Measurement-based IoT Server Selection for Mobile Edge Computing

Nuntanut Bhooanusas and Sok-Ian Sou

***Abstract*—The exponential growth in a number of the smart devices connected to the Internet of Things (IoT) in recent years has resulted in a massive increase in the volume of user generated content. Edge computing has been proposed to reduce the communication latency and improve the security of the user data by moving the computation and storage functions of these services closer to the users. However, the transmission delay between the devices and the edge server has a critical effect on the performance of the data offloading service since most IoT applications are delay-intolerant. Accordingly, the present study presents the use of measurement-based selection to determine the server with the shortest delay. Overall, the results show that irrespective of the scale of the networks, this kind of method helps to select the server with a reasonable cost.**

***Index Terms*—IoT, fog-based storage offloading, edge computing, RTT, delay.**

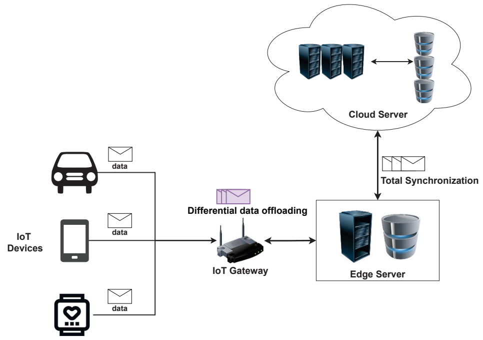


Fig. 1: A brief concept of storage offloading in IoT applications.

I. Introduction

The Internet of Things (IoT) has dramatically expanded in recent years and includes many potential applications such as smart home security system, remote health care, inventory tracking, autonomous vehicles, and so on. According to a recent Cisco report, the number of devices connected to the IoT is likely to exceed 500 billion by 2025, where these devices will include not only computers and smartphones, but potentially any IoT-enabled physical object such as electronic home appliances and sensors [1].

The rapid increase in the number of devices connected to the IoT has led to an exponential growth in the volume of user generated content. This poses a significant data storage challenge since most IoT devices have only limited storage resources [2], [3]. To ease this problem, several cloud computing companies have developed centralized cloud-based storage services (e.g., Apple iCloud, Dropbox and Google Drive) such that IoT devices can offload some of their computational tasks and even storage management to the cloud; thereby resulting in significant local memory space savings. However, much of the data sensed by IoT sensors is sensitive. For example, home security system data is highly attractive to intruders and other cyber criminals [4], if not properly secured. Hence, the storage and protection of sensitive data poses a significant challenge to the future development of the IoT.

Furthermore, storage offloading also raises various connection-based issues, such as traffic congestion, lengthy delays, and high energy consumption. Mobile edge computing (MEC) is an emergent architecture where cloud computing services are extended to the edge of networks [5], [6]. Fig. 1 presents a schematic illustration of the IoT storage offloading paradigm with edge server, where the computational and storage task of the user is distributed to the edge servers via an IoT gateway. The connection speed between the IoT gateway and the edge servers has a critical effect on the performance of the data offloading service. Accordingly, many applications provide the method to measure the delay. For example, in [12], Voice over IP (VoIP) communications were improved by means of QoS-enabled Access Points (APs) in which traffic streams with different priorities were identified based on an inspection of the Round-Trip-Time (RTT) values determined from the Real-time Transport Control Protocol (RTCP) packet header information.

Besides of the RTT, many studies use the Received Signal Strength (RSS) as the selection criterion for investigating the best WiFi AP. In [8], if mean RSS value and loss rate of a particular AP fell below pre-defined threshold, that AP would not be selected for offloading purposes. Shafi et al. [7] extended the AP selection process to consider not only the RSS value, but also the WiFi bandwidth availability. Hence, using the measurement-based selection is much better than the RSS-based one in case of considering a transmission speed.

In order to understand the performance of the delay measurement-based selection, for instance, how many times the delay should be measured in order to claim that it is the “best” server with the fastest path, we use simulation to assess the performance in terms of the precise selection probability and the selection delay. We assume that the storage server is chosen as the server with the fastest response time over K measurements; the measurment can be provided by the Resquest/Response round trip time provided in some application headers (such as RCTP mentioned before) or simply using the measurement delay obtained via the Internet Control Message Protocol (ICMP) ping commands. The delay measurement is used to explore the total waiting time for the transmission delay and the processing time in the server.

