

Analytical Delay Model of virtual CPE for 5G eMBB services in SD-CDN

Müge Erel-Özçevik*, Ferdi Tekçe^{†‡}, Fuat Kayapınar[‡]

*The Department of Software Engineering, Manisa Celal Bayar University, Manisa, TURKEY

[†]The Department of Electrical and Communication Engineering, Yıldız Technical University, Istanbul, TURKEY

[‡]Andasis, Istanbul, TURKEY

Emails: muge.ozcevik@cbu.edu.tr, ferdi@andasis.com, fuat.kayapinar@andasis.com

Abstract—The high bandwidth and moderate latency requirements of enhanced Mobile BroadBand (eMBB) content in 5G are handled by Content Delivery Network (CDN) components such as customer premises equipment (CPE) in edge, branches in fog, and servers in the cloud. The indispensable goal of 5G network providers is meeting these requirements on the same physical component that reduces operational and capital expenditures (OPEX/CAPEX). However, we’ve believed that different configuration policies of providers can be only handled by Software-Defined CDN (SD-CDN). Moreover; due to the flow characteristics of eMBB and handling packet forwarding by only software-based component in an OpenFlow (OF) switch, obtained packet processing delay in conventional CPE has become challenging. Therefore, we propose SD-vCPE with a novel physical block diagram which includes hardware and software-based OF tables. The analytical models of conventional and SD-vCPE are also derived by using G/G/1 Markov queues, considering general eMBB flow characteristics, and SpeedUp parameters in the software part. According to analytical and simulation results, as the resolution of video content in eMBB is increased, the conventional CPE cannot handle delay in the microsecond level; whereas, proposed SD-vCPE outputs approximately 1-millisecond decrease on packet delay which differentiates according to matching rules in TCAM or CPU. Namely, the proposed analytical model for SD-vCPE results in a delay in an acceptable rate while comparing the simulation results performed in MATLAB.

Index Terms—CDN, SDN, eMBB, Markov Model, vCPE

I. INTRODUCTION

In 5th Generation (5G) New Radio (NR); there are different contents available in the cloud market which are newly called enhanced-mobile broadband (eMBB), ultra-reliable low latency (URLLC), and massive machine-type communication (mMTC) [1]. They meet customers with different requirements. For example; URLLC requires strict latency, mMTC uses less bandwidth; whereas, eMBB requires higher bandwidth with moderate latency. To handle these requirements at an acceptable rate, 5G contents are routed by different components of Content Delivery Networks (CDN) such as customer premises equipment (CPE) in edge, branches in fog, and cloud servers in the cloud network.

The indispensable goal of 5G network providers is meeting the requirements of contents on the same physical

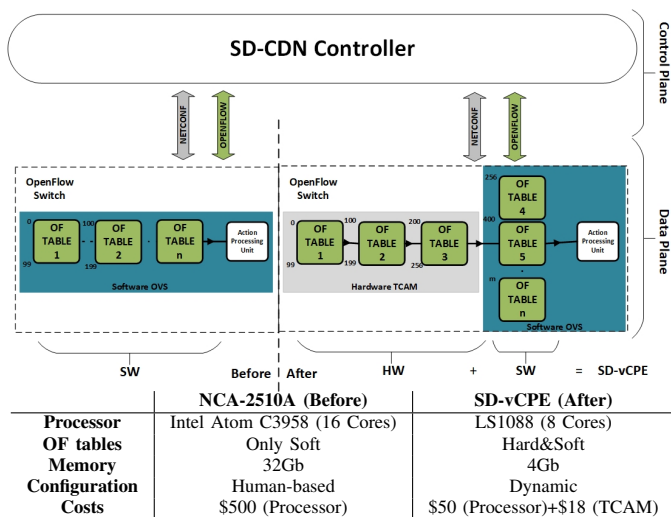


Fig. 1: The OpenFlow Switch Comparison.

