

NFV-Based Mobile Edge Computing for Lowering Latency of 4K Video Streaming

Linh Van Ma, Van Quan Nguyen, Jaehyung Park, Jinsul Kim

School of Electronics and Computer Engineering

Chonnam National University

77, Yongbong-ro, Buk-gu, Gwangju 500-757, South Korea

linh.mavan@gmail.com, quanap5@gmail.com, hyeoung@jnu.ac.kr, jsworld@chonnam.ac.kr

Abstract—Ultra-high quality video streaming is expected to be the standard for multimedia in the next decade. However, it poses many problems with the current internet transmission technologies, such as network latency, stalling while a video is playing back. Researchers and network operators have spent much attention on solving such problems such as the latest video compression technique using for 4K and 8K video. In this article, we present another approach to improving ultra-high quality video streaming over the internet mainly focusing on mobile edge computing (MEC). Specifically, we propose an NFV-based MEC to transmit ultra-high quality multimedia efficiently. Besides, our proposed streaming system is dynamic adaptive streaming which supports streaming seamlessly with bandwidth fluctuation. As a result, our system provides 4K and 8K video streaming efficiently with lower latency up to 10% compared to existing approaches.

Index Terms—Mobile Edge Computing, Network Function Virtualization, Ultra-high Quality Video, 4K, 8K, Latency

I. INTRODUCTION

Video streaming quality has been improved significantly compared to the last decade since the development of technologies as well as network transmission techniques. However, a user continuously requires higher experience while service providers are trying to maximize their profit. For example, 4K screen resolution is broadly selling in the market. In oppose, 4K video content is not commonly produced or published due to copyright. Moreover, network transmission still has many problems during delivery multimedia between providers and demanders. First, video streaming requires stable bandwidth to ignore stalling while playing back. Quality of service (QoS) or quality of experience (QoE) is increased along with the allocated bandwidth for a particular network. Secondly, multimedia requires enough computational capacity to compress and decompress video content at the processing side and receiving side. As such, QoS and QoE can be only improved if we could allocate powerful computational resource for both side. Thirdly, network service usually composes of several layers which might increase latency once a packet is passing each network layer. It poses a question that how we can reduce such latency to improve service response to a particular request from users.

Dynamic adaptive streaming [1] is a technique which has recently emerged to solve bandwidth fluctuation problems. Video streaming quality seamlessly changes along with bandwidth condition. For instance, if the network bandwidth is

high around 10Mbps, a user might have experience with high-quality such as 1080p (progressive video). In contrast, if the allocated bandwidth is about 500Kps, the streaming might be provided with low quality such as 360p. If the network bandwidth fluctuates between 10Mbps and 500Kbps, the video streaming quality adaptively adjusts without stalling in play-back. The technique has several version of implementation, such as MPEG-DASH (Dynamic Adaptive Streaming over HTTP) [1], [2], HAS (HTTP Adaptive Streaming) [3]. In spite of its network adaptation, it still has some problem to determine which segments are suitable for a particular network condition meaning that we need to implement intelligence algorithms with context-awareness to predict and download segments in advance. Network function virtualization [4] is a platform to solve recent network equipment allocation issues. It allows and spins up network components simply and prevent network chaos. With this technology, we can virtualize every single network component and manage them efficiently. For example, we can virtualize a transcoding component when a user uploads a video to a DASH server, then it can transcode the video source file to DASH content into small segments. This task requires intensive computation resource since it must encode and decode the file.

Mobile edge computing [5] is a new concept network architecture which was proposed by European Telecommunications Standards Institute (ETSI). It solves the emerging problem from data processing. Traditionally, we have a core network to process every request coming from network users. However, in the big data era, processing such requests causes latency in responding requests. In the near future, they will become pervasive. These applications require a higher memory, battery energy, and computing power than that can be acquired on the resource-constrained mobile devices [6]. In fact, mobile devices normally possess some limitation about battery energy, computation capability, etc. From the practical demand with high traffic well as the current limited resource on mobile communication, that force researcher to find a new way to integrate novel technologies into the network. Especially, the time requirement for real-time applications (e.g., video streaming), Mobile Edge Computing (MEC) and Network Function Virtualization has emerged as the remedy.