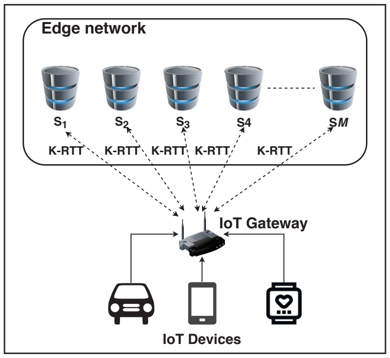


Fig. 2: System model for the storage server selection in IoT offloading.

The remainder of this paper is organized as follows. Section II describes previous related work in the field. Section III presents the simulation model. Section IV describes the measurement-based server selection. Section V further investigates the performance of the measurement-based selection via several numerical examples. Finally, Section VI provides some brief concluding remarks.

II. Background and Related Work

Recently, numereous studies on storage offloading stated the problems of reducing delay between an end-user and the edge server and investigating the edge server with a suitable network condition. Elgazar et al. [9] presented an intelligent offloading system designated as EdgeStore to provide the best edge sever to users in accordance with the place of network environment namely rural, suburban and metropolitan.

Moreover, Edge computing is also beneficial in improving the energy consumption. Chang et al. [10] proposed a joint computation offloading and radio resource allocation algorithm to minimize the system cost; Xu et al. [11] studied the edge server placement for social media offloading in Internet of Vehicles (IoV). Such proposed methods all employed the concept of resource sharing among servers to minimize the energy consumption.

In practice, the performance of data offloading is critically dependent on the network conditions. Accordingly, some studies tried to select appropriate WiFi APs to establish a seamless connection in MEC environments. Fakhfakh and Hamouda [13] developed a distributed Q-learning algorithm to facilitate the WiFi offloading decision based on a joint consideration of the AP load, the handover duration, the offered gain, and the signal-to-interference-plus-noise ratio. Flaithearta et al. [12] selected the best WiFi AP for a VoIP application by considering the RTT value between an end-user and each candidate AP.

III. System model

Fig. 2 shows the system model of the storage offloading scenario, where considering a set of M storage server are deployed in the edge network, denoted by $\mathcal{S} = \{S_1, S_2, \dots S_M\}$. Each edge server S_i can provide offloading storage for IoT devices, which is connected via the serving IoT gateway. An IoT gateway serves a group of nearby IoT devices and offloads/retrieves the store data to an IoT storage server. To reduce the data transmission time between the IoT gateway and edge server, the IoT gateway selects the edge server with the fastest transmission path, which can be determined by the delay measurement, for example, obtaining the RTT via the `Ping` message defined in the ICMP protocol. However, the delay is varying by different factors. In measurement-based selection, the real "fastest" path is measured by the average value of K measurements for each edge server S_i. Based on the measurement, we choose the edge server with the smallest average delay in the test as the storage edge server.

IV. Measurement-based Server Selection

This section studies the performance in the measurement-based selection for the edge storage by testing the transmission latency between the IoT gateway and the candidate edge storage server. In each $S_i \in \mathcal{S}$, the IoT gateway independently issues a `Measurement Request` (e.g., an ICMP `Ping Request` message) to measure the delay or the RTT to each of them. When a `Measurement Response` (e.g., a `Ping Response`) is received, the IoT gateway continuously issues another `Measurement Request` to edge server S_i until K measurement values for each server are obtained.

For $1 \leq k \leq K$, let $t_i^{(k)}$ be the delivery delay between the IoT gateway and Edge server i for the k-th `Request/Response` delivery performed by the IoT gateway. Let $T_i(K)$ be the elapsed time of the measurement-based selection corresponding to Server i, which is expressed as

$$T_i(K) = t_i^{(1)} + t_i^{(2)} + \cdots + t_i^{(k)} + \cdots + t_i^{(K)} \tag{1}$$

