Coordinating Multi-access Edge Computing with Mobile Fronthaul for Optimizing 5G End-to-End Latency

Conference Paper · March 2018

DOI: 10.1364/OFC.2018.M4I.3

CITATIONS
5

READS
293

6 authors, including:

Wei Wang
Beijing University of Posts and Telecommunications
25 PUBLICATIONS **88** CITATIONS
SEE PROFILE

Yongli Zhao
Beijing University of Posts and Telecommunications
314 PUBLICATIONS **1,533** CITATIONS
SEE PROFILE

Massimo Tornatore
Politecnico di Milano
364 PUBLICATIONS **4,576** CITATIONS
SEE PROFILE

Han Li
Harbin Institute of Technology
219 PUBLICATIONS **1,800** CITATIONS
SEE PROFILE

Coordinating Multi-access Edge Computing with Mobile Fronthaul for Optimizing 5G End-to-End Latency

Wei Wang[1, 2], Yongli Zhao[1], Massimo Tornatore[2, 3], Han Li[4], Jie Zhang[1], Biswanath Mukherjee[2]
[1]Beijing University of Posts and Telecommunications, 100876, China; [2]University of California, Davis, CA 95616, USA;
[3]Politecnico, di Milano, Italy; [4]China Mobile Communications Corporation (CMCC), Beijing 100033, China.
{weiw, yonglizhao, lgr24}@bupt.edu.cn; massimo.tornatore@polimi.it; bmukherjee@ucdavis.edu

Abstract: In 5G, latency-sensitive traffic might be processed directly at central-offices by Multi-access Edge Computing (MEC) right after being transported through Mobile Fronthaul (MFH). We investigate how to optimize end-to-end latency by coordinating MEC with MFH.
OCIS codes: 060.4256 Networks, network optimization; 060.4510 Optical communications

1. Introduction

To support low latency services, 5G is expected to introduce significant architectural transformations. From a network perspective, Cloud Radio Access Networks (C-RAN) will connect Remote Radio Heads (RRHs) and Base Band Units (BBUs) through a low-latency network segment referred as Mobile Fronthaul (MFH). Form a service perspective, Multi-access Edge Computing (MEC) aims to reduce latency further by placing application severs at network edge.

In MFH, Passive Optical Network (PON) is a promising solution for transporting MFH traffic between a BBU pool and multiple RRHs through shared fiber connections linking the Optical Line Terminal (OLT) to multiple Optical Network Units (ONUs). To aggregate traffic from multiple ONUs to one OLT, different multiplexing options exist, e.g., Time Division Multiplexing (TDM), Time Wavelength Division Multiplexing (TWDM), and Wavelength Division Multiplexing (WDM). Among them, TDM is a popular option because of its cost efficiency and potential to reuse FTTH fibers for MFH transport [1]. However, the applicability of TDM-PON for 5G MFH is still under study due to the excessive upstream latency incurred by Dynamic Bandwidth Allocation (DBA). Recent works tried to limit upstream latency of TDM-PON in MFH: Refs. [2, 3] proposed an advanced Bandwidth Allocation (BA) scheme to schedule bandwidth prior to uplink data transmission for ONUs; Refs. [4-6] designed a burst scheduling scheme to assign each ONU certain bursts based on fast burst-mode channel tracking (e.g., via advanced burst-mode DSP). Both approaches aim to reduce DBA latency by avoiding Report and Gate messaging and real-time bandwidth computation. Also, note the recent demonstration of the use of ultra-low latency 10G PON for MFH [7].

All these solutions try to optimize TDM-PON-based MFH (TDM-PON MFH) latency from a networking point of view, i.e., by acting on DBA. They can boost the application of TDM-PON for 5G MFH, but might not be enough to support ultra-low latency 5G applications (e.g., self-driving cars). For a 5G application, even though reduction of MFH latency is important, the ultimate latency objective is measured by End-to-End (E2E) latency. E2E latency refers to the round-trip time from user sending data out to receiving corresponding response, which also includes Application (APP) layer queuing and processing latency [8]. MEC is an APP layer approach for optimizing 5G latency, and it can avoid user traffic to travel long distances (and incur high propagation latency) by processing it in MEC severs, which can provide cloud-computing capabilities at network edge [9]. MEC is expected to work also within the footprint of C-RAN, coexisting with TDM-PON MFH. In such a C-RAN, before being scheduled for APP layer processing, user traffic first experiences a certain latency caused by MFH transportation. Thus, APP layer queuing and processing latency might be affected by MFH latency in a dynamic process. We observe that coordinating APP layer task scheduling in MEC and bandwidth scheduling in TDM-PON MFH can optimize E2E latency in 5G.