Consequently, in this article, we present a research on NFV-based mobile edge computing to efficiently delivery ultra-high

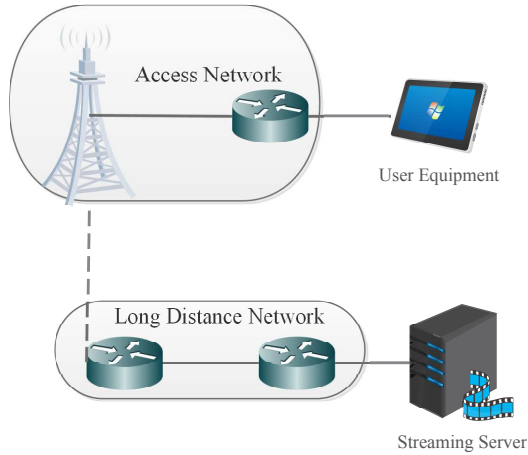


Fig. 1. Two network segments for wireless devices.

quality video streaming over the internet. First, we propose a DASH-based streaming system. Secondly, we address a reason why we need mobile edge computing and network function virtualization in the discipline. Subsequently, we present a context-awareness for ultra-high quality streaming including how DASH segments are retrieved. Finally, we present the research finding in the experiment section.

II. NFV-BASED MEC SYSTEM

In the video delivery process of wireless devices, network packets must pass through two kinds of network segments. The two network has different characteristics as shown in Fig. 1. More specifically, access network that connects subscribers to the streaming service has low latency due to a small number of subscribers. In oppose, long distance network has much higher latency since packet must transport worldwide with long distance. Though DASH is implemented to deliver content over HTTP, it actually transmits packets employing TCP which might degrade performance between the two networks.

Consequently, we come up with a method applying MEC. On the one hand, it provides a similar architecture of content delivery network (CDN) which is widely well known efficient delivery network structure with low latency. In another hand, it solves the problem of CDN policy. Specifically, the content provider must agree with CDN operators to locate content. They cannot flexibly deploy at CDN without any extra charge. Furthermore, we deploy the MEC system using NFV for the following reason. First, NFV can lower negative impacts on local hardware where we cannot economically allocate expensive equipment for small tasks. Secondly, NFV offers the way to decouple data and control planes that is avail for heterogeneous networks for MEC in the local network.

As shown in Fig. 2, latency is increased as we go deeper into the network infrastructure. If MEC locates near user equipment (UE) and contains requested video from UE, it can provide the streaming high-quality efficiently, especially, 4K or ultra-high-quality streaming. However, MEC usually does

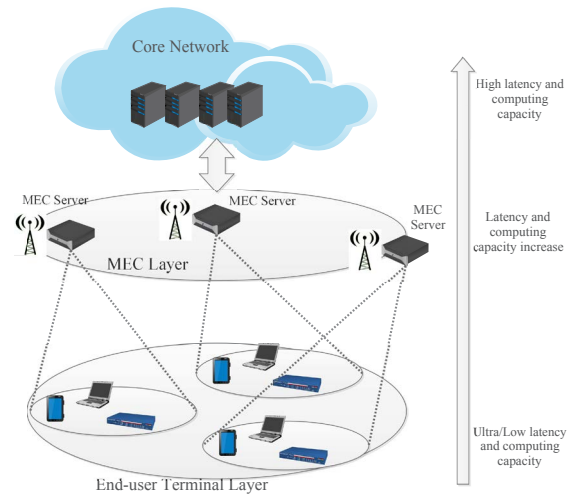


Fig. 2. Overview of MEC video streaming delivery system.

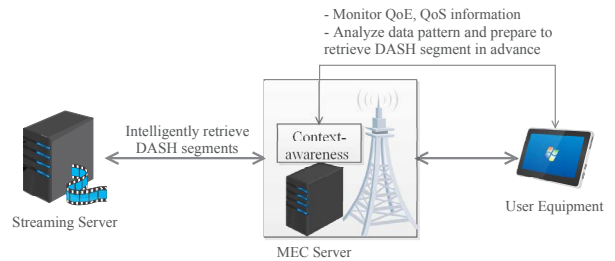


Fig. 3. Content-awareness in the NFV-based MEC streaming system.

not have video at the first request from UE. Crucially, we need a context-awareness component placing in the MEC server as shown in Fig. 3. There have several reasons to implement the component in the MEC server. First, UE commonly has a low computational capacity as well as battery lifetime. Secondly, the component requires statistic and data process with high computation. Last but not least, locating the component might assist other UEs which demands the same source video and reduces waste bandwidth.

In the system, we consider each MEC server as a virtual network function (VNF) which is managed by a virtual network function manager (VNFM) locating in the core network as shown in Fig. 4. As stated in [7], VNFM is the critical component of the NFV architecture, also called management and orchestration (MANO). VNFM is implemented so that it can start and stop VNF based on local demand. By doing this way, we can reduce power consumption to operate a network component as well as improve service readiness.