component that reduces operational and capital expenditures (OPEX/CAPEX) [2]. However, each provider can have different configuration policies on the same component. This causes extra hardware and bandwidth costs in edge networks [3]. For example; URLLC and mMTC contents, need more than 70 percent of computing and communication resources to handle strict latency requirements [4]. According to Telecom’s motivation, it is believed that Software-Defined Network (SDN) driven CDN (SD-CDN) is the only way to provide different content on the same edge component, i.e. newly called virtual CPE (vCPE) [5][6]. Thanks to the dynamic configuration capability of SD-CDN, the plug-and-play vCPE is available for different content providers.

In Fig.1, the comparison of OpenFlow (OF) switches are given. In the current markets, there is a CPE device named NCA-2510A for edge-users. It has a huge memory enabling OVS on it. Namely, it has only soft OF tables and supports only human-based configuration [7]. However, human-based table configuration is not recommended because of not having any authorization between providers. This should be performed by a controller dynamically. Moreover; we’ve believed that the obtained delay over the conventional vCPE for eMBB content is newly challenging due to its main characteristics. It

has continuous and asynchronous communication where data should be executed by certain delay constraints [8]. Before; eMBB content was routed by HTTP1.0, it is now carried by a newly called Quick User Datagram Protocol Internet Connections (QUIC) protocol that reduces the round trip time of stream [9]. This protocol uses more duplicate packets in the downlink stream and it results in higher traffic content than before. To handle this challenge, we propose Software-Defined virtual Customer Premises Equipment (SD-vCPE) in edge network by adding hardware equipment in sub of OvS based software part. The superiority of our SD-vCPE is enabling dynamic configuration via authorization and isolation between multi-vendors. Thanks to our switch design, not only they can use both soft and hard tables via OF protocol, but also they can dynamically configure tables according to their different requirements. Despite less CPU processing, SD-vCPE is cheaper than a conventional one. Because; the cost of OF switches mostly depends on CPU and memory sizes. NCA-2510A has Intel Atom C3958 (16 Cores) priced \$500; whereas we use LS1088 (8 Cores) and 256 TCAM priced as \$50 and \$18 respectively. Here, SD-vCPE also guarantees acceptable forwarding delay due to dynamically using hard and soft tables.

The whole contributions of SD-vCPE are given below:

- A novel SD-CDN architecture by decouples Data and Control planes,
- A novel physical block diagram for SD-vCPE which includes hard and soft OF tables,
- A novel analytical delay model of conventional and proposed SD-vCPE are driven:
 - by using G/G/1 Markov queues per hard and soft components,
 - by using defined eMBB flow characteristics,
 - by using SpeedUp parameters for processors.

II. SYSTEM MODEL

A. Physical Block Diagram of Proposed vCPE

In Fig.2, the CPU block runs an LS1088 communication processor which has 8-cores and a 1.2GHz clock rate. The reason to select this is enabling data path acceleration architecture (DPAA2) to accelerate the software forwarding capacity of CPU. It has multiple storage devices that are internal and external such as NAND flash, internal ROM, on-chip read access memory (OC-RAM), and external double data rate (DDR4) 4GB RAMs. It has NAND flash which is only used as storage and the CPU firstly needs copying the rules from NAND to internal RAM to run programs. Here, DDRs are used for accelerating the processing time of RAMs. As given the above of the block diagram, the proposed vCPE has different communication interfaces such as 2.4/2.5GHz Wi-Fi, 802.11 a/b/g/n/ac 2x2 MIMO, and USB 2.0.

