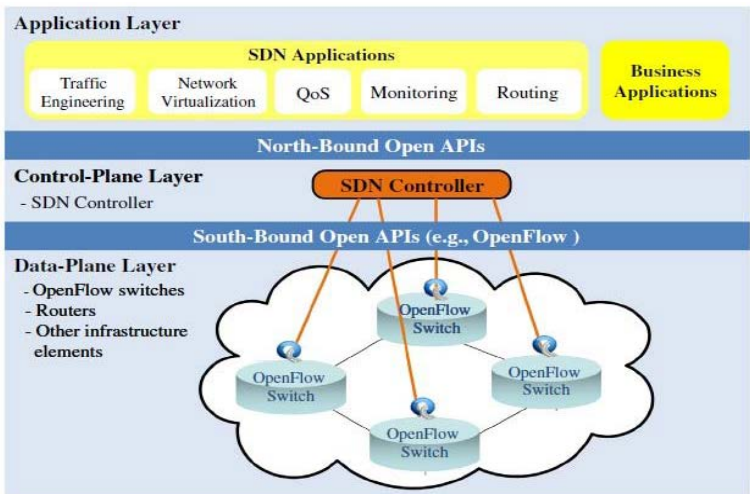

Fig. 5: Three-layer architecture of SDN networks

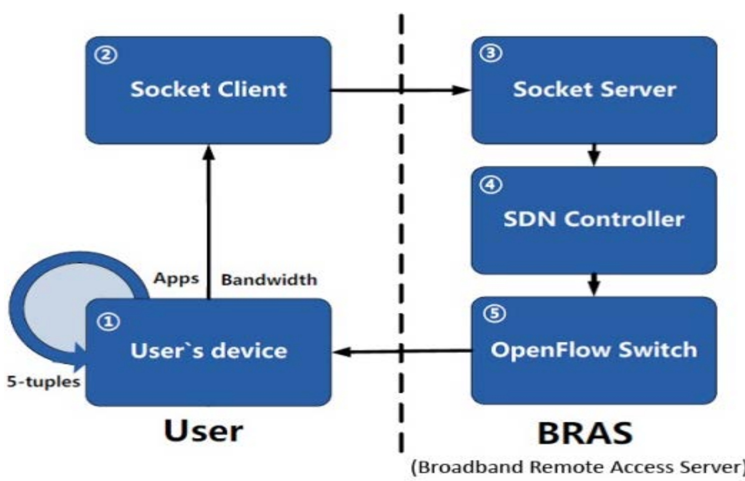

Fig. 6: Workflow for allocation of bandwidth to the user program using socket technology

connect to server socket. Then, client socket must send the information specified in previous stage in a standard format (srcIP, srcPort, proto, dstIP, dstPort, minBW, maxBW) to the server socket. Figure 6 represents a sample of workflow for allocation of bandwidth to the user program using socket technology.

Workflow of the proposed method: Workflow of the proposed method to optimize throughput and efficiency in software-defined data center based on SDN with giving priority to the requests based on QoE is as follow: A new flow for this data is registered in the controller for each new data entered into the controller, allocated to that space and network sources. As mentioned above when the information passes through SDN controller, a flow will be sent to open flow switch regarding the asked information. Flow table adapts the packets with a specific flow entry and specifies a function that must be applied on that packet. On the other hand, due to user's dissatisfaction with received data in the services existing in the network, a flow with higher priority level is considered in controller for that traffic so as to increase user's satisfaction with received traffics. Indeed, function of this method is targeted in improving throughput and efficiency of the system based on management of controller and flows in the controllers.

