Offloading Selection with Unequal Timeslot in Mobile Edge Computing

Naotaka Hirayama, Takuya Kobayashi, and Koichi Adachi
Advanced Wireless & Communication Research Center (AWCC),
The University of Electro-Communications, Tokyo, Japan
E-mail: {hirayama, t.kobayashi, adachi}@awcc.uec.ac.jp

Abstract—The growth of Internet-of-Things increases the number of mobile devices and computationally heavy applications. However, mobile devices typically have insufficient computation capability and battery capacity because of their small physical size. Mobile-edge computing (MEC) is a practical approach to solve these problems, where mobile devices offload their tasks to the MEC server instead of computing them locally. To avoid packet collision between the different packet lengths, we propose to split the transmission frame into multiple timeslots with unequal time lengths. Selecting a transmission timeslot based on the packet length can reduce the number of packet collisions caused by a long-length packet. Simulation results show that the proposed scheme can improve the transmission success rate with fewer retransmissions and the processing delay of tasks compared to the ALOHA-based protocol.

I. Introduction

Recently, the growth of Internet-of-Things (IoT) stimulates the increasing number of mobile devices and heavy applications [1] [2]. However, it is difficult for mobile devices to compute computationally heavy applications due to their limited computational capability and battery capacity. Mobile-edge computing (MEC) has been expected to solve these problems [3]. Mobile devices offload their heavy tasks to the MEC server instead of computing them locally. In the MEC system, the physical distance between mobile devices and the server is closer than in the mobile cloud computing (MCC) system [4]. Since, for the MEC system, multiple mobile devices need to share the MEC server and radio resource for task offloading, it is necessary to allocate radio resources to them appropriately. Some conventional works considered minimizing energy consumption of mobile devices [5] and minimizing processing delay of tasks [6]. In general, there is a trade-off between energy consumption and processing delay. A system that can arbitrarily decide whether to prioritize energy consumption or processing delay is considered in [7]. However, most existing works adopt centralized control at the access point (AP) to allocate resources to mobile devices. The centralized control incurs overheads for exchanging control information between mobile devices and an AP. On the other hand, the decentralized transmission control, e.g., Pure ALOHA [8], Slotted ALOHA [9], and carrier sense multiple-access (CSMA) [10], can reduce overheads but cannot completely avoid packet collisions. One of the access methods to reduce the number of packet collisions is dynamic frame length ALOHA [11]. The dynamic frame length ALOHA scheme adds a dynamic frame structure to the Slotted ALOHA scheme. The frame consists of multiple timeslots, and the frame length dynamically varies according to the expected value of the backlog. The backlog represents the number of mobile devices that are ready to transmit a packet. This scheme can improve throughput by 16 % compared to the Pure ALOHA scheme. However, this scheme requires the information exchange about backlog between mobile devices and an AP.

When multiple mobile devices offload their tasks to the MEC server, packet collisions may happen. The packet with a long time length may collide with multiple short time length packets, which significantly deteriorates the overall system performance. Thus it is effective to avoid such packet collisions. This paper proposes splitting the transmission frame into multiple timeslots with different timeslot lengths based on that observation. Each mobile device estimates the packet length and selects one of the unequal length timeslots. Selecting transmission timeslots based on the packet length can lower the packet collision probability caused by long-length packets. The system can reduce the overheads for exchanging control information between mobile devices and an AP. The computer simulation results show that the proposed scheme can improve the transmission success rate with fewer retransmissions than the ALOHA-based protocol. In addition, it can reduce the processing delay of tasks.

The remainder of this paper is organized as follows. In Section Ⅱ, we describe the system model. We present the proposed scheme in Section Ⅲ and the numerical results in Section Ⅳ. Finally, we conclude the article in Section V.

II. System Model

This paper considers a multi-user MEC system as shown in Fig. 1. An AP equipped with a MEC server is placed at the center and K mobile devices (set $\mathcal{K} = \{1, ..., K\}$) are placed randomly and uniformly within the communication area of the AP.

