Load-Aware Edge Server Placement for Mobile Edge Computing in 5G Networks

Xiaolong Xu[1], Yuan Xue[1], Lianyong Qi[2](✉), Xuyun Zhang[3], Shaohua Wan[4], Wanchun Dou[5](✉), and Victor Chang[6]

[1] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China
njuxlxu@gmail.com, xueyuannuist@gmail.com

[2] School of Information Science and Engineering, Qufu Normal University, Qufu, China
lianyongqi@gmail.com

[3] Department of Electrical and Computer Engineering, University of Auckland, Auckland, New Zealand
xuyun.zhang@auckland.ac.nz

[4] School of Information and Safety Engineering, Zhongnan University of Economics and Law, Wuhan, Hubei, China
shaohua.wan@ieee.org

[5] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
douwc@nju.edu.cn

[6] School of Computing & Digital Technologies, Teesside University, Middlesbrough, UK
Victor.Chang@xjtlu.edu.cn

Abstract. Edge computing is a promising technique for 5G networks to collect a wide range of environmental information from mobile devices and return real-time feedbacks to the mobile users. Generally, the edge servers (ESs) are both contributing in macro-base station (MABS) sites for large-scale resource provisioning and micro-base station (MIBS) sites for light-weighted resource response. However, to lower the investment of construing the edge computing systems in the MIBS sites, limited number of ESs are employed, since there is an intensive distribution of MIBSs in 5G networks. Thus, it remains challenging to guarantee the execution efficiency of the edge services and the overall performance of the edge computing systems with limited ESs. In view of this challenge, a load-aware edge server placement method, named LESP, is devised for mobile edge computing in 5G networks. Technically, a decision tree is constructed to identify the MIBSs served by a definite ES and confirm the data transmission routes across MIBSs. Then, the non-dominated sorting genetic algorithm II (NSGA-II) is employed to obtain the balanced ES placement strategies. Furthermore, simple additive weighting (SAW) and multiple criteria decision making (MCDM) techniques are leveraged to recognize the optimal ES placement strategy. Finally, the experimental evaluations are implemented and the observed simulation results verify the efficiency and effectiveness of LESP.

S. Yangui et al. (Eds.): ICSOC 2019, LNCS 11895, pp. 494–507, 2019.
https://doi.org/10.1007/978-3-030-33702-5_38

Keywords: 5G networks · Edge computing · Edge server placement

1 Introduction

Nowadays, with the development of Internet of Things (IoT), the increment rate of smart mobile device scale has reached a ninety-two percent per year since 2006 [1]. Cellular providers intend to enhance mobile applications which require high quality and low latency to these mobile devices [2]. However, considering the spectrum technologies in the fourth generation (4G) networks, the remaining spectrum resources are insufficient to support the harsh requirements for the delay and the bandwidth from mobile applications such as telesurgery, health monitoring and transportation cruise control, which compels providers to surmount the lack of the spectrum resources [3].

Aiming to remedy the shortage of spectrum resources, cellular providers employ a brand-new spectrum technology called the millimeter-wave (mm-Wave) in 5G, which is capable of taking full advantage of spectrum resources [4]. Technically, the wave length of the mm-wave is 1 m to 10 mm, which results in 5G bandwidth reaching up to 273.5 GHz. Nevertheless, the attenuation degree of mm-Wave signal is severely impacted by fog/cloud conditions [5]. For the sake of decreasing the attenuation of the mm-Wave signal and make use of spectrum resources, the base stations in 5G are divided into macro base stations (MABSs) and micro base stations (MIBSs) in line with their coverage [6]. Then, arranging MIBSs intensively increases the spectrum density and improves the spectrum efficiency.

As massive 5G smart applications appear continuously, it is approximately impossible to deal with computation-intensive services locally due to the limited computing resources in smart mobile devices [7]. Thus, mobile devices ask for the computing resources provided by the cloud platform to execute the services [8]. In spite of that the pressure for mobile devices to handle services is relieved by the cloud, the quality of experience (QoE) for users is hard to satisfy, especially for the real-time applications like virtual reality (VR) games. Therefore, edge computing, as a significant paradigm, is utilized to shorten the delay and make users experience the applications real-timely. Detailedly, edge computing endows the computing resources to edge servers (ESs), the gathered applications on the network edge are in a position to obtain the resources in the ES close by.