The policies to retrieve segments is discussed in [8] which is our previous research in DASH streaming in the mobile environment to deal with frequent bandwidth fluctuation employing fuzzy logic and entropy theory. Briefly, the research applies fuzzy logic to deduce the next request segment of DASH streaming as shown in Fig. 5. It considers two input

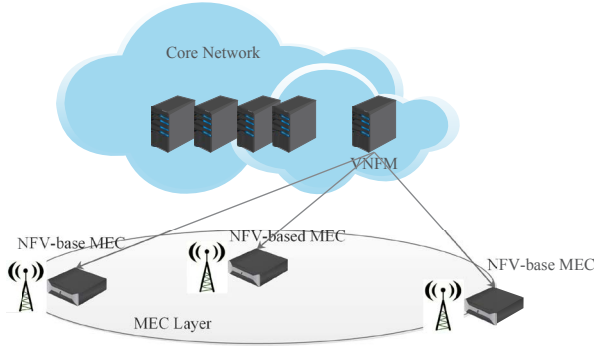


Fig. 4. NFV-based MEC system.

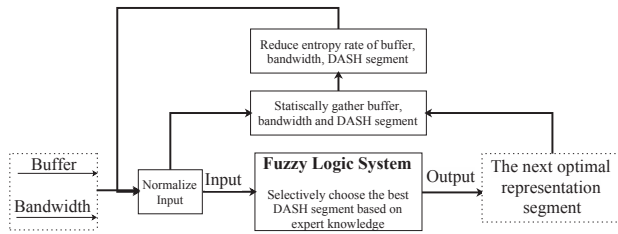


Fig. 5. Illustration of fuzzy-based adaptive streaming method in [8].

parameters which are bandwidth and buffer value at the client side. It first employs moving average technique to reduce statistical fluctuation value in a period. Subsequently, it applies the entropy method to deduce a number of trace-back steps (the measurement point referring back from the current). Theoretically, this research is an extended version of our previous work with the different environment and advanced technology.

III. EXPERIMENT

We implement the system on the main campus of Chonnam National University, Gwangju. In the main campus, we locate the core network which is running on Ubuntu 14.04. More specifically, the main server has two main components which are DASH streaming server and NFV system. DASH is implemented by using Node.js, and NFV employs our cooperating project, FNCP (Future Network Computing Platform) which is the system developed by ETRI (Electronics and Telecommunications Research Institute). The FNCP is built based on the NFV architecture proposed by ESTI (European Telecommunications Standards Institute).

We locate three wireless access points in three different locations which are far from each other. In each access point, we arrange one computer acting as a MEC server which is managed by the core network. Subsequently, we equip three Samsung Galaxy S8 in one location, two smart TVs supporting 4K streaming in another site, and five computers connected to the 8K screen in the last location. All virtualized network function such as DASH server, MEC servers run on Docker [9].

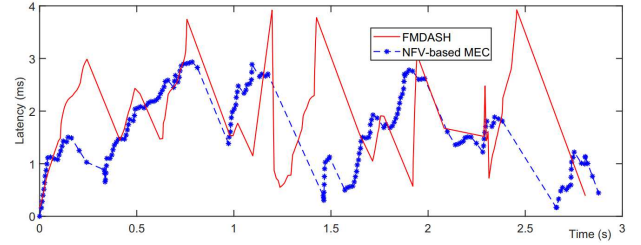


Fig. 6. Latency comparison between NFV-based MEC and FMDASH in [8].

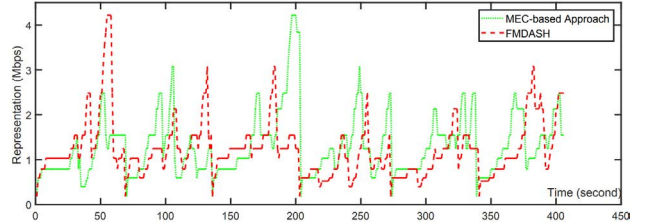


Fig. 7. Segment retrieval comparison between MEC-based approach and FMDASH.

Regarding network, we place a component in the middle between the core network and MEC. This component is also managed by VNFM to simulate network as long-distance connections. In detail, every packet must pass through this component. Network characteristics such as latency, bandwidth is changed and fluctuated unpredictably.

We consider latency as a metric to measure performance between two methods which are FMDASH (Fuzzy Moving DASH Algorithm) and NFV-based MEC. Throughout the experiment, the two method has similar QoS (Quality of Service). More specifically, the result is shown in Fig. 6 with a 4K video 3 minutes length. It states that with the same policies of retrieving segments with NFV-based architecture, the NFV-based system got lower latency around 10% compared to traditional approach core network client driven based.

Recall that, our research-based method, which is addressed in [8], has the lower fluctuation of segment representation when network condition changes frequently. It also utilizes network resource more efficient compared to FMDASH in [10]. However, we did not consider the network latency between clients and the streaming server at that time yet. Consequently, it might be a drawback of our proposed method in the real application. Since our based rules are based on FMDASH, the segment quality retrieval is similar to FMDASH. The comparison between MEC-based approach and FMDASH is shown in Fig. 7. As a result, the MEC-based approach utilizes network resource which is bandwidth more efficiency compared to FMDASH up to 6.3%.