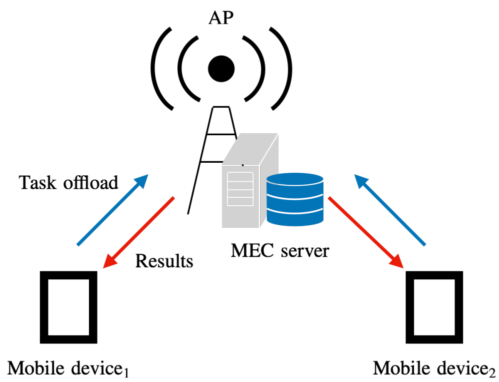

Fig. 1. System model

A. Task Generation Model

We assume that computational tasks are generated independently at each mobile device according to Poisson process [12]. Mobile device $k \in \mathcal{K}$ generates task i at

$$t_{k,i} = \begin{cases} -\log\left(\frac{W}{\lambda}\right) & (i=0) \\ t_{k,i-1} - \log\left(\frac{W}{\lambda}\right) & (i \neq 0) \end{cases}, \quad (1)$$

where λ denotes the expected value of Poisson process [/sec] and W denotes the random variable following to the uniform distribution $(0, 1)$.

B. Channel Model

The path loss is given by [13]

$$L(d_k, f_{\mathrm{c}}) = 10a\log_{10}(d_k) + b + 10c\log_{10}(f_{\mathrm{c}}), \quad (2)$$

where d_k denotes the 3D direct distance between mobile device k and the AP [m], f_{c} denotes the carrier frequency [GHz], a denotes the coefficient associated with distance, b denotes the offset value, and c denotes the coefficient associated with frequency. The transmission rate of mobile device k [bits/sec] is given by

$$R_k = \begin{cases} BC_k & (C_k \leq C_{\max}) \\ BC_{\max} & (\text{otherwise}) \end{cases}, \quad (3)$$

where B denotes the bandwidth [Hz], C_k denotes the channel capacity of mobile device k between the AP [bits/sec/Hz] given by $\log_2(1+\gamma_k)$, $C_{\max}$ denotes the maximum channel capacity [bits/sec/Hz], and γ_k denotes the signal-to-noise power ratio (SNR) of mobile device k. Here we assume that the mobile devices can adjust the transmit power ideally so that the channel capacity will not exceed $C_{\max}$ with high SNR.

C. Local Processing Delay

The processing delay of mobile device k can be calculated as

$$t_k^{\mathrm{loc}} = \frac{A_k X}{f_k}, \quad (4)$$

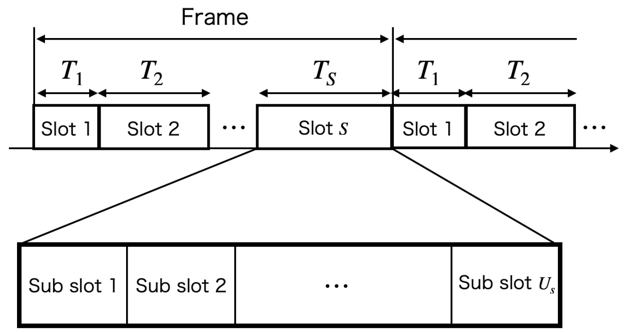

Fig. 2. Frame structure

where A_k denotes the input data size of computational task [bits], X denotes the number of CPU cycles required for processing single bit [CPU cycles/bit] called as workload, and f_k denotes the CPU cycle frequency of mobile device k [Hz].

The energy consumption of mobile device k can be calculated by [14]

$$E_k^{\mathrm{loc}} = \kappa A_k X f_k^2, \quad (5)$$

where κ denotes the CPU effective capacitance coefficient.

D. Offloading Delay

The total delay due to offloading consists of task transmission to the MEC server, task processing at the MEC server, and reception of the computation result from the MEC server.

