Quality of Service Control and Resource Prioritization with Software Defined Networking

Melih A. Karaman*†, Burak Gorkemli*, Sinan Tatlicioglu*, Mustafa Komurcuoglu* and Ozgur Karakaya*
*ARGELA, Research and Technology Business Unit, Maslak, Istanbul 34469, Turkey
†Dept. of Electrical and Electronics Engineering, Koc University, Sariyer, Istanbul 34450, Turkey
Email: melih.a.karaman@ieee.org, {burak.gorkemli, sinan.tatlicioglu, mustafa.komurcuoglu, ozgur.karakaya}@argela.com.tr

***Abstract*—In this study, we aim to analyze how Software Defined Networking (SDN) can help to ensure a high-quality uninterrupted Voice over Internet Protocol (VoIP) service accompanied by video for prioritized users and entities in a network congested with background traffic. Guaranteeing a level of Quality of Service (QoS) and resource prioritization are key concepts for future networks (e.g., 5G, NGSON) and may be required in an emergency situation such as an international disaster recovery operation. Providing QoS in such scenarios with the help of SDN is promisingly feasible, via limiting bandwidth, assigning flows to different queues and/or adapting routing decisions based on network conditions. The experiments conducted throughout the study are the substances of the EC FP7 project OFERTIE and Celtic-Plus project SIGMONA. Here we compared loss rate, latency and the jitter as QoS metrics under several scenarios and results showed that we can flexibly manage the QoS by reducing more than 5% loss rate and more than 50% of latency and jitter.**

I. Introduction

Software Defined Networking (SDN) is a new paradigm that promises to change the ossified, complex and cumbersome structure of traditional IP and cellular networks by enabling separability of the control plane from the data plane and programmability of the future networks. Conventional IP networks are hard to manage, error-prone and generally based on manual configuration with a vertical integration logic. Thus, it is difficult for these networks to adapt to the new defined policies and organize themselves against to changes, loads and faults. On the other hand, SDN is well-known with its two main characteristics (viz., decoupling and network programmability) by breaking this vertical integration logic into decoupled control and data planes and by providing the flexibility of developing network applications and creating services that directly interacts with the infrastructure.

The tremendous growth in data traffic requires ubiquitous availability, sufficient bandwidth and dynamic management with efficient configuration capability for future networks to enable network evolution, abstraction and simplification. Emerging SDN solutions are resolving these problems and addressing the aforementioned challenges day by day, and it will continue its evolution to let the revolution in the network domain. A detailed definition and more information about SDN concept, layers and architecture are available in [1] and [2] including an in-depth analysis of the hardware infrastructure, southbound and northbound application programming interfaces (APIs), network virtualization layers, network operating systems (SDN controllers), network programming languages, and network applications. For a more standard perspective research focused on OpenFlow, which is the most popular SDN protocol/standard, the reader is recommended to review [3] and [4].

The new trend in both cellular and public networks is pushing Future Internet and future networks to guarantee a predefined level of Quality of Service to their users or the devices/machines that are attached on. This is simply because the future networks are not being designed to depend on the infrastructure; quite the contrary, they should be capable of instant service and network creation via the virtualization flexibility provided by SDN. Some of the prominent scenarios include emergency situations such as an international disaster recovery operation, where a first responder team should be able to communicate with each other and probably with their headquarters with minimal distortion. Other public safety applications and medical applications also fall in this category. Managing the enterprise cloud with charging policies based on provided QoS and video-to-video live chat are other applications that may be listed to emphasize the importance of QoS control and resource prioritization for upcoming networks.

The experiments and the study presented in this paper is an output of the joint works of OFERTIE[1] [5] and SIGMONA [6] projects. OpenFlow Experiment in Real-Time Internet Edutainment (OFERTIE), aims to use SDN approaches to improve delivery of an emerging class of distributed applications for the Future Internet known as Real-Time Online Interactive Applications (ROIA). In [7] OFERTIE's OpenFlow and Network as a Service (OpenNaaS) paradigm is detailed. SDN Concept in Generalized Mobile Network Architectures (SIGMONA), aims at evaluation, specification and validation of a Software Defined Mobile Network concept designed onto the SDN, network virtualization and cloud computing principles.

The remainder of this paper is organized as follows: Section I-A provides the state-of-the-art in the literature. Section II details the system model and the configuration of the testbed. Moreover, it defines metrics to measure the QoS and describes the test cases. Subsequently, Section III provides the performance evaluation of the proposed SDN-enabled solution under different parameter configurations. Finally, Section IV concludes the paper.

[1]This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 318665.