On account of that mobile applications are usually in the vicinity of MIBSs, ESs are co-located with MIBSs, which are seen as the edge nodes (ENs) to cooperate with MABS for executing the applications. Nonetheless, MIBSs are arranged intensively because of its relatively narrow range. With a view to the high cost of purchasing ESs, it is unpractical to equip each MIBS with an ES, which makes some tasks offloaded to a distant ES and generates unbearable delay for users. In addition, the finite computing resources in ESs are unable to handle such abundant tasks, which makes some services wait in the queue of ESs and severely affects the efficiency of service execution. Hence, to ensure the stability and the performance of ESs, it is urgent to achieve the load balance of ESs. Given these facts, it is a truly difficult challenge to realize the reasonable ES placement for improving the performance of all ESs by reducing the transmission delay and achieving load balance. A load-aware edge server placement method named as LESP, is devised for edge computing in 5G networks in this paper. In conclusion, the primary contributions are presented as follows.

- A decision tree is structured to record the routings of edge services according to the balanced distribution of load.
- Non-dominated sorting genetic algorithm II (NSGA-II) is used to formulate the appropriate balanced ES placement strategies.
- Simple additive weighting (SAW) and multiple criteria decision making (MCDM) are adopted to select the optimal ES placement strategy.
- Simulation experiments are conducted to confirm the efficiency of the devised method LESP.

The remaining part of this paper is divided into five sections. The system model is shown in Sect. 2. A load-aware edge server placement method is designed in Sect. 3. The simulation experiment results and the comparison analysis are conducted in Sect. 4. Related Works are summed up in Sect. 5. Conclusions are outlined in Sect. 6.

2 System Model

In this section, the overview of the ES placement framework in 5G networks is presented first. Then, according to the specific ES placement strategy, transmission delay and load balance analyses are conducted. Finally, the ES placement problem is formulated as a multi-objective optimization problem.

2.1 Resource Model

In 5G networks, multiple macro-base stations (MABSs) are deployed to provision services for mobile applications and the MABS covers several micro-base station (MIBS) to improve the service quality [9]. With wireless signals, the MIBSs receive service requests from mobile devices. In Fig. 1, the framework for supporting edge computing in 5G networks is shown. In the range of the MABS, several MIBSs are deployed to receive service requests. Assume that there are Q MIBSs in the range of the MABS, denoted as $M = \{m_1, m_2, \ldots, m_Q\}$. As MIBSs fail to meet the process requirements of massive services, the computing and storage capacity of MIBSs are extended through realizing the cooperative placement of ESs and MIBSs. Thus, the ESs are co-located with the specific MIBSs for helping process transmitted services, denoted as $S = \{s_1, s_2, \ldots, s_N\}$. Notably, the number of ESs is not equal to the number of MIBSs. Besides, with the virtualization technique in edge computing, the capacity of the ES is measured by the number of virtual machines (VMs) in the ES, denoted as φ_n [10,11]. Provided that all VMs in the ES are occupied, unprocessed edge services need to wait for the completion of the services in the previous round.

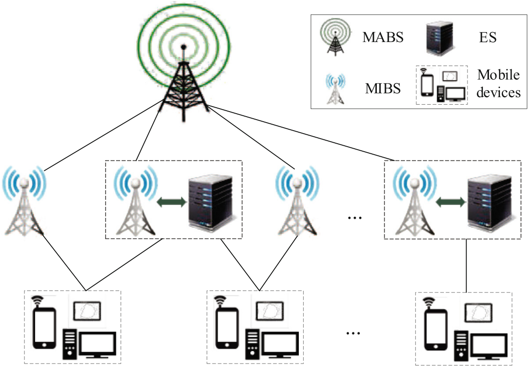


Fig. 1. An edge computing framework in 5G networks.

2.2 Transmission Delay Model

The transmission delay is composed of the transmission time from the MIBS to the destination ES, the waiting time of the edge services in the ES, the service execution time of the ES and the transmission time of the feedbacks.