Besides, we also compare network latency while video playback at a client by reusing NS3 source code (Network Simulation Tool), which is provided following the publication of FMDASH [10]. We set up one streaming server and three MEC servers which serve fifty clients randomly. For example,

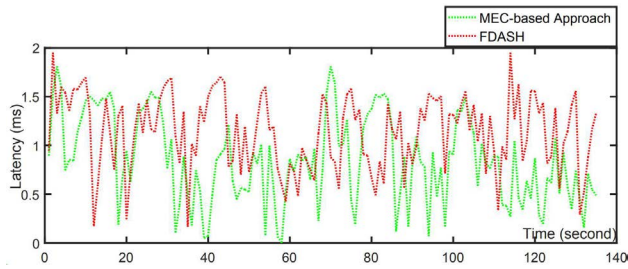


Fig. 8. Latency comparison between our MEC-based approach and FDASH.

a MEC might be in-charge of an incoming response from the streaming server and out-going streaming request from twenty clients, and other two MEC server are in-charge for the rest thirty clients. As a result, the latency of our approach is statically less than 26% compared to FDASH. It is reasonable for the achieved result while the rules-based method to retrieve DASH segment is proved with more efficient result compared to FDASH. Furthermore, our approach is not the server-client driven. Content-awareness component is located on MEC servers which can statically anticipate and deduce the next optimal streaming segment better than client itself.

IV. CONCLUSION

In this research, we presented a research on recent network issues mainly focusing on MEC. The research provides a new approach to transmit 4K or ultra-high-quality video streaming over the current internet infrastructure. In detail, we proposed an NFV-based MEC to transmit ultra-high quality multimedia efficiently. In the experiment, we exploited DASH to build the system streaming. As a result, our system provided 4K and 8K video streaming efficiently with lower latency up to 10% compared to existing approaches.

In the next phase, we are going to spend our effort to improve the context-awareness component by employing the latest research result gotten from data mining, big data literature.

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science, and Technology (MEST) (Grant No. NRF-2017R1D1A1B03034429), [Grant No. NRF2015R1D1A1A01056937]. Besides, this research was supported by the IT R&D program of MSIT (Ministry of Science and ICT), Korea / NIPA (National IT Industry Promotion Agency) 12221-14-1001, Next Generation Network Computing Platform Testbed. Lastly, this research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program(IITP-2018-2016-0-00314) supervised by the IITP(Institute for Information & communications Technology Promotion).

REFERENCES

- [1] J. Kua, G. Armitage, and P. Branch, "A survey of rate adaptation techniques for dynamic adaptive streaming over http," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1842–1866, 2017.
- [2] S. Zhao, Z. Li, D. Medhi, P. Lai, and S. Liu, "Study of user qoe improvement for dynamic adaptive streaming over http (mpeg-dash)," in *Computing, Networking and Communications (ICNC), 2017 International Conference on*. IEEE, 2017, pp. 566–570.
- [3] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hossfeld, and P. Tran-Gia, "A survey on quality of experience of http adaptive streaming," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 469–492, 2015.
- [4] V. Eramo, E. Miucci, M. Ammar, and F. G. Lavacca, "An approach for service function chain routing and virtual function network instance migration in network function virtualization architectures," *IEEE/ACM Transactions on Networking*, vol. 25, no. 4, pp. 2008–2025, 2017.
- [5] B. Liang, *Mobile edge computing*. Cambridge University Press, 2017.
- [6] L. Tianze, W. Muqing, Z. Min, and L. Wenxing, "An overhead-optimizing task scheduling strategy for ad-hoc based mobile edge computing," *IEEE Access*, vol. 5, pp. 5609–5622, 2017.
- [7] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "Nfv: state of the art, challenges, and implementation in next generation mobile networks (vepc)," *IEEE Network*, vol. 28, no. 6, pp. 18–26, 2014.
- [8] L. V. Ma, J. Park, J. Nam, H. Ryu, and J. Kim, "A fuzzy-based adaptive streaming algorithm for reducing entropy rate of dash bitrate fluctuation to improve mobile quality of service," *Entropy*, vol. 19, no. 9, p. 477, 2017.
- [9] D. Merkel, "Docker: lightweight linux containers for consistent development and deployment," *Linux Journal*, vol. 2014, no. 239, p. 2, 2014.
- [10] D. J. Vergados, A. Michalas, A. Sgora, D. D. Vergados, and P. Chatzimisios, "Fdash: a fuzzy-based mpeg/dash adaptation algorithm," *IEEE Systems Journal*, vol. 10, no. 2, pp. 859–868, 2016.