To synchronize the physical and MAC layers, the CPU uses a data path mac controller (DPMAC). It includes a direct memory access controller (DMAC) which efficiently forwards packet data to CPU RAM, formats data an IEEE

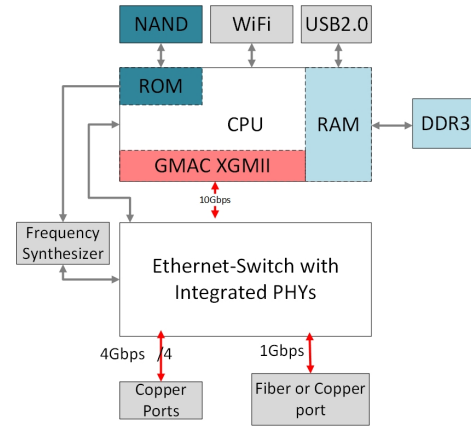


Fig. 2: The Physical Block Diagram of Proposed vCPE.

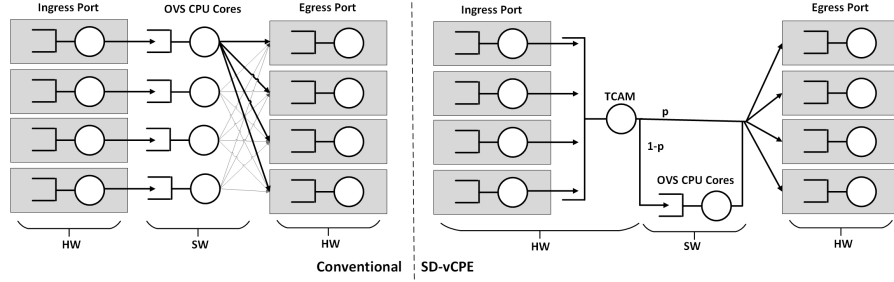
802.3-2002 compliant packet and transmits it to the physical layer, i.e. ethernet switch block[10]. For this communication, it uses a 10-gigabit media-independent interface (XGMII) to transmit and receive Ethernet packets, perform forward error connection (FEC), etc. XGMII is a standard defined in IEEE 802.3 for connecting full-duplex 10GbE [11].

In the physical layer, the proposed SD-vCPE includes an Ethernet switch fabric which communicates with CPU via XGMII interfaces. It has 4-gigabit copper Ethernet ports shown by red arrows numbered from 1 to 4. It also has a 1-gigabit port which is copper or fiber shown by red arrow numbered 5. This block includes ternary content-addressable memory (TCAM) which is hard OF table. The soft OF tables are configured in the CPU. Here, the header of the received packet from ingress ports is firstly matched in TCAM OF headers. If the packet is hit; it can be directly forwarded to egress ports without sending the packet to CPU. If the packet is a miss; the packet forwards to the CPU to looking up soft OF tables in its memory. According to matching rules, the packet is executed. Here, the content processing time of the hard OF table is much higher than soft OF tables. Therefore, the main motivation of the paper for the proposed vCPE is to forward at least 80% of the content via hard OF tables to remove extra packet processing delay.

B. Analytical Delay Model of SD-vCPE

In Fig. 3, the analytical Markov models of conventional and proposed SD-vCPE based OF switches are given. They are modeled by using G/G/1 Markov queues in the left and right sides of the figure, respectively[12]. According to the technical report of conventional CPE, each ingress port is assigned to one core of the CPU. The incoming packets are performed directly in the assigned one. Therefore, we've modeled the packet flow in OF switch by using G/G/1 Markov queues for packet process in ingress ports, OvS CPU cores and egress ports, and the total delay is calculated by using Jackson's theorem [12] as shown in eq.1.