In view of that not all MIBSs are cooperating with the ES, we first adopt a binary variable B_Q^N to judge whether the q-th (q = 1, 2, ..., Q) MIBS m_q is co-located with the n-th (n = 1, 2, ..., N) ES s_n.

$$B_q^n = \begin{cases} 1, & \text{if } m_q \text{ is cooperatively placed with } s_n, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

The time consumption of data transmission from the MIBS to the destination ES is calculated by

$$DT_q = (1 - B_q^n) \cdot \frac{ds_q}{\theta} \cdot \omega_q, \tag{2}$$

where ds_q represents the data size of the edge service in m_q and θ represents the data transmission rate between MIBSs. In addition, ω_q is the number of passing MIBSs in the process of data transmission.

By means of the virtualization technique, the resource units in ESs are normalized as VMs. Therefore, the service execution time in s_n is

$$ET_q = \frac{ds_q}{ru_q \cdot \rho}, \tag{3}$$

where ru_q represents the VMs demanded by the edge service transmitted from s_n and ρ is the processing power of each VM.

In order to calculate the waiting time of the edge services in the ES, the waiting rounds of the edge services need to be calculated first, which is shown as

$$R = wr_n, \tag{4}$$

where wr_n represents the waiting rounds in n-th ES s_n.

Let ET_r denote the corresponding execution time of processing all services in the z-th round. The waiting time of edge services offloaded from s_n is calculated by

$$WT_q = \begin{cases} 0, & \text{if } wr_n = 0, \\ \sum\limits_{r=1}^{R-1} \max(ET_r), & \text{otherwise.} \end{cases} \tag{5}$$

The transmission time of feedbacks from s_n is calculated by

$$FT_q = \frac{ds'_q}{\theta} \cdot \omega_q, \tag{6}$$

where $ds_q^{'}$ is the data size of the processing results of the edge service offloaded from m_q.

The total transmission delay of the edge service in m_q is calculated by

$$D_q = DT_q + ET_q + WT_q + FT_q. \tag{7}$$

The average delay for all edge services is calculated by

$$A = \frac{1}{Q} \cdot \sum_{q=1}^{Q} D_q. \tag{8}$$

2.3 Load Balance Model

The load balance conditions of ESs are measured by the load balance variance. Specifically, the occupy conditions of ESs and the number of running VMs are described. F_n is a binary variable to judge whether s_n is occupied, which is calculated by

$$F_n = \begin{cases} 1, & \text{if } s_n \text{ is occupied,} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Besides, P_q^n is a binary variable to judge whether the edge service in m_q is offloaded to s_n for execution, which is defined by

$$P_q^n = \begin{cases} 1, & \text{if } m_q \text{ offloads the service to } s_n, \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

Consequently, the number of running ESs is calculated by

$$\xi = \sum_{n=1}^{N} P_q^n. \tag{11}$$

The resource utilization of s_n is measured by the usage of VM instances, which is calculated by

$$RU_n = \frac{1}{\varphi_n} \sum_{q=1}^{Q} P_q^n \cdot \varepsilon_q, \tag{12}$$

where ε_q is the number of VMs required by the edge service in m_q.

Thus, the average resource utilization of ESs is calculated by

$$U = \frac{1}{\xi} \sum_{n=1}^{N} RU_n. \tag{13}$$

The load balance variance of s_n is calculated by

$$b_n = (RU_n - U)^2. \tag{14}$$

Then, the average load balance variance of occupied ESs in 5G networks is calculated by

$$B = \frac{1}{\xi} \sum_{n=1}^{N} b_n \cdot F_n. \tag{15}$$

2.4 Problem Formulation

In this paper, the ES placement is defined as a multi-objective optimization problem. We minimize the transmission delay in (8) and the load balance variance in (15), which is given as

$$\min A, \min B. \tag{16}$$

$$s.\,t. \quad N \le Q, \tag{17}$$

$$\sum_{q=1}^{Q} \varepsilon_q \le \sum_{n=1}^{N} \varphi_n. \tag{18}$$

3 A Load-Aware Edge Server Placement Method

In this section, the routing confirmation of edge services is presented first. Then, NSGA-II is adopted for the multi-objective optimization problem. Finally, SAW and MCDM are used to select the optimal ES placement strategy.