A. Literature Review

Here we provide a brief review about the literature. The latest articles in the literature give insight and express the criticality of the topic in all aspects. In [8], the authors propose novel QoS extensions to distributed control plane architectures for multimedia delivery over large-scale, multi-operator SDNs. It provides topology aggregation and link summarization methods, an optimization framework for flow-based end-to-end QoS provision, and two scalable distributed secure inter-domain QoS routing control plane designs. In [9], the authors present an SDN-enabled convergent network for efficient resource management, E2E QoS enforcement, and flexibility and scalability for future network evolution to benefit from the SDN and converged network techniques. Similar to our experiments, they implemented some promising applications of the proposed network and evaluated based on an open test platform to show its superiority to current network architecture. [10] gives integrated traffic control structures for SDN while providing traffic engineering (TE) and fast failure recovery (FFR) features for class-of-service (CoS)-aware flow aggregates.

In [11] the authors propose an SDN-enabled capacity sharing framework for User-Centric Networks (UCNs) in order to mitigate the problem of sharing limited network capacity and resources efficiently and fairly by taking into account current network load and conditions, and QoS requirements. Likewise, another use of a different network domain is in [12]. It extends this context-aware composition and delivery concept to utilize Next Generation Service Overlay Networks (NGSONs [13]) over SDN. Similarly, [14] combines SDN with Autonomic Network technologies to construct a software defined self-managing solution for the future network. The importance of Self-Organizing Network (SON) concept is described in [15] by one of the authors of this paper. Previously, the two authors of the paper proposed system architectures for application management in SDN as well [16].

II. Baseline System Model, Test Case and Metric Definitions

This business experiment aims to analyze how SDN can help to ensure a high-quality uninterrupted VoIP service accompanied by video in an emergency situation. On the other hand, there will also be regular users trying to communicate with one another in a network congested with background traffic, which is created by sources other than the VoIP service. These users should also receive some QoS, although less than premium, so that the impact of background traffic may be limited. Providing QoS in such scenarios with the help of SDN is very feasible, via limiting bandwidth, assigning flows to different queues and/or adapting routing decisions based on network conditions. In this experiment, Argela SDN Controller, so called *yakamOS*, is utilized to manage QoS for different flows, which communicates with Argela SIP Server for receiving Service Level Agreement (SLA) parameters per VoIP subscriber and making QoS decisions based on these parameters. Argela SIP Server that is utilized during the tests is a development version of the Wirofon Server used in Turk Telekom's network for giving VoIP service to its subscribers. Hence, apart from the emergency case, this experiment has another important business potential: Transferring Wirofon into an SDN-enabled VoIP service for easy resource and QoS management. The following stakeholders are involved in the experiment: Internet Service Provider (ISP), Turk Telekom as VoIP Provider and VoIP Users (premium and regular).