Based on the results, measurement-based selection chooses the storage server which has the minimum $T_i(K)$ value for data transmission. Hence, the IoT gateway selects the storage server Δ_1, where $i^* = \arg_i \min(T_i(K)), \forall S_i \in \mathcal{S}$, and

$$\Delta_1 = S_{i^*} \tag{2}$$

To evaluate the method, we assess two output metrics, including the correct selection probability

$$\alpha = \Pr[\Delta_1 = E[\Delta_1]] \tag{3}$$

and the selection time

$$t_l = \min\{T_1(K), T_2(K), \dots, T_M(K)\} \tag{4}$$

For validation purposes, when $T_i(K)$ is exponentially distributed with respective rates λ_i. The selection time t_l is also exponentially distributed with rate $\gamma = \sum_{i=1}^{M} \lambda_i$. For $K = 1$, i.e., when a single measurement is performed, the real best server is the one with $\max\{\lambda_i\}$, for $1 \leq i \leq M$. We have $\alpha = \max\{\lambda_i\}/\gamma$.

It is obvious that with a larger parameter K, we have a higher best selection probability α but a longer selection time t_l. On the other hand, with a large set of servers, i.e., M is large, we have a smaller t_l but also a lower α. In the next section, we need to consider the trade-off between them.

V. Numerical examples

In this section, the effectiveness of the measurement-based selection is investigated by means of several numerical examples conducted in C++. The performance of the proposed method is evaluated in terms of two metrics, namely the correct selection probability α, the IoT gateway correctly selects the best edge server, and $E[t_l]$ and the average waiting time for determining the selected server. The numerical examples presents the results of the effect of K and the effect of the number of edge servers, M.

A. Effect of number of measurement rounds, K

In the edge server selection process, the IoT gateway uses the measurement-based selection to choose the fastest offloading path among the edge servers. We assume that the measurement delay between each edge server S_i and the IoT gateway follows an Erlang distribution with mean $\frac{n_i}{\lambda_i}$, which have the scale parameter $1/\lambda_i$ and the shape parameter n_i. When $n_i = 1$, the measurement delay is exponentially distributed. The edge server generating the smallest measurement delay is then regarded as the best server for performing the storage offloading process. We consider that there are five edge servers with parameters of $n_i = 3/K$ for all five servers, and $\lambda_1 = 15$, $\lambda_2 = 12$, $\lambda_3 = 9$, $\lambda_4 = 6$, $\lambda_5 = 3$, respectively. Thus, S_1 is expected to be the best serving edge server Δ_1 since it has an average measurement delay of 0.2 seconds, whereas the measurement delays of the other four servers are all greater than 0.25 seconds.

Table I shows the simulation results for the distribution of the top three edge servers selected from the five edge servers. It is seen that increasing K can make S_1, which is actually the best server, get chosen more frequently. It is seen that S_1 and S_2 are selected as storage server (denoted as Δ_1), as the two highest probability values that the IoT gateway correctly select a couple of servers to perform storage offloading process, while the S_3 is not selected to join the activity.

For $K = 1$, a single measurement round is performed in the server selection process, every edge server has a amount of probability of producing the smallest measurement delay between S_i and the IoT gateway in some tests. As a result, even though S_1 is selected most frequently among all the servers, its selection probability does not reach 50%, while those of servers S_2 and S_3 are just 29% and 16%, respectively. For higher values of K, however, the probability of S_1 being selected increases markedly to almost 70% at $K = 5$, while those of S_2 and S_3 reduce to approximately 25% and 5%, respectively. In other words, as the number of measurement rounds increases, the correct best server S_1 is more frequently selected as the probability of the best edge server also increases. However, the improved reliability of the selection process is obtained at the expense of a higher latency (t_l). For example, given the use of a single measurement round, the IoT gateway requires just 0.1317 seconds to determine the best edge server for offloading purposes. However, for $K = 5$, the processing time increases to 0.9250 seconds. Thus, in determining the optimal value of K, it is necessary to reach a satisfactory tradeoff between the latency of the server selection process and the reliability of the selection results.

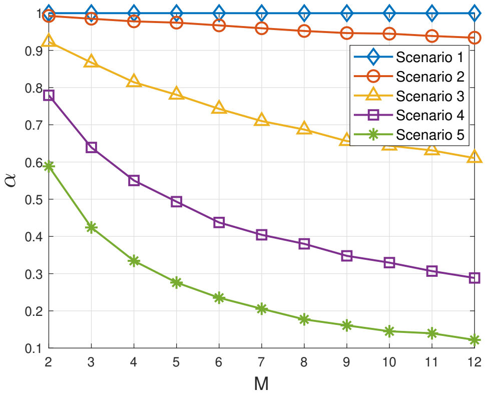
TABLE I: The performance against the number K measurements collected.