Algorithm 1. Routing Confirmation of edge services in MIBSs

```
Require: $M$, $S$, an empty decision tree
Ensure: Routing confirmation of edge services
  Set $flag_q$ ($q$=1, 2, ..., $Q$) as 0
  Co-locate each ES with a random MIBS
  for the edge server $s_n$ in $S$ do
    for the MIBS $m_q$ in $M$ do
      if $s_n$ is co-located with $m_q$ then
        Offload the edge service in $m_q$ to $s_n$
        Insert $m_q$ into the decision tree
        $m_q$.flag = 1
      else
        Calculate the distance between $s_n$ and $m_q$
      end if
    end for
  end for
  for the MIBS $m_q$ in $M$ do
    if $m_q$.flag = 0 then
      Seek out the minimum distance
      Calculate the depth difference $g$ between the subtrees
    end if
    if $\mid g \mid \leq 1$ then
      Offload the service and insert $m_q$ into the current subtree
    else
      Offload the service and insert $m_q$ into the opposite subtree
    end if
  end for
  return Routing confirmation of edge services
```

3.1 Routing Confirmation of Edge Services

Aiming to offload the edge services from MIBSs to ESs, the transmission routings of edge services need to be confirmed. As all ESs have been placed, decision trees are used to record the routes of edge services from MIBSs. A two-dimensional matrix is set up and for each ES, the distance between the ES and every MIBS is entered into the matrix. The MIBS with the smallest distance value offload the edge services to the ES. Provided that there are more than one ES which have the same distance with a certain MIBS, the number of MIBSs connected to each ES is compared and the edge service in the MIBS is offloaded to the ES which connects the few MIBSs.

The specific routing confirmation of edge services is presented in Algorithm 1. First, the flag of each MIBS is initialized as 0, which means the MIBS has not been traversed. Then, every ES is co-located with a MIBS randomly. The MIBSs which are co-located with an ES offload the edge service to the co-located ES. Finally, other MIBSs determine the offloading destination ES according to the ES load situation.

3.2 Edge Server Placement Strategy Generation Based on NSGA-II

In order to minimize the transmission delay and the load balance variance of all ESs, NSGA-II is used to solve the multi-objective optimization problem in (16). Firstly, the ES placement strategy is encoded, which is known as a gene. As a gene represents a placement strategy of a certain ES, multiple genes represent the placement strategy of all ESs, which constitute a chromosome. The integer coding method is adopted and the ESs are encoded by 1, 2, ..., N.

Aiming at select optimal solutions in a chromosome, fitness functions work as the standards. As shown in Sect. 2, the transmission delay and the load balance variance are the standards to select the appropriate solution. For further selection, the size of population H, the crossover capacity R_c and the mutation capacity R_m and the maximum iteration times V are determined.

Based on the existing population, crossover and mutation operations are conducted to generate new solutions. The crossover in this paper is single-point crossover, which means that two chromosomes swap genes around a predetermined intersection. Through combining two chromosomes, a better chromosome is obtained. In addition, the genes are modified randomly to generate chromosomes which have higher fitness values, known as mutation operation. During the mutation operation, each gene has the same possibility to modify.

For $2H$ solutions after the crossover and mutation operation, the selection operation is conducted to select H solutions. Specifically, the fitness functions of each solution are calculated based on the model.

According to the usual dominating principle, the solutions are sorted and the selection operation is conducted. The population generates non-dominated layers and each placement strategy owns a crowding distance respectively. Through the comparison of the crowding distances, the appropriate individuals are used to form the next population, which is calculated by

$$j_g = j_g^D + j_g^L = |D^{j+1} - D^{j-1}| + |L^{j+1} - L^{j-1}|, \tag{19}$$

where j_g represents the j-th ES placement strategy. j_g^D as well as j_g^L represents the objective functions. D_{j+1} and L_{j+1} represent the objective values of the j+1-th placement strategy. D_{j-1} and L_{j-1} represent the objective values of the j−1-th placement strategy.