The transmission delay required for offloading is given by

$$t_k^{\mathrm{tr}} = \frac{A_k}{R_k}. \quad (6)$$

Meanwhile, the energy consumption required for offloading of mobile device k is given by

$$E_k^{\mathrm{tr}} = P_k t_k^{\mathrm{tr}}, \quad (7)$$

where P_k denotes the transmit power of mobile device $k \in \mathcal{K}$ [dBm].

The processing delay at the MEC server can be calculated as

$$t_k^{\mathrm{MEC}} = \frac{A_k X}{f_{\mathrm{s}}}, \quad (8)$$

where f_{s} denotes the CPU cycle frequency of MEC server [Hz].

This paper assumes that the computation result size is small, and thus, we ignore the delay and energy consumption for receiving results.

III. Proposed Method

A. Offloading Scheme

This subsection describes the proposed offloading scheme. If packets with different time lengths are transmitted randomly, the long-length packets collide with multiple short-length packets with high probability. Hence, we consider an offloading scheme that selects the transmission timeslot based on packet length. We introduce the frame structure for uplink as shown Fig. 2. In each frame, there are unequal S main timeslots (set $\mathcal{S} = \{1, ..., S\}$) whose length satisfy $T_1 < T_2 <, ..., < T_S$ and each timeslot has U_s equal sub timeslots whose length are T_s^{sub}. In the following, we describe the procedure for processing tasks and how to select the transmission timeslot.

1) Timeslot selection: Mobile device k selects the timeslot according to task size $A_{k,i}$ and SNR γ_k. Thus, each mobile device selects timeslot O_k according to

$$O_k = \begin{cases} 1 & \text{if } 0 < t_k^{\mathrm{tr}} \le t_1^{\mathrm{th}} \\ s & \text{if } t_{s-1}^{\mathrm{th}} < t_k^{\mathrm{tr}} \le t_s^{\mathrm{th}} \\ S & \text{if } t_k^{\mathrm{tr}} > t_{S-1}^{\mathrm{th}} \end{cases}, \tag{9}$$

where t_k^{tr} is given by (6) and t_s^{th} denotes the threshold value of selecting the timeslot s [sec].

2) Sub timeslot selection and task transmission: In the 1st to $(S-1)$th timeslot, mobile devices select the timeslot so that the packet length is shorter than the sub timeslot length. Then, mobile devices randomly select a sub timeslot from U_s sub timeslots to start transmission. The mobile device randomly selects a sub timeslot so that transmission is completed within the timeslot if its packet length is shorter than the timeslot length. Otherwise, the mobile device selects the 1st sub timeslot in the Sth timeslot.

3) Retransmission: A mobile device judges whether or not task transmission was successful based on the acknowledgment (ACK) signal from an AP. If the mobile device does not receive the ACK signal, it retransmits the task at a random sub-timeslot in the same timeslot of the next frame. The retransmissions are repeated at most M times.

4) Computation execution: If the task transmission is successful, the MEC server processes the task and returns it to each mobile device. If the transmission is not successful after M retransmissions, the mobile device processes the task locally.

B. Theoretical Performance Analysis

In this subsection, we introduce the theoretical performance of the proposed scheme. The packet delivery rate (PDR), p^{PDR}, and the transmission success rate, p^{suc}, are given by

$$\begin{cases} p^{\mathrm{PDR}} & = \frac{Y}{N} \\ p^{\mathrm{suc}} & = \frac{Y}{G} \end{cases}, \tag{10}$$

where Y denotes the number of tasks successfully received by the AP, N denotes the number of tasks transmitted by the mobile devices, and G denotes the number of generated tasks. The average processing delay and the average energy consumption can be calculated from the transmission success rate.