K	$\Pr[\Delta_1 = S_1]$	$\Pr[\Delta_1 = S_2]$	$\Pr[\Delta_1 = S_3]$	$E[t_l]$
1	45.8%	29.2%	16.5%	0.1317
2	55.8%	30%	11.5%	0.3239
3	61.1%	28.6%	9.1%	0.5254
4	66.2%	26.3%	7%	0.7230
5	69.3%	25.3%	5%	0.9250

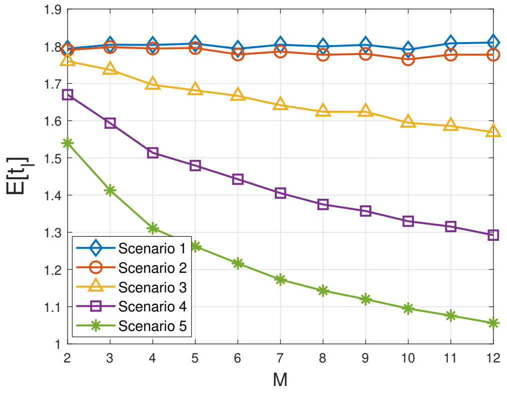
Network condition	λ_i $(i \neq 1)$
Scenario 1	$0.1\lambda_1$
Scenario 2	$0.3\lambda_1$
Scenario 3	$0.5\lambda_1$
Scenario 4	$0.7\lambda_1$
Scenario 5	$0.9\lambda_1$

TABLE II: The initial network settings of five storage servers for each scenario ($M = 5$, $m_{1-5} = 3$, $\lambda_1 = 5$).

B. Effect of number of edge servers, M



(a) α against M.



(b) $E[t_l]$ against M.

Fig. 3: Effect of a number of edge servers, M.

Table II shows the initial settings of the five edge servers in each network configuration scenario. In this table, λ_1 is fixed and always larger than other λ_i, for $i \geq 2$. Therefore, S_1 is the best server. As discussed above, the probability that the IoT gateway correctly selects the best edge server (α) and the latency of the server selection process ($E[t_l]$) both increase with increasing K. However, in practice, if the same edge server is always selected as the storage server, its workload increases dramatically. Therefore, increasing the competition among the servers by adding more servers to the node is beneficial in easing the workload of the storage server since the IoT gateway then has a greater freedom of choice in selecting the good-quality server. Consequently, the second numerical example investigated the performance of the measurement-based selection for various numbers of edge servers in the node ranging from $M = 2 \sim 12$.

In scenario 1, the measurement performance of server S_1 is configured to be higher than that of any of the other servers, and hence it is always selected as the best server, irrespective of the number of servers in the scenario (see Fig. 3a). However, for all of the other scenarios, the probability that S_1 is correctly selected as the storage server reduces with increasing M. For example, in the second scenario, the other servers, S_i, have a greater likelihood of being selected in place of S_1 as the number of servers increases. In particular, when the scenario contains just two servers, server S_1 is almost always selected as the storage server Δ_1. However, as the number of servers increases to $M = 12$, the other servers, S_i, are more frequently selected as the best server and hence α reduces to a final value of 93%. In the fifth scenario, α reduces rapidly with increasing M since all of the edge servers have a comparable RTT performance and thus have an almost equal chance of being selected as the storage server. As a result, the probability of S_1 being correctly chosen as the best server reduces to just 12% when 12 servers are deployed at the same group.

For the first and second scenarios, server S_1 is more frequently selected as the best server than the other servers when the number of measurement rounds is set as $K = 3$. Thus, the latency prediction in both scenarios is determined mainly by the delay of S_1. Consequently, referring to Fig. 3b, the latency, $E[t_l]$, in scenario 1 maintains a constant value of approximately 1.8 seconds as the number of servers increases, while that in scenario 2 has a stable value of approximately 1.78 seconds. For the other three measurement value of S_1 since all of the other servers have a relatively higher chance of generating the lowest delay value. Thus, the latency is lower than that in scenarios 1 and 2 and reduces with increasing M. In scenario 5, every server has an approximately equal chance of being selected as the storage server and hence the latency value reduces rapidly as a greater number of servers are deployed in this scenario. For example, given two servers in the same network group, the latency of the IoT gateway in searching for the best server is around 1.54 seconds. However, as the number of servers is increased to $M = 12$, the latency reduces to just 1.056 seconds.