3.3 Edge Server Placement Strategy Selection Using SAW and MCDM

For the last generated chromosome, SAW and MCDM are employed to select the optimal ES placement strategy. The transmission delay is normalized as

$$V(D) = \begin{cases} \frac{D^{\max} - D}{D^{\max} - D^{\min}}, D^{\max} - D^{\min} \neq 0, \\ 1, D^{\max} - D^{\min} = 0, \end{cases} \tag{20}$$

where D^{max} and D^{min} represent the maximum and minimum transmission delay of the solutions in the last population respectively.

Moreover, the load balance variance is normalized as

$$V(L) = \begin{cases} \frac{L^{\max}-L}{L^{\max}-L^{\min}}, L^{\max} - L^{\min} \neq 0, \\ 1, L^{\max} - L^{\min} = 0, \end{cases} \tag{21}$$

where L^{max} and L^{min} represent the maximum and minimum load balance variance of the solutions in the last population respectively.

Aiming to achieve the optimization of normalized transmission delay and load balance, the utility value of the h-th solution needs to calculated, which is shown as

$$V(C_h) = w_1 V(D) + w_2 V(L), \tag{22}$$

where w_1 and w_2 are the weight of transmission delay and load balance variance respectively.

Based on the utility value of the solution, the optimal strategy is selected by

$$V(C) = \max_{h=1}^{H} V(C_h)(1 \leq h \leq H). \tag{23}$$

3.4 Method Overview

In this paper, a load-aware edge server placement method is devised to minimize the transmission delay and the load balance variance. The specific procedure of this method is shown in Algorithm 2. First, a decision tree is structured and the routing of edge services is determined. Then, NSGA-II is adopted to generate balanced ES placement strategies. Finally, SAW and MCDM are utilized to identify the optimal ES placement strategy from the generated population.

Algorithm 2. Load-aware edge server placement

```
Require: $M$, $S$
Ensure: The optimal edge server placement strategy
    $e = 1$
    while $e \leq E$ do
        Complete routing confirmation by Algorithm 1
        $e = e + 1$
    end while
    Obtain balanced ES placement strategies by NSGA-II
    for $h = 1$ to $H$ do
        Calculate utility values by formulas (20-22)
    end for
    Select the optimal ES placement strategy by formula (23)
    return The optimal edge server placement strategy
```

4 Experimental Evaluation

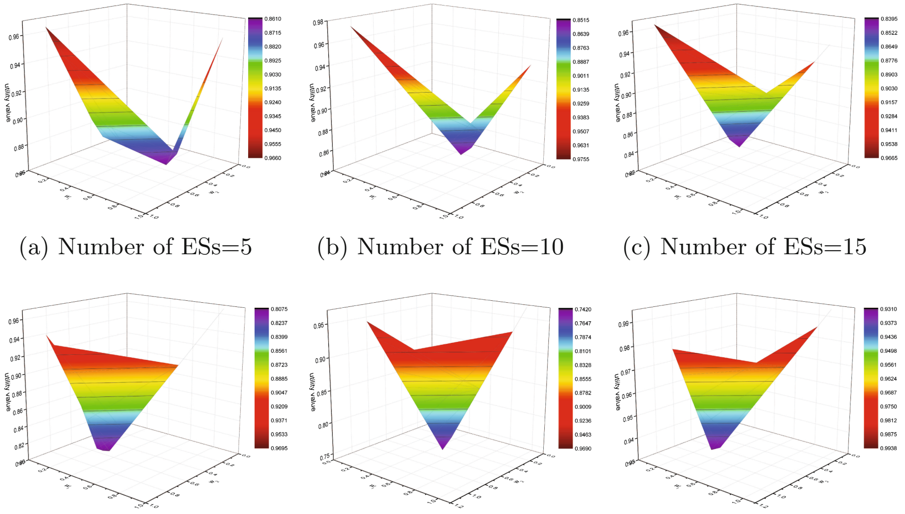
In this section, the efficiency and effectiveness of LESP are verified by conducting simulation experiments. Firstly, the parameter settings are presented in Table 1. Then, the comparative methods are introduced. Finally, the influences of different MIBS scales on the transmission delay and load balance variance performance of LESP and the comparative methods are evaluated.