1) Without retransmission: The expected value of the number of tasks per sub timeslot in the timeslot s is given by

$$\lambda_s = \frac{\lambda K}{S U_s} T_{\mathrm{frame}}, \tag{11}$$

where $T_{\mathrm{frame}} = \Sigma_{s=1}^{S} T_s$ denotes the frame length. The probability that the number of mobile devices that transmit at a sub timeslot in timeslot s being z follows Poisson distributed with mean λ_s, which is given by

$$p_{s,z} = \frac{\exp(-\lambda_s)\lambda_s^z}{z!}. \tag{12}$$

Packet collision happens when multiple mobile devices transmit in the same sub-timeslot. Thus PDR and transmission success rate in timeslot s without retranmission written as p_s^{PDR} and $p_{s,0}^{\mathrm{suc}}$ are calculated as

$$p_s^{\mathrm{PDR}} = p_{s,0}^{\mathrm{suc}} = \frac{p_{s,1}}{\lambda_s}. \tag{13}$$

Without retransmission, the PDR and the transmission success rate are equal. For each timeslot, the average processing delay and the average energy consumption can be calculated using transmission success rate p_s^{suc} as

$$\begin{cases} t_{s,0}^{\mathrm{ave}} & = (T_{\mathrm{frame}}/2 + t_s^{\mathrm{MEC}}) p_{s,0}^{\mathrm{suc}} + (T_{\mathrm{frame}}/2 + t_s^{\mathrm{loc}})(1 - p_{s,0}^{\mathrm{suc}}) \\ E_{s,0}^{\mathrm{ave}} & = P_{\max} t_s^{\mathrm{tr}} \times p_{s,0}^{\mathrm{suc}} + E_s^{\mathrm{loc}}(1 - p_{s,0}^{\mathrm{suc}}) \end{cases}, \tag{14}$$

where t_s^{MEC} denotes the average delay to calculate the tasks in timeslot s by the MEC server [sec], t_s^{loc} denotes the average delay to calculate the tasks in timeslot s by the mobile device [sec], $P_{\max}$ denotes the maximum transmit power for transmission [dBm], t_s^{tr} denotes the average task transmission delay in timeslot s [sec], and E_s^{loc} denotes the average energy consumption to calculate the tasks in timeslot s [J]. Therefore, the overall transmission success rate, the average processing delay and the average energy consumption are as follows

$$\begin{cases} p_0^{\mathrm{suc}} & = \frac{1}{S}\sum_{s=1}^{S} p_{s,0}^{\mathrm{suc}} \\ t_0^{\mathrm{ave}} & = \frac{1}{S}\sum_{s=1}^{S} t_{s,0}^{\mathrm{ave}} \\ E_0^{\mathrm{ave}} & = \frac{1}{S}\sum_{s=1}^{S} E_{s,0}^{\mathrm{ave}} \end{cases}. \tag{15}$$

2) With retransmissions: The derivation of the transmission success rate considering retransmissions is very complex when mobile devices are not in a state of transmitting task continuously. Since it is difficult to derive a convergence value of transmission success rate, we derive an approximate. Specifically, we consider the transmission success rate from the 1st to $(M+1)$th frame. To express p^{suc} with retransmissions, we use a posteriori expected value of the number of mobile devices in the case of packet collisions, which is given by

$$g_s = \sum_{z=2}^{K} z \cdot \frac{p_{s,z}}{1 - p_{s,0} - p_{s,1}}. \tag{16}$$

The probability that task transmission is successful in the 1st frame, $p^{\mathrm{f}}_{s,1}$, is given by

$$p^{\mathrm{f}}_{s,1} = \frac{p_{s,1}}{\lambda_s}. \tag{17}$$

Then, the probability that the transmission of a task failed m times, but succeeds in the $(m+1)$th frame, $p^{\mathrm{f}}_{s,m+1}$, is given by

$$p^{\mathrm{f}}_{s,m+1} = \sum_{u=1}^{U_s} p_{s,0} \times_{U_s} \mathrm{C}_u \left(\sum_{n=1}^{m} p^{\mathrm{f}}_{s,n}\right)^{U_s-u} \left(1 - \sum_{n=1}^{m} p^{\mathrm{f}}_{s,n}\right)^{u} \times \frac{1}{U_s}\left(\frac{U_s-1}{U_s}\right)^{g_s \times u-1}, \tag{18}$$