Here, it should be noted that allocating new flows for the new data entered in the controllers and adding flow to the controllers to increase users' satisfaction with the purpose of increasing throughput and efficiency in software-defined data center result in increase of flows and number of levels of flows in controllers and increase of complexity and inefficiency and decrease of throughput because adding new flows for the entered data and creating flows with higher priority for existing services in the controller result in increase of complexity, resulting in decrease of system throughput and network inefficiency. For this, we have considered management of controllers based on type of service. Four categories of general flow have been defined; in following if a new flow is entered into the controller or increase of the related service priority is required due to users' dissatisfaction, this increase of priority will be made in the associated group, related to the group without any effect on other groups. By increasing number of requests to avoid loss of flows with lesser priority and pave the way for optimal allocation of resources to defined traffic levels which have been described using token bucket algorithm, the sources are provided in an optimal way to increase throughput and efficiency to the services. For instance, in best effort traffic, a token is allocated to a flow with more susceptibility regarding number of existing packets in queue and priority of each queue. For instance, more token is allocated to flows susceptible to jitter such that quality of service has been ensured in the network and throughput and efficiency have improved in software-defined data center regarding nature of traffic. Open flow switch will transmit the status of switch to the controller after processing and forwarding packets. After completing the stages above, the user will experience the improved program and will enable to register his request shown in Fig. 7.