In G/G/1 Markov queues, arrival (λ) and service (μ) rates of incoming packets are assumed to be Gamma distributed. To model eMBB, we focus on UDP based QUIC



$$D_{CON} = \overbrace{D_{Ingress}}^{HW} + \overbrace{D_{CPU}}^{SW} + \overbrace{D_{Egress}}^{HW} \quad (1)$$

$$D_{vCPE} = \overbrace{D_{Ingress}}^{HW} + \overbrace{D_{TCAM}}^{HW} + \overbrace{(1-p) \cdot D_{CPU}}^{SW} + \overbrace{D_{Egress}}^{HW} \quad (2)$$

Fig. 3: Analytical Markov Models of conventional and proposed SD-vCPE OpenFlow switches in SD-CDN.

$$D_{CON}^{eMBB} = \sum_{c=\{Ingress,CPU,Egress\}} \frac{(\lambda^{eMBB})^2(t) \cdot \sigma^2 + (\mu_c^{eMBB})^2(t) \cdot \sigma^2}{2} \cdot \frac{1}{\mu_c^{eMBB} - \lambda^{eMBB}} \quad (3)$$

$$D_{vCPE}^{eMBB} = \sum_{c=\{Ingress,TCAM,Egress\}} \left[\frac{(\lambda^{eMBB})^2(t) \cdot \sigma^2 + (\mu_c^{eMBB})^2(t) \cdot \sigma^2}{2} \cdot \frac{1}{\mu_c^{eMBB} - \lambda^{eMBB}} \right] + (1-p) \cdot \left[\frac{(\lambda^{eMBB})^2(t) \cdot \sigma^2 + (\mu_{CPU}^{eMBB})^2(t) \cdot \sigma^2}{2} \cdot \frac{1}{\mu_{CPU}^{eMBB} - \lambda^{eMBB}} \right] \quad (4)$$

protocol[13][14][15]. We firstly consider M/M/1 Markov model and then we consider this traffic effect on the delay formula. By implementing L'Hospital rule on $\sum n \cdot P_n$, N is calculated as $N = \rho/(1-\rho)$ in M/M/1 queues. By using N , the delay (D) in per M/M/1 component is calculated as $D = N/\lambda = \rho/(\lambda(1-\rho)) = 1/(\mu - \lambda)$. For eMBB traffic type, this delay is calculated as follows:

$$D^{eMBB} = \frac{C_A^2 + C_B^2}{2} \cdot \frac{1}{\mu^{eMBB} - \lambda^{eMBB}} \quad (5)$$

where $C_A = (\lambda^{eMBB})^2(t) \cdot \sigma^2$ and $C_B = (\mu^{eMBB})^2(t) \cdot \sigma^2$ indicates the coefficient of inter-arrival and service time variation. By using Jackson's theorem and eqs.1,5; the delay for eMBB in conventional OF switch is modeled as in eq.3.

As in the right side of Fig.3, the proposed SD-vCPE has one physical TCAM which is a hard OF table. It is assumed that $p\%$ of incoming packets are executed by TCAM without forwarding it user-space, i.e. software part of CPU cores. By using eqs.2 and 5, the delay is calculated as in eq.4.

C. Speed Up of proposed SD-vCPE based OpenFlow switch

According to Amdahl's law, SpeedUp (γ) is defined as the acceleration factor of a processor [16] and calculated as $\gamma = \frac{1}{(1-a) + \frac{a}{m}}$ where a shows the percentage of an OF module that can be operated in parallel independently, and m indicates the number of threads/cores in a processor. In order to reduce extra processing on delay (D_{vCPE}) as defined in eq.4, proposed SD-vCPE enables to implementation of DPAA and also hyper-threading with an increased core number of CPUs [7]. Therefore, we propose two heuristics that speed up SD-vCPE based OF switch as follows:

- **Heuristic 1** (γ_1): It defines DPAA based speed up. While processing a packet, it uses kernel space instead of user

space which extremely accelerates the forwarding time of a switch[7]. According to these results, we define the processing time of accelerated CPU as $\mu_{CPU}^{DPAA} = \frac{\mu_{CPU}}{\gamma_1}$. The speed up (γ_1) effect is taken from previously analyzed data of conventional OvS [7].