where u denotes the number of sub timeslots happening packet collisions in timeslot s of the mth frame. Therefore, transmission success rate with the maximum number of retransmissions M, $p^{\mathrm{suc}}_{s,M}$, is given by

$$p^{\mathrm{suc}}_{s,M} = \sum_{m=1}^{M+1} p^{\mathrm{f}}_{s,m}. \tag{19}$$

For each timeslot, the average processing delay and the average energy consumption with the maximum number of retransmissions M are given by

$$\begin{cases} t^{\mathrm{ave}}_{s,M} = (T_{\mathrm{frame}}/2 + t^{\mathrm{MEC}}_s)p^{\mathrm{suc}}_{s,M} \\ \qquad + (T_{\mathrm{frame}}/2 + t^{\mathrm{loc}}_s)(1 - p^{\mathrm{suc}}_{s,M}) \\ E^{\mathrm{ave}}_{s,M} = P_{\max} t^{\mathrm{tr}}_s \times p^{\mathrm{suc}}_{s,M} + E^{\mathrm{loc}}_s(1 - p^{\mathrm{suc}}_{s,M}) \end{cases}. \tag{20}$$

Then, the overall transmission success rate, the average processing delay, and the average energy consumption are given by

$$\begin{cases} p^{\mathrm{suc}}_{M} = \frac{1}{S}\sum_{s=1}^{S} p^{\mathrm{suc}}_{s,M} \\ t^{\mathrm{ave}}_{M} = \frac{1}{S}\sum_{s=1}^{S} t^{\mathrm{ave}}_{s,M} \\ E^{\mathrm{ave}}_{M} = \frac{1}{S}\sum_{s=1}^{S} E^{\mathrm{ave}}_{s,M} \end{cases}. \tag{21}$$

IV. Numerical Results

This section provides the computer simulation results regarding the task transmission success rate, the average energy consumption, and the average processing delay. The theoretical value obtained by (21) and computer simulation results are included.

A. Simulation Parameters

The simulation parameters are given in Table 1 [15] [16]. The maximum channel capacity is $C_{\max} = 4$ [bit/sec/Hz]. The maximum transmit power is $P_{\max} = 13$ [dBm]. The number of timeslots is $S = 3$. The number of sub timeslots is $U_s = 1$ (2) with the maximum number of retransmissions $M = 0$ ($M > 0$). These values are set so that the average processing time obtained by (21) is minimized. Each plot is obtained by averaging over 10000 simulation runs.

TABLE I
Simulation Parameters

Prameters	Value
Simulation time	10 [min]
The number of trials	10000
Simulation area	50×50 [m^2]
The number of mobile devices K	30
Task size A_k	$[1, 10]$ [Mbits]
Workload X	10^3 [CPU cycles/bit]
CPU capacitance coefficient κ	10^{-28}
Task generating rate λ	0.1 [/sec]
CPU frequency of mobile devices f_k	1 [GHz]
CPU frequency of MEC f_s	20 [GHz]
Noise power spectrum density N_0	-174 [dBm/Hz]
Carrier frequency f_c	2.4 [GHz]
Bandwidth B	20 [MHz]
The number of timeslot S	3
The number of sub timeslot U_s	$U_s \in \{1, 2\}$
Sub timeslot length T^{sub}_s ($s \in \mathcal{S}$)	$0.05 + 0.0375 \times (s+1)$ [sec]