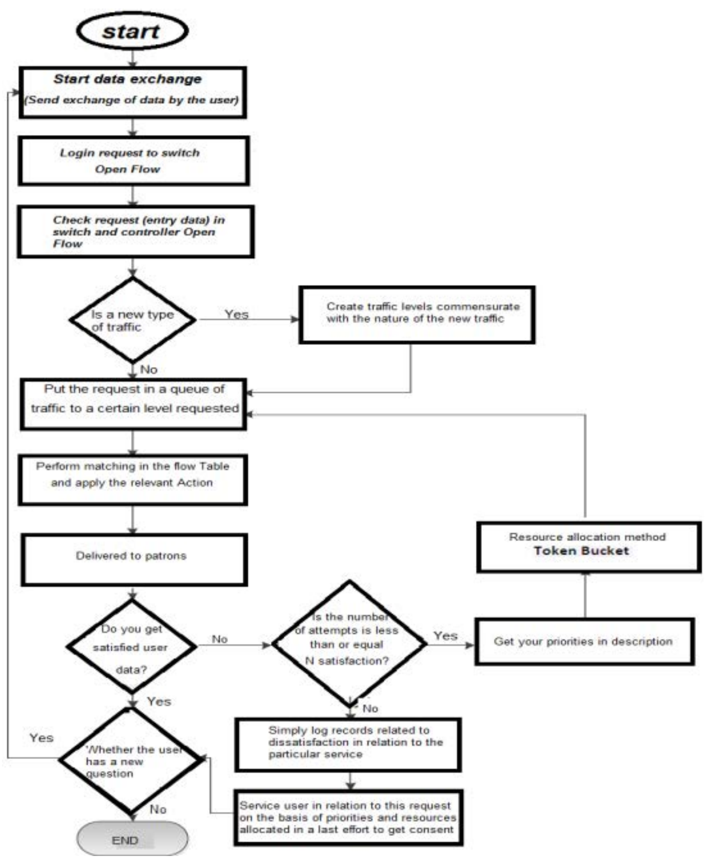

Fig. 7: Flowchart of the proposed method

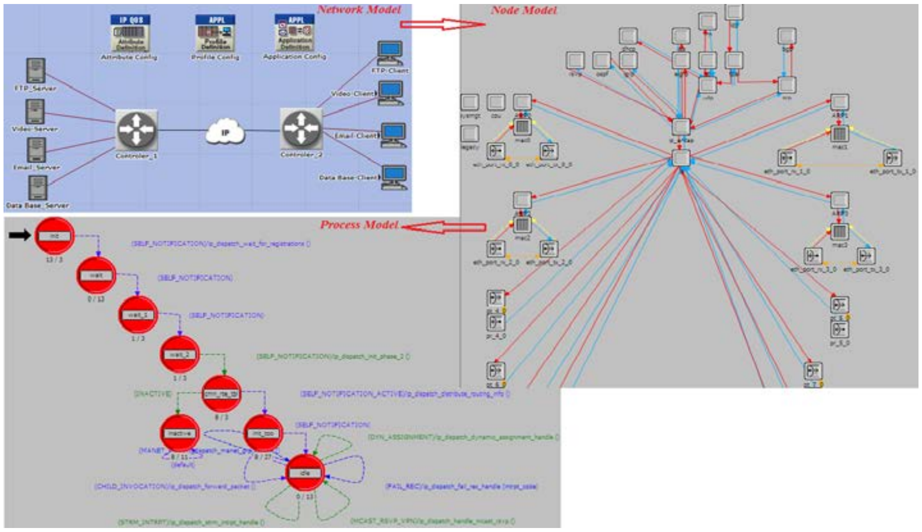

Fig. 8: General procedure of the simulation structure

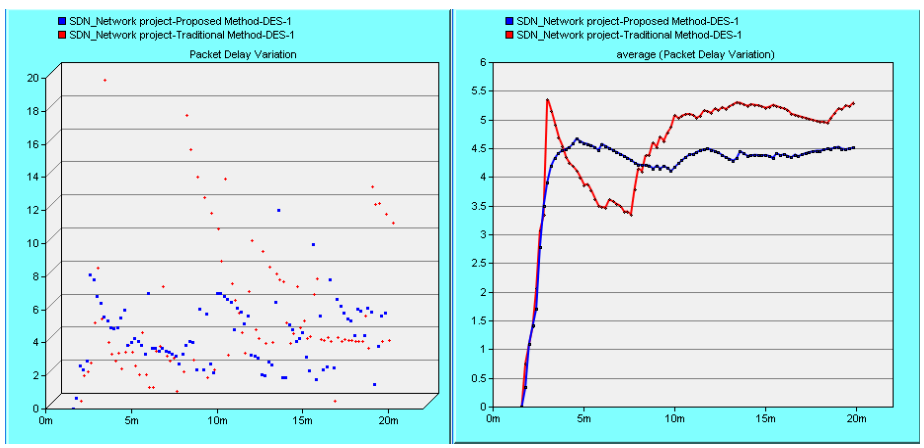

Fig. 9: Rate of changes in delay in the proposed method to the previous method

RESULTS AND DISCUSSION

This part of the study represents the simulation and analysis of the proposed method. Figure 8 represents the general procedure of OPNET simulator regarding the simulation including a schematic of network level, nod level and process level in a general schematic.

Function of previous method: In the simulation, we compare our proposed method with a method in which the controller creates a flow with higher priority per any user's dissatisfaction and the complexity occurs in the network due to lack of characterization of a variety of flows and lack of use of suitable algorithm for optimal allocation of resources. This is different than gradient-based flow method proposed in Babaeean *et al*. (2008), since the allocation of resources is not as complex as computing gradient.

OPNET simulator: In this research, OPNET simulator has been used to simulate the proposed method and compare it with previous method.

Simulation parameters: The parameters considered in the simulation are proposed in Table 3. In this section with regard to what mentioned previously, the results have been proposed and some explanations have been elaborated about any tested criterion. The changes in delay to function of methods in throughput. Changes in delay refer to the volatilities of delay. Figure 9 represents rate of changes in delay in the proposed method to the previous method. In the proposed method, network sources have been allocated to services more efficiently and resulted in improvement in delay to increase efficiency and throughput using allocation of network sources via the concept of token bucket. With regard to

Table 3: The parameters considered in the simulation

Parameters	Explanations	Values
Simulator	OPNET simulator	Version 14.5
Dimensions of enviornment	Simulation context	Virtual internet network
Packet	Data packet volume	1024 byte
Destination of packets	To which destinations, the services are transmitted?	Network users
Simulation type	The duration to compare methods in simulation	1200 sec
Type of services	The services supplied and supported in the network	Video (VCR quality video) FTP (high load), email (high load), data base (high load)
Ending simulation	The ending time for collection of results	End of simulation
Startup time of simulation	Start up of the network after the early setup	100 sec
Occupied channel	An assumed traffic which has existed on channel	(Byte/sec) 50000