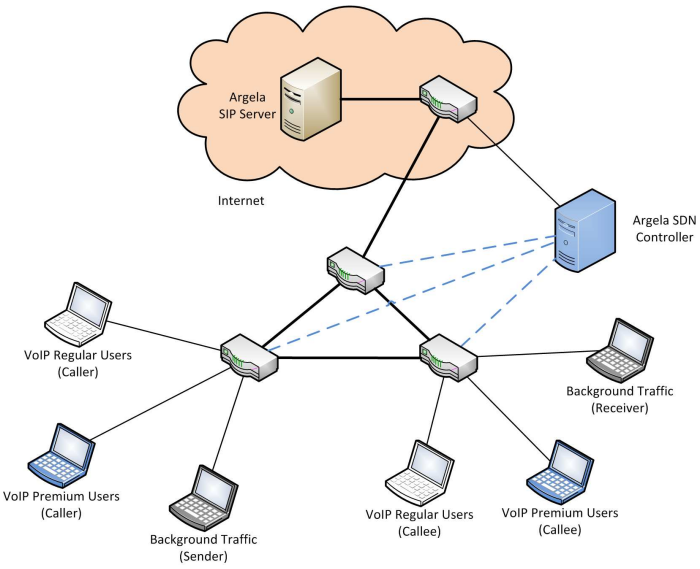


Fig. 1. Topology of the SDN test-bed deployed

In order to push the SDN test-bed to its limits and make tests under extraordinary conditions, we have gone far more than Wirofon's daily usage. But since we have used SIPp[2], which is a popular SIP testing tool for generating SIP traffic, it is possible to run the tests with less demanding parameters to realistically simulate the bandwidth usage of Turk Telekom's VoIP solution. In addition, SDN enables fine granular control over network traffic, where it is possible to classify and prioritize VoIP data (audio and video separately). Hence, Required Extra Capacity (REC) for VoIP is kept at minimal, if not zero. It is possible to differentiate audio and video flows of a VoIP session with SDN, via the source and destination ports of the connection. That is because in VoIP service, audio and video data are sent using different transport layer ports. Determination of the ports that are to be used can be achieved by two methods: The SDN Controller can communicate with the SIP server and learn which ports are to be used for audio and video, or the SDN Controller can capture SIP signaling traffic and parse "SIP INVITE" and "SIP 200 OK" messages to get the port information. Once emergency service officers (i.e., first responders) are classified as premium users, it is possible to prioritize their traffic via bandwidth management, priority queuing and adaptive routing. In the results given in Section III, it is evident that the premium VoIP traffic can be prioritized over background traffic and regular VoIP traffic. It is also possible to prioritize VoIP audio flows over VoIP video.

A. Baseline System Model

Argela test-bed is composed of multiple OpenFlow enabled switches, controlled by the Argela SDN Controller via OpenFlow, which connect VoIP caller and callee pairs, as well as background traffic generators. The topology of the test-bed is

[2]http://sipp.sourceforge.net/

illustrated in Figure 1. All VoIP clients connect to the Argela SIP Server through the OpenFlow enabled switches. There exists a separate connection between Argela SDN Controller and the SIP Server, which is used for managing QoS for VoIP users. Below are the details of the nodes taking part in the experiments:

1) Argela SIP Server: SIP Proxy and RTP Bridging server, which handles SIP signaling and bridges RTP flows between callers and callees. A version of this server runs in Turk Telekom's network, with the "Wirofon" brand, handling live VoIP traffic.

2) Argela SDN Controller: Server in charge of controlling the SDN network. It is capable of limiting bandwidth, giving priority to flows and performing adaptive routing depending on congestion. It also communicates with applications over a RESTful API, for instance with the Argela SIP Server, for managing QoS. During the tests, parameters for identifying premium and regular VoIP subscribers and their flows are exchanged between Argela SIP Server and SDN Controller.

3) VoIP Regular/Premium Users: VoIP users are emulated via SIPp, which is a popular open source test tool / traffic generator for the SIP protocol. For the audio and video data a pre-recorded SIP session is utilized, which is composed of a one minute video and audio data, with video data encoded with H.264 at about 430 kbps and audio encoded with G.722 at about 70 kbps, totaling to 500 kbps. Regular users are programmed to send only audio data, whereas premium users send both audio and video.

4) Background Traffic: Background traffic is generated via iperf[3], which is highly used in the Internet community for generating TCP and UDP traffic for various purposes. During our tests, we used iperf for injecting UDP traffic into the network.

B. Test Cases

Following are the variables that we controlled during the experiments: number of VoIP premium/regular users, number of concurrent VoIP premium/regular calls, background traffic bandwidth, and QoS parameters (minimum bandwidth/loss/latency) of VoIP premium/regular users. The parameters that we measured during the tests are as follows: latency and jitter for audio/video packets, loss rate for audio/video packets, sent/received audio/video data rate, sent/received background traffic rate.

This experiment is divided into three subsections, each of which focuses on a different aspect of the overall scenario. First, we want to see the effect of background traffic on VoIP service, and do tests with varying background traffic volumes. Second, we try to analyze the result of regular VoIP users on the premium subscribers. Last, we perform tests to see how video traffic affects audio flows. In all these test sections, after seeing the effect that we are after, SDN is utilized to limit the examined effect and increase the quality experienced by VoIP users.

[3]https://iperf.fr/

C. QoS Metric Definitions

We defined three metrics that are commonly used in the literature to measure the QoS, these metrics are namely; loss rate, latency and jitter. The calculations of these metrics are based on the following model. Assume the sender side sends N packets to the receiver side, and i^{th} packet is transmitted at time t_{s_i} and received at the receiver at t_{r_i} where $t_{r_i} > t_{s_i}$. Ignoring Operating System (OS) and framing delays by assuming they are slightly same at both ends, we can calculate the latency Δ_i for i^{th} packet as,

$$\Delta_i = t_{r_i} - t_{s_i}. \tag{1}$$

Jitter, denoted by δ, is known as the deviation of the latencies, hence we used the average absolute value deviation from the mean of the latencies. The mean latency μ_Δ is calculated as,

$$\mu_\Delta = \frac{1}{\|\mathbb{C}\|} \sum_{\forall i \in \mathbb{C}} \Delta_i, \tag{2}$$

where $\mathbb{C}$ denotes the set of indices for correctly transmitted packets and $\|\mathbb{C}\|$ is the cardinality of this set. Hence lost packets are not being taken into account. Consequently, the jitter is calculated as follows,

$$\delta = \frac{1}{\|\mathbb{C}\|} \sum_{\forall i \in \mathbb{C}} |\Delta_i - \mu_\Delta|. \tag{3}$$