- **Heuristic 2** (γ_2): It defines the hyper-threading approach. The accelerated processing time of CPU in SD-vCPE is calculated as $\mu_{CPU}^{Thread} = \frac{\mu_{CPU}}{\gamma_2}$. Here, the expected number of γ_2 is found according to percentage of a [0-1]. Therefore, hyper-threading directly depends on the percentage of a and the number of thread/core as $1 \leq \frac{1}{\gamma_2} < m$. This inequality can be further derived as:

$$\mu_{CPU}^{Thread} = \mu_{CPU} \leq \frac{\mu_{CPU}}{\gamma_2} < m * \mu_{CPU} \quad (6)$$

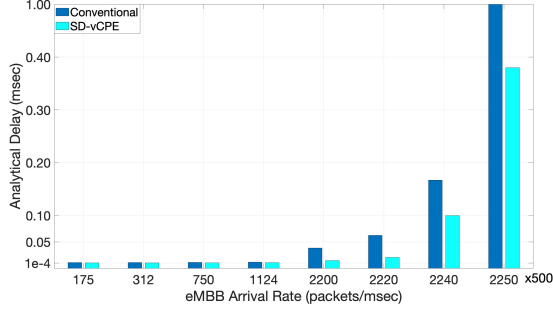
If the both heuristics are performed on software part of SD-vCPE, the accelerated processing time of CPU is calculated as follows:

$$\mu_{CPU}^{DPAA,Thread} = 1 \leq \frac{\mu_{CPU}}{\gamma_1 \cdot \gamma_2} < \frac{m * \mu_{CPU}}{\gamma_1} \quad (7)$$

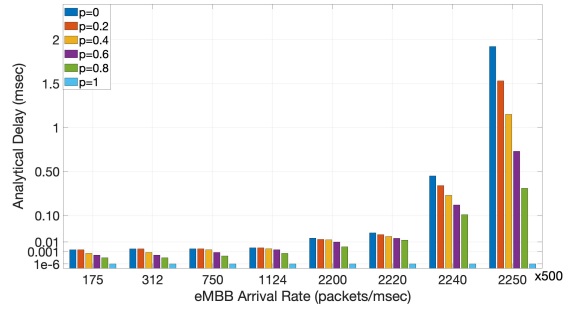
Above the upper level is defined as infeasible space for the acceleration of the processing unit by using 1 to 8 cores CPU. If we use a multi-core and hyper-threading approach on CPU, it outputs nearly x4 acceleration on CPU. If DPAA is also implemented on this CPU of SD-vCPE, the speed-up factor becomes up to x18. If the core number increases to 8 with hyper-threading and DPAA implementation, the theoretic Speed Up limit becomes x23 [7][17][18].

III. PERFORMANCE EVALUATION

The proposed SD-vCPE is evaluated by both analytical delay model and simulation environment in MATLAB^{2019b}

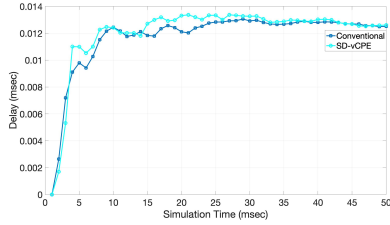


(a) The Analytical Delay (msec).

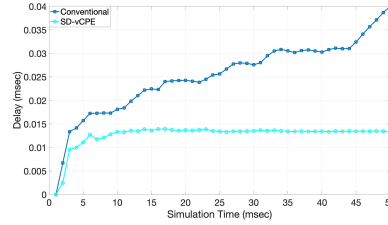


(b) The p effect on the Analytical Delay (msec) of SD-vCPE.

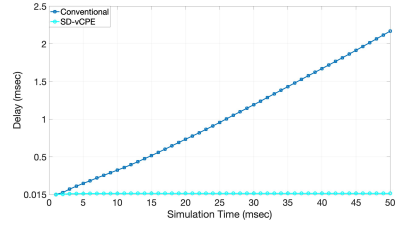
Fig. 4: The Analytical Delay (msec) comparison for conventional CPE and SD-vCPE.