Simulation and analysis of results

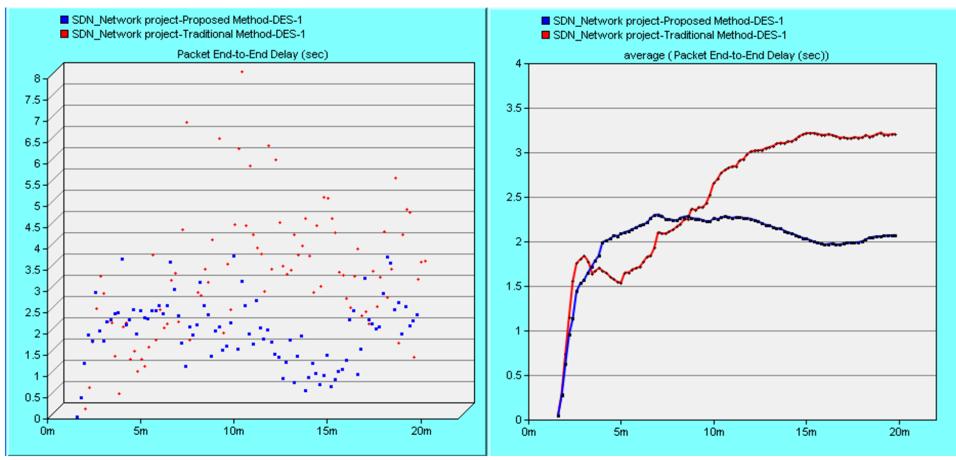

Fig. 10: End-to-end delay in the proposed method compared to previous method

what mentioned in previous method at the area of different flows to controller and allocation of new flows under arrival of new services and users' dissatisfaction, firstly this method has been assumed efficient yet in following it has resulted in inefficiency of previous method during simulation by increase of levels of flows in controllers and inefficiency of controller for various flows. Yet with regard to leveling traffics dependent on their types, increase of flows and management of flows under users' dissatisfaction, this has caused more efficiency and improvement in the proposed method during simulation than previous method.

End-to-end delay based on function of the methods under comparison in throughput: End-to-end delay encompasses the duration of time that the packet is transmitted from source to destination. This criterion refers to the spent time to transfer of services, differed than function of methods under comparison in throughput. Figure 10 represents end-to-end delay in the proposed method compared to the method under comparison with the purpose of increase in efficiency and throughput. In the proposed method, this criterion has been found with more favorable process due to leveling services with the same nature and increase of their priority in the related level under users' dissatisfaction.

Rate of packet receiving: Rate of packet receiving refers to the throughput and efficiency of SDN network in exchanges of services. Throughout SDN network regarding the simulation, the services of the sources have been exchanged for destinations which these services are conducted in controllers to reach to the final destination dependent on the methods under comparison. This process throughout network on one hand and improvement of function of SDN controllers per various services on the other hand have resulted in improvement of throughput and efficiency in network. Figure 11 represents the diagrams relating to the explanations given at the area of rate of packet receiving in OPNET simulator in the methods under comparison.

Rate of packet retransmission due to loss of packets under reduction of throughput and efficiency: Rate of packet retransmission refers to the packet retransmission due to loss of packets throughout the network under reduction of throughput and efficiency than function of the methods under comparison. Due to optimal allocation

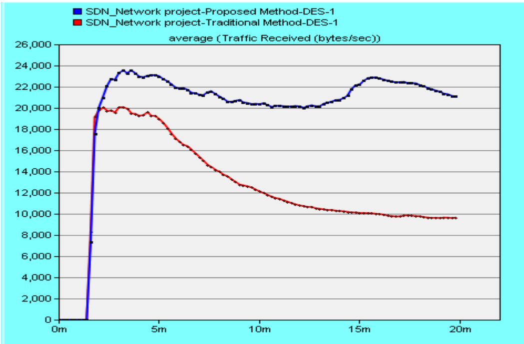

Fig. 11: Rate of packet receiving in the proposed method than previous method

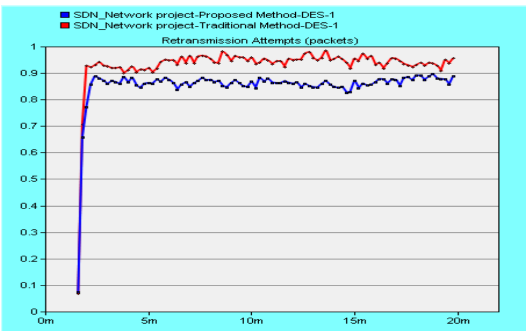

Fig. 12: Rate of retransmissions in the proposed method compared to previous method