In this paper, we first briefly introduce a C-RAN architecture based on MEC and TDM-PON MFH, and discuss the latency components in this architecture. We then propose two schemes to coordinate MEC with TDM-PON MFH. Simulation results show that such coordination can reduce E2E latency and E2E latency violation ratio in 5G.

2. C-RAN Architecture with TDM-PON MFH and MEC

To study the coordination of MEC and TDM-PON MFH, we need to first clarify how MEC can interwork with TDM-PON MFH in C-RAN. MEC servers can be placed at OLT (BBU) side (in central offices); this approach is compatible with the idea of C-RAN, which also intends to virtualize BBUs using standard IT resources in central offices. Another option is to place MEC servers at ONU side, but the baseband signal at ONU/RRH is hardly to be used by a MEC server directly in a practical scenario. Therefore, we take the first option, which is MEC at OLT/BBU side, as the basis for our following discussions, and the corresponding C-RAN architecture is shown in Fig. 1.

In a C-RAN, where MEC servers are co-located with BBUs, traffic from MFH can be processed directly at MEC servers, and thus its E2E latency is composed of two main parts: 1) MFH latency; 2) MEC latency. Precisely, in TDM-

PON MFH, traffic flows as shown in right part of Fig. 1: (1) User Equipment (UE) → ONU (RRH); (2) ONU → OLT (BBU); (3) OLT (BBU) → UE via ONU (RRH), and each step incurs a certain *propagation latency*. Between (1) and (2), bandwidth allocation needs to be performed since MFH upstream bandwidth is shared. According to [2-6], latency caused by Report and Gate messaging and bandwidth calculation in traditional DBA can be eliminated by either (a) advanced BA or (b) burst scheduling. Even with these approaches, traffic will still experience a certain *bandwidth scheduling latency* (waiting time for an available time slot) [8]. In a MEC server, traffic from BBU will then experience a certain *APP layer queuing and processing latency*.

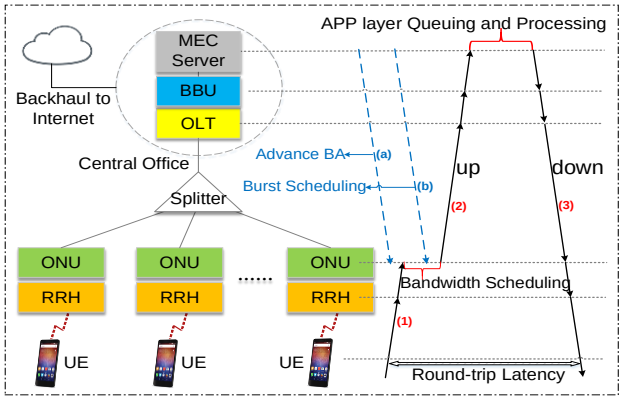

Fig. 1 C-RAN architecture and latency components.

Propagation latency is fixed as it is determined by distance. Thus, bandwidth scheduling latency and APP layer queuing and processing latency are two major components of 5G E2E latency [8], and we will take them into account for optimizing MFH latency and MEC latency in a coordinated manner.

3. Schemes for Coordinating MEC with TDM-PON MFH

Let t_{MFH} denote the bandwidth scheduling latency in MFH, and t_{MEC} denote APP layer queuing and processing latency in MEC. In TDM-PON MFH, t_{MFH} of a frame is the waiting time from arriving at ONU to getting an available time slot, and it depends on the adopted bandwidth scheduling strategy. In general, average t_{MFH} is half the OLT cycle time, during which the OLT goes through all ONUs once. In MEC servers, t_{MEC} of a task (data unit at APP layer) is caused by queuing and processing, and it is set by task scheduling, which is responsible for assigning processing resource and priority to each task. Note that processing resources in MEC servers are application-specific and finite. A task will wait in a queue when no processing resource can be assigned to it immediately. Technically, to coordinate MEC with MFH, we can jointly manage: 1) bandwidth scheduling in MFH; and 2) task scheduling in MEC. We propose two schemes to illustrate how coordination between MEC and TDM-PON MFH can be implemented, and they are: 1) MFH Latency-driven Task Scheduling (ML-driven TS); 2) Task Queue-aware Bandwidth Scheduling (TQ-aware BS). As the two names suggest, in ML-driven TS, task scheduling is performed by taking into account the MFH bandwidth scheduling latency, while in TQ-aware BS, bandwidth scheduling is performed by taking into account the queueing status of previous tasks in MEC. To discuss cross-layer scheduling problems in a generic manner, we will take the term "frame" as the basic data unit for both bandwidth scheduling and task scheduling.