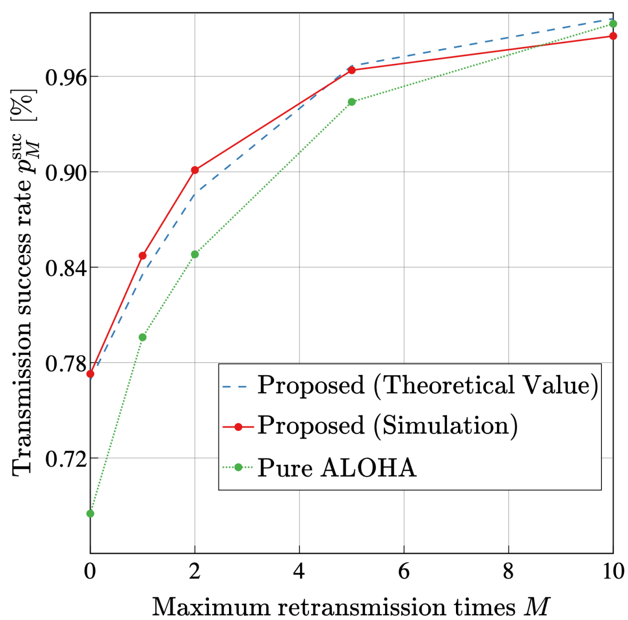

Fig. 3. Transmission success rate

B. Simulation Results

1) Transmission Success Rate: Fig. 3 shows the task transmission success rate as a function of the maximum number of retransmissions. The proposed scheme can increase the transmission success rate by approximately 12 % compared to the Pure ALOHA scheme without retransmission because the proposed scheme reduces the number of packet collisions. It can be seen from the figure that the transmission success rate is over 0.98 for both the proposed scheme and the Pure ALOHA scheme when the maximum number of retransmissions is $M = 10$. Since the Pure ALOHA scheme exponentially increases the maximum backoff time as the number of retransmissions

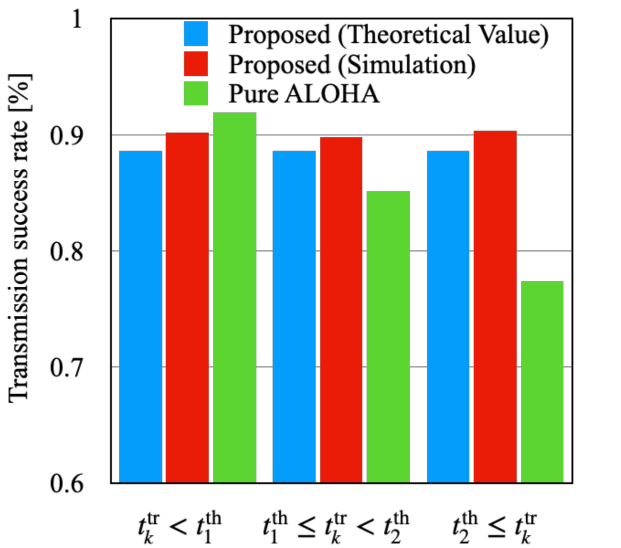

Fig. 4. Transmission success rate for each transmission time ($M = 2$)

increases, it can avoid packet collision at the cost of increased delay. On the other hand, the proposed scheme fixes the backoff time to retransmit the task in the same timeslot in the next frame. Therefore, the transmission success rate of the proposed scheme deteriorates when the maximum number of retransmissions is $M = 10$. The computer simulation results of the proposed scheme agree well with the theoretical values. As explained in Section III, it is very complex to calculate transmission success rate with retransmissions exactly. Then, we appropriate the theoretical value of transmission success rate as the transmission success rate from the 1st to $(M+1)$th frame. Thereby, the theoretical values do not equal the simulation results completely.

2) Transmission Success Rate for Different Packet Length: Fig. 4 shows the transmission success rate for each packet length with the maximum number of retransmissions $M = 2$. In the system, most of the devices can transmit tasks with the maximum channel capacity. Therefore, the number of mobile devices that select each timeslot is equal when the task size follows a uniform distribution. It is found that the Pure ALOHA scheme decreases the transmission success rate for long-length packets. On the other hand, the proposed scheme can keep the transmission success rate constant regardless of the packet length.