Finally, the loss rate is calculated as the percentage of the lost packets. Assume M out of the N sent packets are transmitted correctly where $M = \|\mathbb{C}\|$. Hence $N - M$ packets are lost in the transmission phase, then the lost rate Λ is straightforward to compute as,

$$\Lambda = \frac{N - M}{N}, \tag{4}$$

where $N \neq 0$, later Λ is converted into percentage format for the sake of simplicity in the evaluation phase.

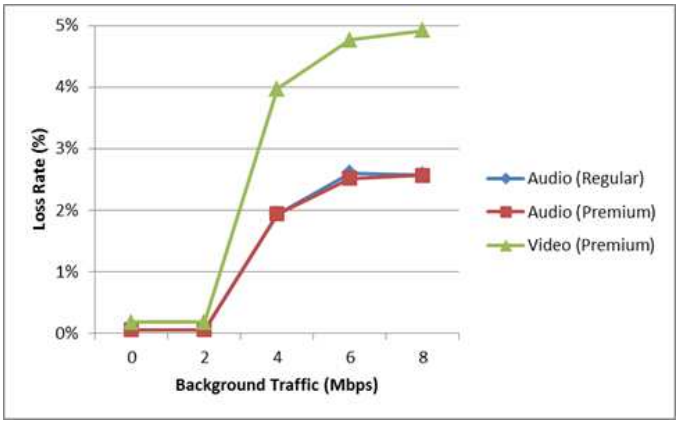
III. Experimental Results and Discussions

A. Analyzing the effects of background traffic

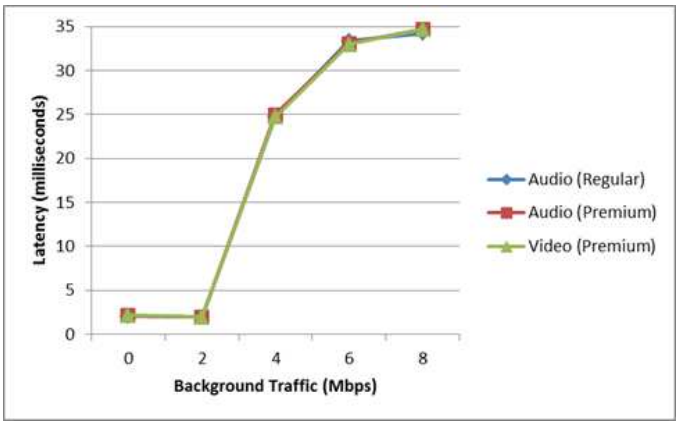
First, we focus on the effect of background traffic on VoIP service, and perform tests with UDP traffic that have transmission rates varying from 0 Mbps (no UDP traffic) to 8 Mbps. As shown in Figure 2, when there is no QoS control, the effect of background traffic becomes obvious after a threshold value, where both video and audio data face loss and latency. Utilizing SDN to enable QoS for VoIP users and in that way limiting background traffic helps to keep VoIP service above a certain quality, which is to be determined by the SLAs between subscribers and service providers, as shown in Figure 3.

B. Analyzing the effects of regular user traffic

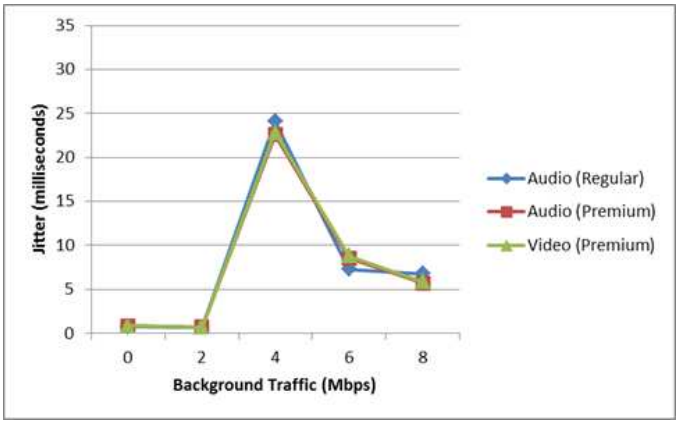
As the number of concurrent regular (audio) calls increases, premium calls, especially video sessions start to be affected, in terms of increased loss, latency and jitter, as shown in Figure 4. In this part of the experiment, we kept the number of concurrent premium calls (audio and video) at two, and increased the number of concurrent regular calls. At 20 concurrent regular calls, the premium video flows started to experience about three percent loss rates, while the latency and jitter values were