ML-driven TS: After passing through MFH upstream, each frame experiences a t_{MFH}, which can be measured at OLT (BBU). t_{MFH} may be higher or lower than an expected/average MFH latency, as it is associated with ever-changing traffic conditions. To meet APP's E2E latency requirements T, the knowledge of t_{MFH} allows us to decide the residual time left for APP layer queueing and processing. If t_{MFH} of a frame is higher than expected, we can compensate t_{MFH} by processing this frame with higher priority and more processing resource. Else, if t_{MFH} is lower than expected, we can leverage this knowledge and process other urgent frames with higher priority and more processing resource, as long as t_{MEC} is less than $T - t_{MFH}$. Frame #2 in Fig. 2(a) is an example, it arrives at ONU-2 at time t_0, but 4 time slots from t_0 were already assigned to ONU-1. Thus, frame #2 has to be delayed 4 time slots and will arrive at OLT (BBU) at t_2. According to pending queue at t_2, it is expected to be processed at t_4. Let us say $t_4 - t_0$ violates its E2E latency requirement since MFH consumes too much time. In this case, ML-driven TS will process this frame with higher priority or more processing resource to compensate for t_{MFH} and meet E2E latency.

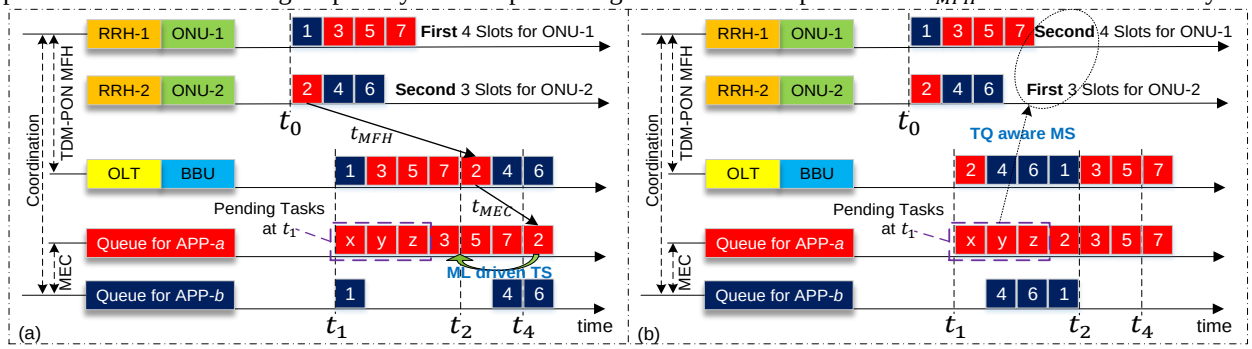

Fig. 2 (a) ML-driven TS, (b) TQ-aware BS.

TQ-aware BS: current pending tasks for an APP may affect t_{MEC} of upcoming tasks from the same APP, hence the knowledge of APP layer queuing status is a valuable indicator for future MFH bandwidth scheduling. The logic of TQ-aware BS is more complex than ML-driven TS, as it is about future operations. When the queue for an *APP-a* is longer (in terms of waiting time) than the queue of another *APP-b*, it is useless to transmit frames for *APP-a* through MFH immediately, as these frames will wait anyway at the MEC server. On the contrary, frames for *APP-b* can be transmitted quicker in this case. In practice, at ONU side of a TDM-PON MFH, we cannot assign bandwidth per APP. But we can adapt this case as "assign higher priority and more bandwidth for the ONU, whose buffer has more frames for the APP with a shorter queue in MEC". In Fig. 2(b), with the knowledge that queue of *APP-a* is longer than that of *APP-b*, OLT schedules ONU-2, which has more pending frames for *APP-b*, prior to ONU-1. Result of "TQ-aware BS" is that frames of *APP-a* are processed at almost the same time, but frames of *APP-b* are processed much earlier (compared with Fig. 2(a)). Another practical problem is that it is not easy to know which ONU has upcoming frames for which APP, as ONU is not aware of APPs. Fortunately, this is not a major problem as such information can be predicted at BBU/MEC side using data analysis (e.g., Machine Learning).