3) Average Energy Consumption: Fig. 5 shows the average energy consumption against the maximum number of retransmissions. The proposed scheme can reduce the average energy consumption by approximately 34 % compared to the Pure ALOHA scheme without retransmission. It can be seen that the average energy consumption decreases for both the proposed scheme and

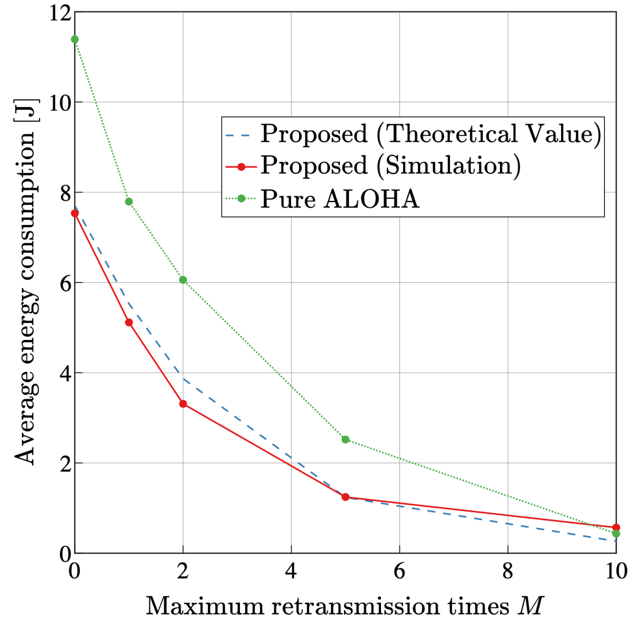

Fig. 5. Average energy consumption

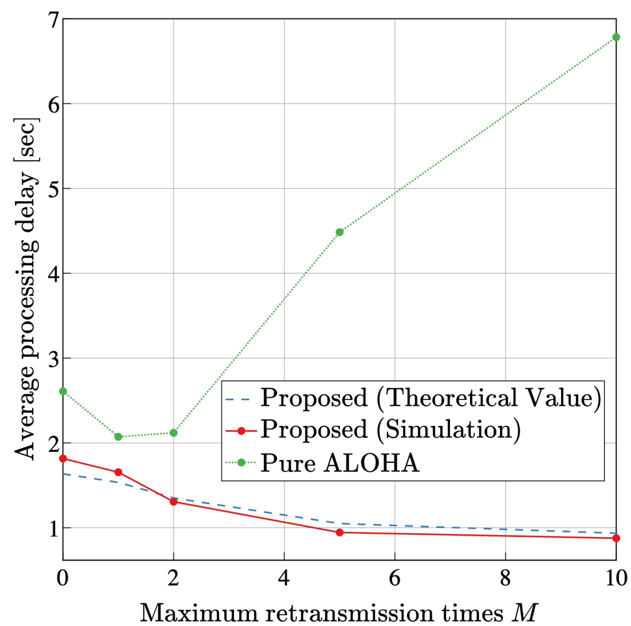

Fig. 6. Average processing delay

the Pure ALOHA scheme as the maximum number of retransmissions increases. Local execution requires larger average energy consumption than task offloading, and thus, the energy consumption decreases with increasing transmission success rate.

4) Average Processing Delay: Fig. 6 shows the average processing delay against the maximum number of retransmissions. The Pure ALOHA scheme increases

the maximum backoff time exponentially as the number of retransmissions increases. Thus, the Pure ALOHA scheme significantly increases the processing delay as the maximum number of retransmissions increases. On the other hand, the proposed scheme can reduce the processing delay even if the maximum number of retransmission becomes large. For example, when the maximum number of retransmissions is 2, the proposed scheme can reduce the processing delay by approximately 38 %. In addition, it can be seen from Fig. 5 and Fig. 6 that when the maximum number of retransmissions is 10, the energy consumption of the proposed scheme is almost the same as that of the Pure ALOHA scheme. However, the processing delay of the proposed scheme is much lower than that of the Pure ALOHA scheme.