(a) 1080p @30fps eMBB flow



(b) 1080p @60fps eMBB flow



(c) 1440p @30fps eMBB flow

Fig. 5: Simulated Delay (msec) for different eMBB flows.

9.7.0.1190202. Here, the arrival rate of eMBB packets are taken from live bitrate settings of Youtube videos such as 1124, 2220, 2250 (x500) packets/sec for 1080p @30fps, 1080p @60fps and 1440p @30fps video resolutions, respectively. The packet size is determined as 500 bytes. The Markov parameters are determined according to conventional and LS1088 based communication processors[7]. The serving rates of Conventional CPU (2.60GHz), SD-vCPE CPU (1.2GHz), TCAM, and Ing. Eg. ports are 1126, 844, 1128000, and 1350000 packets/msec.

A. Analytical Results

In Fig.4, the Analytical Delay (msec) in eqs.3 and eq.4 are performed. On the left side, the x-axis shows different eMBB arrival rate (packets/sec) from 360p to 1440p video resolutions per flow; whereas the y-axis shows the analytical delay (msec)¹ According to results; the delay is under 5 microseconds up to 2200x500 packets/msec for both conventional and proposed SD-vCPE.² However, conventional OF switch cannot serve eMBB flows in an acceptable rate after 2250x500 packets/msec. SD-vCPE can serve it under 4 msec level for the higher resolutions thanks to the added hard OF tables. On the right side, the p effect on the analytical delay of proposed SD-vCPE is given. As the whole eMBB incoming flows are considered, p can be between 0-1. It has an approximately 1-msec decrease for 2250x500 packets/msec arrival rate which cannot be neglected while considering the end-to-end delay of a flow.

¹The p parameter of proposed SD-vCPE was taken as 0.8 which is believed to be an ideal case for TCAM usage in an OF switch.

²According to IMT-2020, eMBB flows should be served under 4 milliseconds end-to-end delay [19]

B. Simulation Results

In Fig.5, the simulation results of proposed Markov models are evaluated for different eMBB flows. The x-axes show simulation time in msec, and the y-axes output the delay observed by MATLAB. Here; the p of SD-vCPE is taken as 1, namely, all flows are routed by hardware-based TCAM. In Fig.5a, the SD-vCPE and conventional OF outputs delay in 0.013 msec. However; as the arrival rate is increased in Figs.5b and 5c, the gap between conventional and proposed one also extremely rises. Here, the proposed SD-vCPE enables approximately 0.03 msec and 1.9 msec decrease on packet processing delay in an OF switch for 1080p and 1440p.

IV. ACKNOWLEDGEMENT

This work is supported by TUBITAK with a project number: 3200326.

V. CONCLUSION AND FUTURE WORK

This paper proposes a novel SD-CDN architecture. It builds a novel physical block diagram including hard and soft OF tables. A novel analytical model is derived by considering G/G/1 queues, eMBB flow characteristics, and SpeedUp parameters. Due to handling as many flows as in TCAM based OF tables, the analytical delay can be also kept under microsecond levels. As a future work, this analytical model is planned to be compared with the real test results that will be obtained from the physical OF switch. Moreover; the flow matching probability p, will dynamically be managed by AI based SDN controller to keep forwarding delay in acceptable levels for different 5G traffics such as eMBB, URLLC, and mMTC.