4. Illustrative Numerical Results

We evaluated the idea of coordinating MEC with TDM-PON MFH through simulation. In our setting, MFH has 5 ONUs (RRHs) and 1 OLT (BBU), and we have 1 MEC server, supporting 3 APPs. Time dimension is simulated by a series of contiguous time slots; and a set of frames at each time slot for each ONU is randomly generated as input. Each frame with valid payload belongs to one of the 3 APPs with equal probability, and requires a certain number of processing operations. To eliminate the influences of factors that are not related to MEC-MFH coordination, we consider a fixed ONU burst time (4 time slots). To conserve space, other setting details are not presented here.

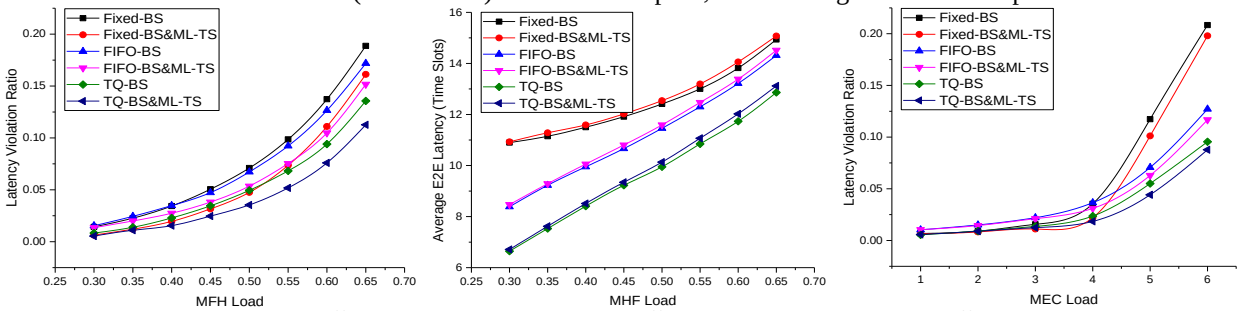

Fig. 3 (a) LVR under different MFH loads, (b) AEL under different MHF load, (c) LVR under different MEC load.

TQ-aware BS is compared with two benchmarks: Fixed Bandwidth Scheduling (Fixed-BS) and First-In-First-Out Bandwidth Scheduling (FIFO-BS). Fixed-BS assigns the same time slots for each ONU in every cycle, and FIFO-BS assigns next available burst duration to the ONU, whose pending frames arrived the earliest. In addition, ML-driven TS is applied to all BS schemes for further comparison. Figs. 3(a) and 3(b) show the Latency Violation Ratio (LVR) and Average E2E Latency (AEL) of all schemes under various MFH Load -- the ratio between the MFH frames, which have valid payloads, and all frames. First, compared with Fixed-BS and FIFO-BS, TQ-aware BS (TQ-BS) has the lowest LVR and AEL, indicating that APP layer information can help to schedule MFH bandwidth more efficiently. Second, ML-driven TS further reduces LVR for each BS scheme, as compensating MFH latency at MEC servers can save some frames, which are going to violate E2E latency. However, ML-driven TS slightly increases AEL of each BS scheme. The reason is that urgent frames, which need latency compensation immediately, will interrupt some other tasks. Fig. 3(c) shows LVR under different MEC load (in terms of average time slots for processing a task), which is positively correlated with APP layer latency. TQ-aware BS and ML-driven TS can also reduce LVR in this case, and their advantages are more significant when APP layer latency accounts for more in E2E latency.

5. Conclusion

This paper studied latency issues in C-RAN for 5G and proposed two schemes to coordinate bandwidth scheduling in TDM-PON MFH and application layer task scheduling in MEC. Results verified that such coordination could further optimize E2E latency in 5G. (This work is supported by NSFC (61571058), NSTMP (2017ZX03001016) and NSF (1716945)).

6. References

[1] B. Skubic, et al., JOCN, vol. 9, no. 9, pp. D10-D18, 2017.
[2] T. Tashiro, et al., OFC, Tu3F.3, 2014.
[3] J. Kani, et al., JLT, vol. 35, no. 3, pp. 527-534, 2017.
[4] X. Liu, et al., JOCN, vol. 8, no. 12, pp. B70-B79, 2016.
[5] S. Zhou, et al., OFC, Th3A.3, 2017.
[6] X. Liu, et al., OFC, W4C.4, 2017.
[7] "Nokia Bell Labs News", [online] Available: https://www.nokia.com/en_int/news/releases/2017/06/20/, 2017.
[8] R. Bonk, et al., "Latency Challenges for 25/50/100G EPON", [online] Available: http://www.ieee802.org/3/ca/public/meeting_archive/2017/09/powell_3ca_1a_0917.pdf, 2017.
[9] Y. Hu, et al., ETSI white paper on MEC, 2015.