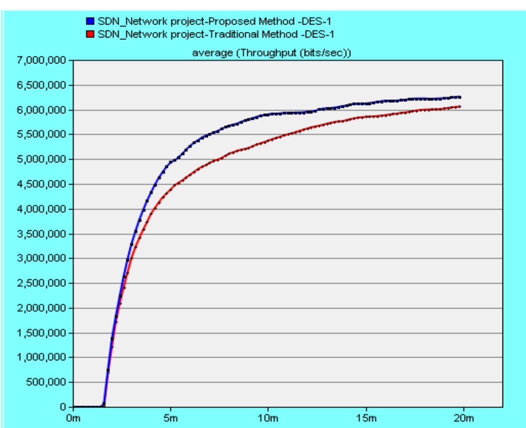

Fig. 13: Rate of throughput in the proposed method compared to previous method

of resources by means of the concept “token bucket” regarding nature of each service in the proposed method than previous method without any specific mechanism for allocation of resources and support from services throughout the network, this has caused lesser packets due to lack of optimal allocation of resources and reduction of retransmission. This criterion has varies dependent on function of the proposed method in throughput and efficiency which will be followed by improvement in throughput and efficiency under support. Figure 12 represents the diagrams relating to the explanations given about throughput and rate of retransmission in the methods under comparison in OPNET simulator.

Return of network than throughput and efficiency: Return of network refers to real data transmission in SDN network dependent on function of the methods under comparison in treatment with each service on how to allocate resources and how to treat with services throughout the network. Increase in return of network has been witnessed in the proposed method than previous method, due to general function of method in increase of network throughput and efficiency and the proposed results in previous sections. This increase of return has come to realize based on leveling services in SDN controllers, management of flows, analysis of each service, suitable allocation of resources and use of the concept “token bucket”. Figure 13 displays the diagrams pertaining to the explanations given at the area of return rate of SDN network in OPNET simulator.

CONCLUSION

In this research, we proposed a method to optimize throughput and efficiency in software-defined data center through improving function of controllers in software-defined data center. In this method with regard to the restrictions on scalability of network and diversity of services in these networks to improve throughput and efficiency, we have considered management of controllers based on type of service. Indeed, it is worked out based on type of requested service and the traffic level. Four categories for general flow have been defined that each service is considered in one of these four categories regarding type of its susceptibility.

In following if a new flow is entered to the controller and/or increase of service priority is needed due to users’ dissatisfaction, this increase of priority will be made in the related group without any impact on other groups. Then with regard to what mentioned in the second section, resources are provided to increase throughput and efficiency to the services using concept “token bucket”. As mentioned, an emphasis has been put on controller planning to increase use of network in the proposed method. In future, it can consider development of the programs at user level in which the users enable to manage network resources regarding requirements of a program using software-defined data center. In addition, the application of Data Envelopment Analysis (DEA) (Tavana *et al*., 2016; Li *et al*., 2016; Mobin *et al*., 2015) in optimizing the system can be considered in the future research.

REFERENCES

Babaeean, A., A.B. Tashk, F. Barzin and S.M. Hosseini, 2008. Target tracking using mean shift and dynamic directional gradient vector flow. Proceedings of the 40th Southeastern Symposium on System Theory, March 16-18, 2008, New Orleans, LA., USA., pp: 366-370.

Bayestehtashk, A., I. Shafran and A. Babaeian, 2016. Robust speech recognition using multivariate copula models. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, March 20-25, 2016, Shanghai, China, pp: 5890-5894.