(a) Loss Rate

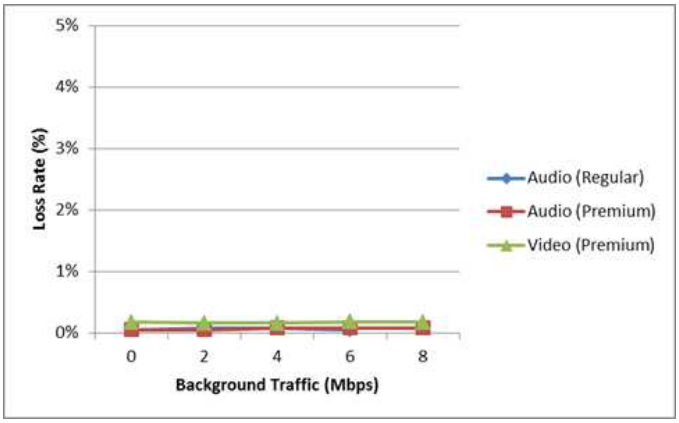


(b) Latency

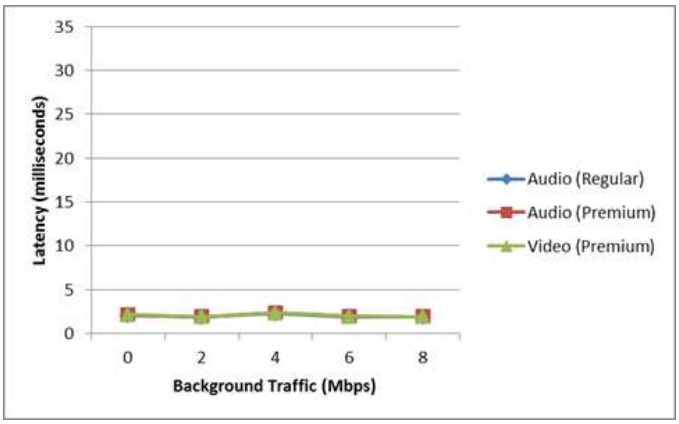


(c) Jitter

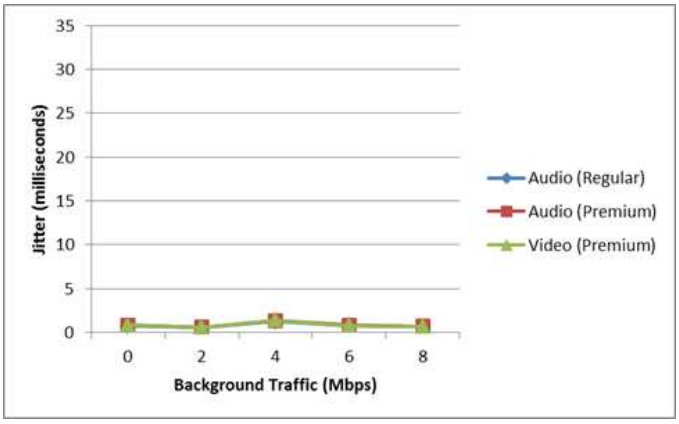
Fig. 2. Performance under varying background traffic with QoS disabled