Table 1. Parameter settings

Parameter description	Value
The total number of MIBSs	200
The number of VMs in each edge server	40
The number of VMs required by each MIBS	[1, 7]
The transmission rate between MIBSs	5000 Mb/s
The execution capacity of edge server	2000 MHz
The scales of MIBSs	5, 10, 15, 20, 25, 30

4.1 Simulation Setup

We adopt 5 different ES scales in the experiments and the number of ESs is set to 5, 10, 15, 20, 25 and 30. To more intuitively evaluate the performance of LESP, two comparative methods are utilized, which are shown as follows.



(a) Number of ESs=5

(b) Number of ESs=10

(c) Number of ESs=15

(d) Number of ESs=20

(e) Number of ESs=25

(f) Number of ESs=30

Fig. 2. The utility value of solutions at different MIBS scales.

- *Greedy-D*: Each MIBS offloads its edge service to the nearest ES. Considering that the nearest ES is likely to have no spare computing resources, the edge service is offloaded to a neighbor ES with enough needed computing resources. The system would repeat this procedure until all edge services have been offloaded.
- *Greedy-L*: Each MIBS offloads its edge service to the ES which has most idle computing resources. Provided that several ESs have the similar resource usage, the service is offloaded to the ES which is nearest to the resource MIBS. This procedure is repeated until all edge services have been offloaded.

4.2 Performance Evaluation

When the MIBS scales are 5, 10, 15, 20, 25 and 30 respectively, the weight of $V(D)$ and $V(L)$ in the formula (22) are changed, which are from 0 to 1. In Fig. 2, as the weight of $V(D)$ and $V(L)$ change, the utility value alters correspondingly. Considering that there is a linear relationship between the two weights and the utility value, the utility value graph is divided into two sections and there is a minimum utility value. By means of comparing utility values, the most balanced ES placement strategy is selected. The solution with the maximum utility value is selected as the optimal ES placement strategy.

4.3 Comparison Analysis

For different MIBS scales, the comparisons between LESP and comparative methods are presented. The transmission delay and the load balance variance are functioned as two key criteria. The corresponding experimental results are shown in Figs. 3 and 4.

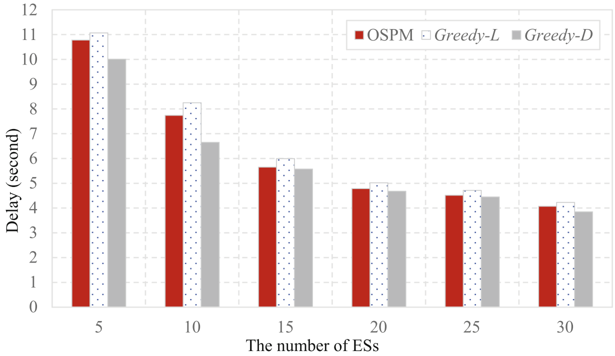


Fig. 3. Comparison of the delay for different MIBS scales between OSPM and comparative methods.

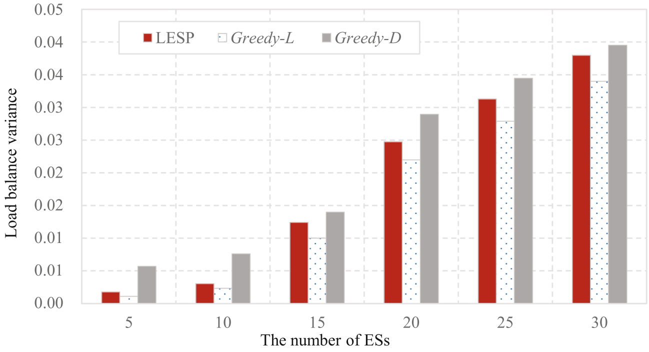


Fig. 4. Comparison of the load balance variance for different MIBS scales between OSPM and comparative methods.