VI. Conclusion

In IoT storage offloading, the end-users do not need to communicate with the cloud server directly, and hence the communication delay is reduced and the data security improved. However, the quality of the communication channel between the IoT gateway and the edge server has a critical effect on the offloading performance; particularly for delay-intolerant IoT applications sensing real-time data. Accordingly, the present study demonstrates the performance in measurement-based approach towards choosing the edge server capable of responding rapidly to the user offloading requirements. The simulation results have shown that the proposed method not only provides an effective means of identifying the most suitable edge server for offloading purposes, but also incurs only a short latency in the case where all of the edge servers have a comparable network environment. Furthermore, since the K-Measurement method is specifically designed to discover the edge servers with the fastest offloading path, users can be accordingly confident to complete their activities.

Acknowledgment

The work of S.-I. Sou was sponsored in part by Ministry of Science and Technology (MOST), Taiwan, under the contract number MOST 108-2628-E-006-006-MY3.

References

[1] J. Yao and N. Ansari, "QoS-Aware Fog Resource Provisioning and Mobile Device Power Control in IoT Networks," *IEEE Transactions on Network and Service Management*, vol. 16, no. 1, pp. 167-175, March 2019.

[2] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the edge: A scalable IoT architecture based on transparent computing," *IEEE Netw.*, vol. 31, no. 5, pp. 96–105, Aug. 2017.

[3] A. Colakovic and M. Hadzialic, "Internet of Things (IoT): a review of enabling technologies, challenges, and open research issues," *Computer Networks*, vol. 144, pp. 17-39, Oct. 2018.

[4] Z. Guan et al., "Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities," *IEEE Commun. Mag.*, vol. 56, no. 7, pp. 82–88, 2018.

[5] K.-K. R. Choo, R. Lu, L. Chen, and X. Yi, "A foggy research future: Advances and future opportunities in fog computing research," *Future Gener. Comput. Syst.*, vol. 78, pp. 677–697, Jan. 2018.

[6] Y.-Y. Shih, W.-H. Chung, A.-C. Pang, T.-C. Chiu, and H.-Y. Wei, "Enabling Low-Latency Applications in Fog-Radio Access Network," *IEEE Network*, vol. 31, no. 1, pp. 52–58, 2017.

[7] U. Shafi, M. Zeeshan, N. Iqbal, N. Kalsoom and R. Mumtaz, "An Optimal Distributed Algorithm for Best AP Selection and Load Balancing in WiFi," *International Conference on Smart Cities: Improving Quality of Life Using ICT IoT (HONET-ICT)*, Islamabad, 2018, pp. 65-69.

[8] W. Zhang, K. Yu, W. Wang and X. Li, "A Self-Adaptive AP Selection Algorithm Based on Multiobjective Optimization for Indoor WiFi Positioning," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1406-1416, 1 Feb.1, 2021.

[9] A. Elgazar, M. Aazam, and K. Harras, "EdgeStore: Leveraging Edge Devices for Mobile Storage Offloading," *IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, Nicosia, 2018, pp. 56-61.

[10] Z. Chang, L. Liu, X. Guo and Q. Sheng, "Dynamic Resource Allocation and Computation Offloading for IoT Fog Computing System," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3348-3357, May 2021.

[11] X. Xu et al., "Edge Server Quantification and Placement for Offloading Social Media Services in Industrial Cognitive IoV," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2910-2918, April 2021.

[12] P. O. Flaithearta, H. Melvin and M. Schukat, "A QoS enabled WiFi AP," *IEEE Network Operations and Management Symposium (NOMS)*, Krakow, 2014, pp. 1-4.

[13] E. Fakhfakh and S. Hamouda, "Incentive reward for efficient WiFi offloading using Q-learning approach," *International Wireless Communications and Mobile Computing Conference (IWCMC)*, Valencia, 2017, pp. 1114-1119.