(a) Loss Rate



(b) Latency



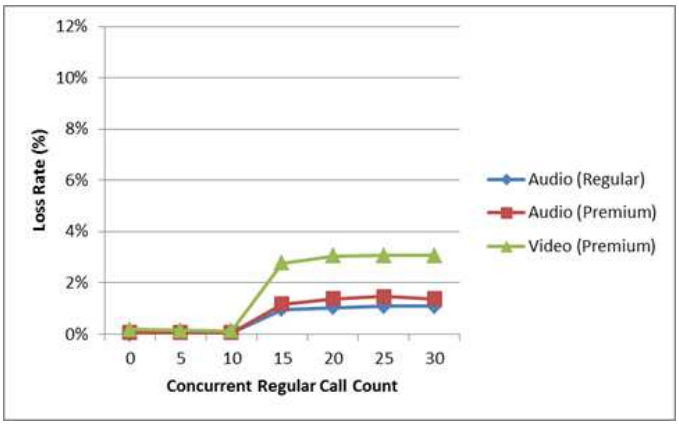
(c) Jitter

Fig. 3. Performance under varying background traffic with QoS enabled

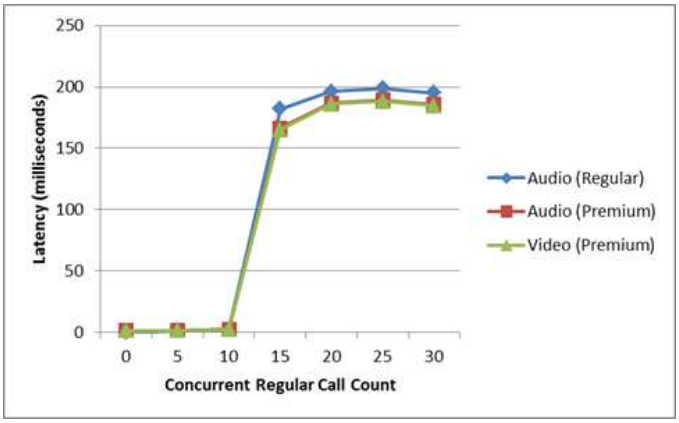
about 200 milliseconds and 150 milliseconds, respectively. Enabling QoS and giving priority to premium flows minimizes the effects of regular calls, keeping loss rate under one percent for premium audio flows and under one and a half percent for premium video, in return for loss rates reaching to 12 percent for regular audio data. Similar improvements are experienced for latency and jitter values, as indicated in Figure 5.

C. Analyzing the effects of video traffic

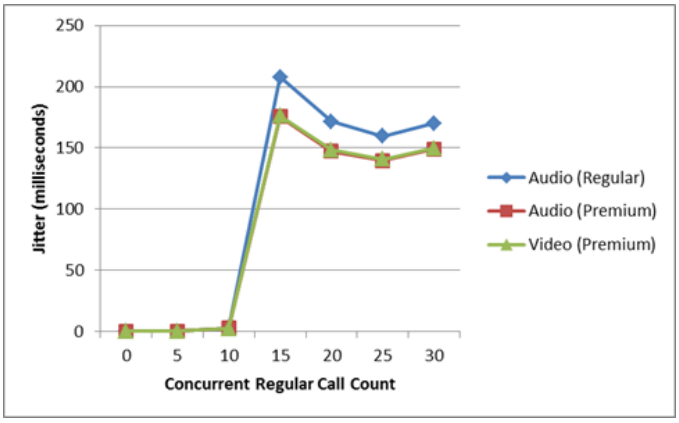
In order to see the effects of video traffic on audio, we run tests with different number of concurrent premium VoIP calls, having both audio and video flows, where no regular call existed. As the number of concurrent premium calls increases, both the video and audio flows start to face loss and increased latency/jitter values, as shown in Figure 6. It is clear that as network gets congested with new premium calls, it is better to give preference to audio flows and guarantee an acceptable quality at least for audio, at the price of degraded video. For this reason, once the QoS is enabled in the tests, we treat audio and video flows differently, prioritizing audio data over video. In this way, the effects of congestion is limited, and we can guarantee that the loss and jitter values are kept at