V. Conclusion

This paper proposed splitting the transmission time frame into multiple timeslots with unequal time lengths to avoid packet collisions between different time length packets. Mobile devices select the transmission timeslot according to their packet length. The simulation results showed that the transmission success rate for long-length packets could be improved. Thus, a larger number of computationally intensive tasks can be executed by the MEC server. Therefore the processing delay of tasks can be reduced. In addition, the theoretical values ensure that the simulation results are valid.

References

[1] A. A. Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications,” IEEE Commun. Surveys & Tuts., vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.

[2] D. Warren, et al., “Understanding 5G: Perspectives on Future Technological Advancements in Mobile,” GSMA Intell., London, U.K., Dec. 2014.

[3] Y. Mao, C. You, J. Zhang, K. Huang, and K. Letaief, “A Survey on Mobile Edge Computing: The Communication Perspective,” IEEE Commun. Survey & Tuts., vol. 19, no. 4, pp. 2322-2358, Aug. 2017.

[4] M. Armbrust, et al., “Above The Clouds: A Berkeley View of Cloud Computing,” Dept. Elect. Eng. Comput. Sci., Univ. California at Berkeley, Berkeley, CA, USA, Feb. 2012.

[5] C. You, Y. Zeng, R. Zhang, and K. Huang, “Resource Management for Asynchronous Mobile-Edge Computation Offloading,” in Proc IEEE ICC Workshops, May 2018.

[6] Z. Ding, D. W. K. Ng, R. Schober, and H. V. Poor, “Delay Minimization for NOMA-MEC Offloading,” IEEE Signal Processing Letters, vol. 25, no. 12, pp. 1875-1879, Dec. 2018

[7] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, “Power-Delay Tradeoff in Multi-User Mobile-Edge Computing Systems,” in Proc. IEEE Global Commun. Conf. (GLOBECOM), pp. 1-6, 2016

[8] N. Abramson, “The ALOHA System - Another alternative for computer communications,” in Proc 1970 Fall Joint Comput. Conf., AFIPS Conf., vol. 37. Montvale, N. J.: AFIPS Press, 1970, pp. 281-285.

[9] J. Arnbak and W. van Blitterswijk, “Capacity of Slotted ALOHA in Rayleigh-Fading Channels,” IEEE J. Sel. Areas Commun., vol. 5, no. 2, pp. 261-269, Feb. 1987.

[10] L. Kleinrock, and F. Tobagi, “Packet Switching in Radio Channels: Part I - Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics,” IEEE Trans. Commun., vol. 23, no. 12, pp. 1400-1416, Dec. 1975.

[11] F. Schoute, “Dynamic Frame Length ALOHA,” IEEE Trans. Commun., vol. 31, no. 4, pp. 565-568, Apr. 1983.

[12] A. F. Molisch, Wireless Communications Second Edition, United States of America: Wiley-IEEE Press, 2011.

[13] ITU-R, “Propagation Data and Prediction Methods for the Planning of Short-Range Outdoor Radiocommunication Systems and Radio Local Area Networks in the Frequency Range 300 MHz to 100 GHz,” Aug. 2019.

[14] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, “Energy-Optimal Mobile Cloud Computing Under Stochastic Wireless Channel,” IEEE Trans. Wireless Commun., vol. 12, no. 9, pp. 4569–4581, Sept. 2013.

[15] J. -B. Wang, H. Yang, M. Cheng, J. -Y. Wang, M. Lin, and J. Wang, “Joint Optimization of Offloading and Resources Allocation in Secure Mobile Edge Computing Systems,” IEEE Trans. Veh. Technol., vol. 69, no. 8, pp. 8843-8854, Aug. 2020.

[16] T. X. Tran and D. Pompili, “Joint Task Offloading and Resource Allocation for Multi-Server Mobile-Edge Computing Networks,” IEEE Trans. Veh. Technol., vol. 68, no. 1, pp. 856-868, Jan. 2019.