Cziva, R., D. Stapleton, F.P. Tso and D.P. Pezaros, 2014. SDN-based virtual machine management for cloud data centers. Proceedings of the IEEE 3rd International Conference on Cloud Networking, October 8-10, 2014, Luxembourg, pp: 388-394.

De Oliveira, R.L.S., C.M. Schweitzer, A.A. Shinoda and L.R. Prete, 2014. Using Mininet for emulation and prototyping software-defined networks. Proceedings of the IEEE Colombian Conference on Communications and Computing, June 4-6, 2014, Bogota, Colombia, pp: 1-6.

Egilmez, H.E., S.T. Dane, K.T. Bagci and A.M. Tekalp, 2012. OpenQoS: An OpenFlow controller design for multimedia delivery with end-to-end quality of service over software-defined networks. Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, December 3-6, 2012, Hollywood, CA., USA., pp: 1-8.

Govindarajan, K., K.C. Meng, H. Ong, W.M. Tat, S. Sivanand and L.S. Leong, 2014. Realizing the Quality of Service (QoS) in Software-Defined Networking (SDN) based cloud infrastructure. Proceedings of the 2nd International Conference on Information and Communication Technology, May 28-30, 2014, Bandung, Indonesia, pp: 505-510.

Li, K., W. Guo, W. Zhang, Y. Wen, C. Li and W. Hu, 2014. QoE-based bandwidth allocation with SDN in FTTH networks. Proceedings of the IEEE Network Operations and Management Symposium, May 5-9, 2014, Krakow, Poland, pp: 1-8.

Li, Z., M. Mobin and T. Keyser, 2016. Multi-objective and multi-stage reliability growth planning in early product-development stage. IEEE Trans. Reliab., 65: 769-781.

Lin, W.C., C.H. Liao, K.T. Kuo and C.H. Wen, 2013. Flow-and-VM migration for optimizing throughput and energy in SDN-based cloud datacenter. Proceeding of the IEEE 5th International Conference on Cloud Computing Technology and Science, Volume 1, December 2-5, 2013, Bristol, UK., pp: 206-211.

Mehrian, S.H.Z., S.A.R. Amrei and M. Maniat, 2016. Structural health monitoring using optimizing algorithms based on flexibility matrix approach and combination of natural frequencies and mode shapes. Int. J. Struct. Eng., Vol. 7, No. 4.

Mobin, M., Z. Li and M.M. Khoraskani, 2015. Multi-objective X-bar control chart design by integrating NSGA-II and data envelopment analysis. Proceedings of the 2015 Industrial and Systems Engineering Research Conference, May 30-June 2, 2015, Nashville, TN., USA.

Nowruzpour Mehrian, S.M., M.H. Naei and S.Z. Mehrian, 2013. Dynamic response for a functionally graded rectangular plate subjected to thermal shock based on LS theory. Applied Mech. Mater., 332: 381-395.

Salvadori, E., R.D. Corin, A. Broglio and M. Gerola, 2011. Generalizing virtual network topologies in OpenFlow-based networks. Proceedings of the IEEE Global Telecommunications Conference, December 5-9, 2011, Kathmandu, Nepal, pp: 1-6.

Tavana, M., Z. Li, M. Mobin, M. Komaki and E. Teymourian, 2016. Multi-objective control chart design optimization using NSGA-III and MOPSO enhanced with DEA and TOPSIS. Expert Syst. Applic., 50: 17-39.

Vaziri, M.R., S.N. Mehrian, M.H. Naei and J.Y.S. Ahmad, 2015. Modification of shock resistance for cutting tools using functionally graded concept in multilayer coating. J. Thermal Sci. Eng. Applic., Vol. 7, No. 1. 10.1115/1.4028982.

Wette, P., M. Draxler and A. Schwabe, 2014. Maxinet: Distributed emulation of software-defined networks. Proceedings of the IFIP Networking Conference, June 2-4, 2014, Trondheim, Norway, pp: 1-9.