(a) Loss Rate

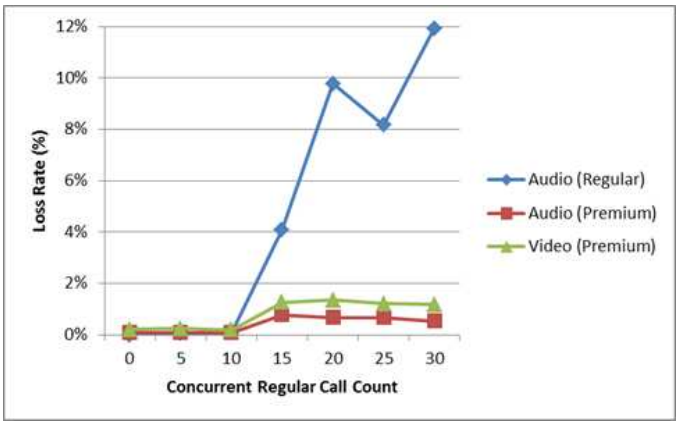


(b) Latency

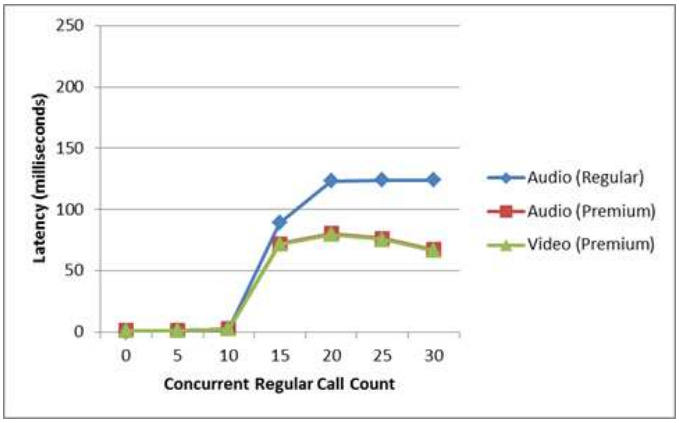


(c) Jitter

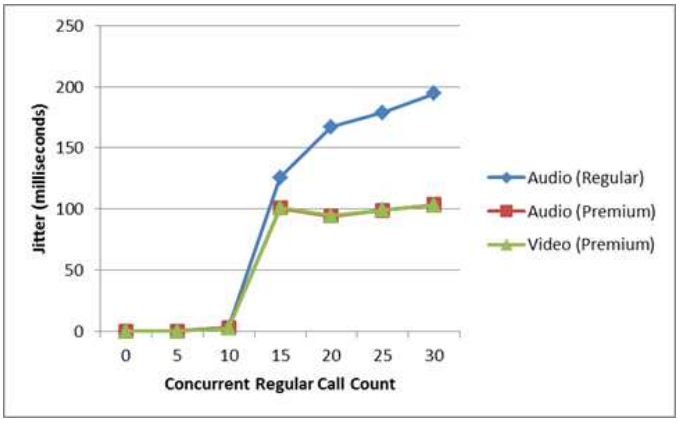
Fig. 4. Performance under varying concurrent regular calls with QoS disabled



(a) Loss Rate



(b) Latency



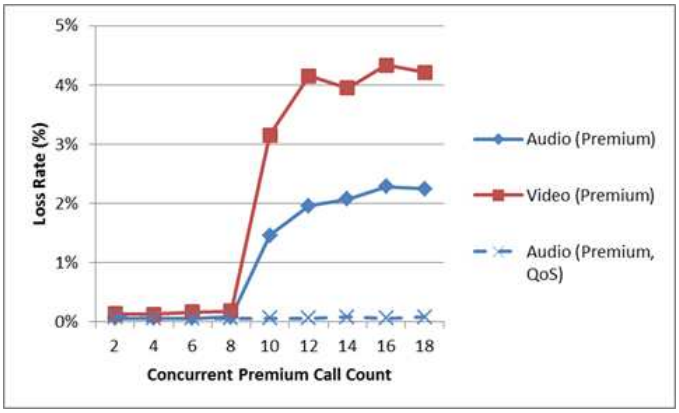
(c) Jitter

Fig. 5. Performance under varying concurrent regular calls with QoS enabled

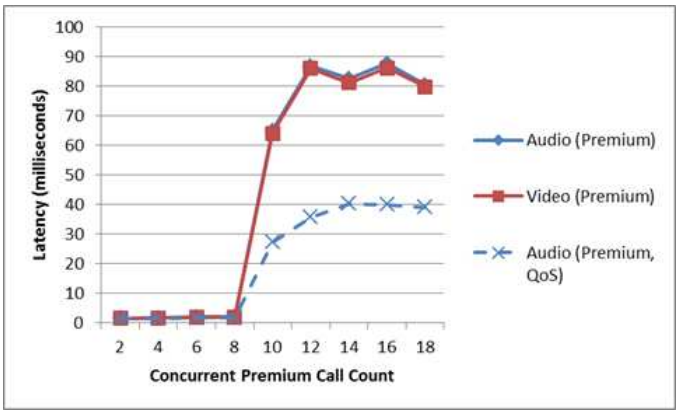
minimum for audio, and the latency is kept below a threshold, as can be noted in Figure 6. The loss rates for video flows are over 50 percent after eight concurrent calls when the QoS is enabled, hence loss/latency/jitter values for video are not shown in Figure 6.