(1) Comparison of Transmission Delay. In the experiment, the processing efficiency of each ES is defined as equal. Therefore, with the determination of edge service routings, the service execution time is the same in LESP, Greedy-D and Greedy-L. In Fig. 3, as the size of the ES expands, the transmission delay decreases because MIBSs are more likely to offload edge services to a close ES than to a remote ES. Nevertheless, Greedy-L aims to minimize load balance variance and the edge services are likely to be offloaded to relatively remote ES, achieving the balanced distribution of load and high transmission delay. Intuitively, LESP is less capable of optimizing transmission delay than Greedy-D and is more effective than Greedy-L in the aspect of optimizing transmission delay.

(2) Comparison of Load Balance Variance. The load balance variance is measured by the occupancy of VM instances. When the ES scale is small, the data waits for limited computing resources in the ES in the queue and the total number of rounds processed by the ES is large. In Fig. 4, as the number of ES increases, more computing resources are vacated and the total round of service execution decreases. In contrast to Greedy-L, Greedy-D is designed to minimize transmission delay. In Greedy-D, the edge services are offloaded to the nearest ES whose VMs are all occupied, making the destination ES overload. Intuitively, the ability to optimize load balancing of LESP is inferior to Greedy-L and superior to Greedy-D.

5 Related Work

Nowadays, as the wireless and mobile communication technologies develop rapidly, the number of smart mobile devices increased dramatically. These smart mobile devices, which possess numerous compute-intensive applications, lead to a large quantity of data needed to be processed [12,13]. However, smart mobile devices are unable to deal with the data for its limited computing resources. Therefore, the 5th generation mobile network (5G), as a novel paradigm, is proposed to help devices offload the data to the remote infrastructures such as the cloud for processing. In [14], Islam et al. probed into a radio access technology called non-orthogonal multiple access (NOMA). NOMA can relieve the pressure from the scarce spectrum resource in 5G for its greater spectrum efficiency. Niu et al. discussed the applying of the millimeter wave (mm-wave) in 5G. The small cell access and the wireless backhaul are investigated to accelerate the mm-wave deployment [13].

Nevertheless, the remote distance between smart mobile devices and the remote infrastructure generates an unbearable delay for users. Hence, edge computing emerges. Edge servers, which own the computing and storage capabilities, are placed close to mobile devices for decreasing the delay [15,16]. In [17], Nunna et al. investigated the combination of 5G and mobile edge computing. In addition, Rimal et al. considered providing mobile edge computing (MEC) capabilities of integrated fiber-wireless (FiWi), making MEC fit in 5G [18]. If all computing tasks are offloaded to the edge servers without planning. The efficiency of edge servers is influenced hugely [19,20]. Therefore, it is significant to realize the optimal edge server placement to achieve the load balance and improve the delay.

6 Conclusion and Future Work

Edge computing emerges as an appropriate paradigm in 5G networks to collect environmental parameters and return processing results to users. As the number of edge services increases, chances are that the execution efficiency of edge services is hardly to guarantee with limited ESs. In this paper, we formulate the ES placement problem as a multi-objective problem. A load-aware edge server placement method named LESP is devised. To demonstrate that LESP is efficient and feasible, the performance of LESP is evaluated through experimental simulations.

In the future, we will improve LESP to adapt to the real scene. The different processing capacities of ESs will be specified and the corresponding offloading strategies will be revised.

Acknowledgment. This work was supported by the National Key Research and Development Program of China (No. 2017YFB1400600). Besides, this research is supported by the National Natural Science Foundation of China under grant no. 61702277, no. 61872219 and no. 616722763. This research is also supported by College Students' Enterprise and Entrepreneurship Education Program of NUIST, CSEEEP.