REFERENCES

- [1] S. E. Elayoubi, S. B. Jemaa, Z. Altman, and A. Galindo-Serrano, "5G RAN Slicing for Verticals: Enablers and Challenges," *IEEE Communications Magazine*, vol. 57, no. 1, pp. 28–34, January 2019.
- [2] X. Li, R. Casellas, G. Landi, A. de la Oliva, X. Costa-Perez, A. Garcia-Saavedra, T. Deiss, L. Cominardi, and R. Vilalta, "5G-Crosshaul Network Slicing: Enabling Multi-Tenancy in Mobile Transport Networks," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 128–137, Aug 2017.
- [3] Y. Zhou, T. H. Chan, S. Ho, G. Ye, and D. Wu, "Replicating Coded Content in Crowdsourcing-Based CDN Systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 12, pp. 3492–3503, 2018.
- [4] H. Chien, Y. Lin, C. Lai, and C. Wang, "End-to-End Slicing as a Service with Computing and Communication Resource Allocation for Multi-Tenant 5G Systems," *IEEE Wireless Communications*, vol. 26, no. 5, pp. 104–112, October 2019.
- [5] J. M. A. Patricia A. Morreale, "Design and deployment," in *Software Defined Networking*. CRC Press, Taylor & Francis Group, 2015.
- [6] W. Guan, X. Wen, L. Wang, and Z. Lu, "On-Demand Cooperation Among Multiple Infrastructure Networks for Multi-Tenant Slicing: a Complex Network Perspective," *IEEE Access*, vol. 6, pp. 78 689–78 699, 2018.
- [7] "Intel Open Network Platform Server Release 1.5 Performance Test Report, SDN/NFV Solutions with Intel Open Network Platform Server, Document Revision 1.2," Intel, Tech. Rep., November 2015.
- [8] M. Erel-Özçevik and B. Canberk, "Road to 5G Reduced-Latency: A Software Defined Handover Model for eMBB Services," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8133–8144, 2019.
- [9] Y. Cui, T. Li, C. Liu, X. Wang, and M. Kühlewind, "Innovating Transport with Quic: Design Approaches and Research Challenges," *IEEE Internet Computing*, vol. 21, no. 2, pp. 72–76, 2017.
- [10] "High Performance and Low-power Processor for Digital Media Application, Chapter 44 GMAC Ethernet Interface, RK3128 Technical Reference Manual Rev 1.0," Rockchip, Tech. Rep., 2013.
- [11] J. Hemanth, X. Fernando, P. Lafata, and Z. Baig, "International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI)," *Lecture Notes on Data Engineering and Communication Technologies*, Springer, vol. 1, no. 2, pp. 161–174, 2018.
- [12] D. Gross, J. F. Shortle, J. M. Thompson, and C. M. Harris, *Fundamentals of Queueing Theory*, 4th ed. NY, USA: Wiley-Interscience, 2008.
- [13] Y. Cui, T. Li, C. Liu, X. Wang, and M. Kühlewind, "Innovating Transport with QUIC: Design Approaches and Research Challenges," *IEEE Internet Computing*, vol. 21, no. 2, pp. 72–76, Mar 2017.
- [14] P. Qian, N. Wang, and R. Tafazolli, "Achieving Robust Mobile Web Content Delivery Performance Based on Multiple Coordinated QUIC Connections," *IEEE Access*, vol. 6, pp. 11 313–11 328, 2018.
- [15] G. Szabó, S. Rác, D. Bezzera, I. Nogueira, and D. Sadok, "Media QoE enhancement With QUIC," in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, April 2016, pp. 219–220.
- [16] M. Erel, Z. Arslan, Y. Özçevik, and B. Canberk, "Grade of Service (GoS) based adaptive flow management for Software Defined Heterogeneous Networks (SDHetN)," *Computer Networks*, vol. 76, pp. 317 – 330, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1389128614004010>
- [17] M. Karsten and S. Barghi, "User-level threading: Have your cake and eat it too," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 4, no. 1, May 2020. [Online]. Available: <https://doi.org/10.1145/3379483>
- [18] "Dpaa2 user manual (compatible with mc firmware v10.24.x)," NXP Semiconductors, Tech. Rep., 08-2020.
- [19] E. Hossain and M. Hasan, "5g cellular: key enabling technologies and research challenges," *IEEE Instrumentation Measurement Magazine*, vol. 18, no. 3, pp. 11–21, 2015.