IV. Conclusion

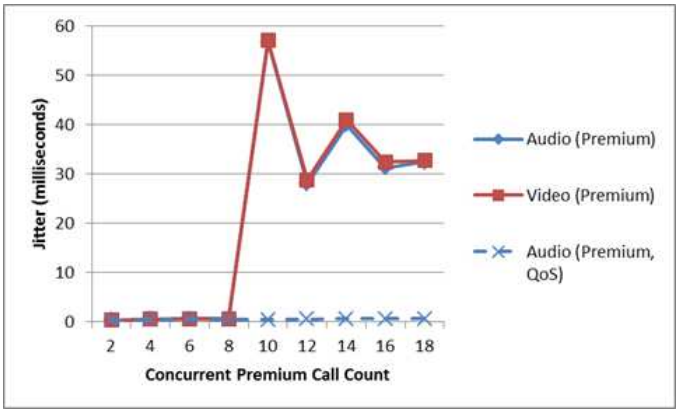
In this study, which is a substrate of the OFERTIE and SIGMONA projects, we utilized the use of SDN to provide a guaranteed QoS to premium users in a VoIP system congested by both from VoIP and non-VoIP background traffic by limiting bandwidth, assigning flows to different queues and adapting routing decisions based on network conditions. In addition, audio traffic is efficiently prioritized over video in order to keep communication reliable for heavy-loaded networks. Results revealed that the flexibility enabled by SDN improved the QoS in the network with a reduction in loss rate more than 5% and more than 50% of reduction in latency and jitter. Accordingly, we can conclude that this enhancement can be further applied to other networks and applications as well. As a future work, we will integrate into this framework the traffic and resource management algorithms developed as part of the SIGMONA



(a) Loss Rate



(b) Latency



(c) Jitter

Fig. 6. Performance of premium audio under varying concurrent premium calls

project to further enhance QoS paremeters.

REFERENCES

[1] D. Kreutz, F. Ramos, P. Verissimo, C. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, Jan 2015.

[2] W. Xia, Y. Wen, C. Foh, D. Niyato, and H. Xie, "A survey on software-defined networking," *Communications Surveys Tutorials, IEEE*, vol. PP, no. 99, pp. 1–1, 2014.

[3] J. Tourrilhes, P. Sharma, S. Banerjee, and J. Pettit, "Sdn and openflow evolution: A standards perspective," *Computer*, vol. 47, no. 11, pp. 22–29, Nov 2014.

[4] F. Hu, Q. Hao, and K. Bao, "A survey on software-defined network and openflow: From concept to implementation," *Communications Surveys Tutorials, IEEE*, vol. 16, no. 4, pp. 2181–2206, Fourthquarter 2014.

[5] [Online]. Available: http://www.ofertie.org/

[6] [Online]. Available: http://celticplus.eu/Projects/Celtic-Plus-Projects/2012/SIGMONA/sigmona-default.asp

[7] I. Bueno, J. Aznar, E. Escalona, J. Ferrer, and J. Antoni Garcia-Espin, "An opennaas based sdn framework for dynamic qos control," in *Future Networks and Services (SDN4FNS), 2013 IEEE SDN for*, Nov 2013, pp. 1–7.

[8] H. Egilmez and A. Tekalp, "Distributed qos architectures for multimedia streaming over software defined networks," *Multimedia, IEEE Transactions on*, vol. 16, no. 6, pp. 1597–1609, Oct 2014.

[9] W. Tan, J. Zhang, C. Peng, B. Xia, and Y. Kou, "Sdn-enabled converged networks," *Wireless Communications, IEEE*, vol. 21, no. 6, pp. 79–85, December 2014.

[10] W. Su, C. Liu, C. Lagoa, H. Che, K. Xu, and Y. Cui, "Integrated, distributed traffic control in multidomain networks," *Control Systems Technology, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.

[11] B. Nunes, M. Santos, B. de Oliveira, C. Margi, K. Obraczka, and T. Turletti, "Software-defined-networking-enabled capacity sharing in user-centric networks," *Communications Magazine, IEEE*, vol. 52, no. 9, pp. 28–36, September 2014.

[12] F. Paganelli, M. Ulema, and B. Martini, "Context-aware service composition and delivery in ngsons over sdn," *Communications Magazine, IEEE*, vol. 52, no. 8, pp. 97–105, Aug 2014.

[13] "Ieee standard for the functional architecture of next generation service overlay networks," *IEEE Std 1903-2011*, pp. 1–147, Oct 2011.

[14] W. Wendong, Q. Qinglei, G. Xiangyang, H. Yannan, and Q. Xirong, "Autonomic qos management mechanism in software defined network," *Communications, China*, vol. 11, no. 7, pp. 13–23, July 2014.

[15] M. Karaman, G. Gur, M. Ergen, and E. Anarim, "Spectrum-aware dynamic frequency reuse (sadfr) for son-based ofdm a hetnets," in *Wireless Communications and Networking Conference (WCNC), 2014 IEEE*, April 2014, pp. 2558–2563.

[16] M. A. Karaman, M. Parlakisik, B. Gorkemli, and O. Ozkasap, "Application management on software defined networks," *The International Symposium on Advances in Software Defined Networks (SOFTNETWORKING 2015)*, April 24, 2015, submitted to.