References

1. Chen, M., Qian, Y., Hao, Y., Li, Y., Song, J.: Data-driven computing and caching in 5G networks: architecture and delay analysis. IEEE Wireless Commun. **25**(1), 70–75 (2018)
2. Sun, S., Rappaport, T.S., Shafi, M., Tang, P., Zhang, J., Smith, P.J.: Propagation models and performance evaluation for 5G millimeter-wave bands. IEEE Trans. Veh. Technol. **67**(9), 8422–8439 (2018)

3. Gringoli, F., Patras, P., Donato, C., Serrano, P., Grunenberger, Y.: Performance assessment of open software platforms for 5G prototyping. IEEE Wireless Commun. **25**(5), 10–15 (2018)
4. Skouroumounis, C., Psomas, C., Krikidis, I.: Heterogeneous FD-mm-wave cellular networks with cell center/edge users. IEEE Trans. Commun. **67**(1), 791–806 (2019)
5. Ordonez-Lucena, J., Ameigeiras, P., Lopez, D., Ramos-Munoz, J.J., Lorca, J., Folgueira, J.: Network slicing for 5G with SDN/NFV: concepts, architectures, and challenges. IEEE Commun. Mag. **55**(5), 80–87 (2017)
6. Duan, P., Jia, Y., Liang, L., Rodriguez, J., Huq, K.M.S., Li, G.: Space-reserved cooperative caching in 5G heterogeneous networks for industrial IoT. IEEE Trans. Industr. Inf. **14**(6), 2715–2724 (2018)
7. Rodrigues, T.G., Suto, K., Nishiyama, H., Kato, N.: Hybrid method for minimizing service delay in edge cloud computing through VM migration and transmission power control. IEEE Trans. Comput. **66**(5), 810–819 (2016)
8. Li, H., Ota, K., Dong, M.: Learning IoT in edge: deep learning for the internet of things with edge computing. IEEE Netw. **32**(1), 96–101 (2018)
9. Kim, Y., Kwak, J., Chong, S.: Dual-side optimization for cost-delay tradeoff in mobile edge computing. IEEE Trans. Veh. Technol. **67**(2), 1765–1781 (2017)
10. Boulogeorgos, A.A.A., et al.: Terahertz technologies to deliver optical network quality of experience in wireless systems beyond 5G. IEEE Commun. Mag. **56**(6), 144–151 (2018)
11. Li, M., Yu, F.R., Si, P., Zhang, Y.: Green machine-to-machine communications with mobile edge computing and wireless network virtualization. IEEE Commun. Mag. **56**(5), 148–154 (2018)
12. Beyranvand, H., Lévesque, M., Maier, M., Salehi, J.A., Verikoukis, C., Tipper, D.: Toward 5G: FiWi enhanced LTE-A HetNets with reliable low-latency fiber backhaul sharing and WiFi offloading. IEEE/ACM Trans. Networking **25**(2), 690–707 (2016)
13. Mozaffari, M., Kasgari, A.T.Z., Saad, W., Bennis, M., Debbah, M.: Beyond 5G with UAVs: foundations of a 3D wireless cellular network. IEEE Trans. Wireless Commun. **18**(1), 357–372 (2019)
14. Richardson, T., Kudekar, S.: Design of low-density parity check codes for 5G new radio. IEEE Commun. Mag. **56**(3), 28–34 (2018)
15. Lyu, X., Tian, H., Ni, W., Zhang, Y., Zhang, P., Liu, R.P.: Energy-efficient admission of delay-sensitive tasks for mobile edge computing. IEEE Trans. Commun. **66**(6), 2603–2616 (2018)
16. Ning, Z., Kong, X., Xia, F., Hou, W., Wang, X.: Green and sustainable cloud of things: enabling collaborative edge computing. IEEE Commun. Mag. **57**(1), 72–78 (2019)
17. Bi, S., Zhang, Y.J.: Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading. IEEE Trans. Wireless Commun. **17**(6), 4177–4190 (2018)
18. Wang, R., Yan, J., Wu, D., Wang, H., Yang, Q.: Knowledge-centric edge computing based on virtualized D2D communication systems. IEEE Commun. Mag. **56**(5), 32–38 (2018)
19. Wang, K., Yin, H., Quan, W., Min, G.: Enabling collaborative edge computing for software defined vehicular networks. IEEE Network **99**, 1–6 (2018)
20. Hou, W., Ning, Z., Guo, L.: Green survivable collaborative edge computing in smart cities. IEEE Trans. Industr. Inf. **14**(4), 1594–1605 (2018)