Distance-aware Edge User Allocation with QoE Optimization

Zhiwei Xu[1], Guobing Zou[1*], Xiaoyu Xia[2], Ya Liu[1], Yanglan Gan[3], Bofeng Zhang[1] and Qiang He[4*]
[1]*School of Computer Engineering and Science, Shanghai University, Shanghai, China*
[2]*School of Information Technology, Deakin University, Burwood, Australia*
[3]*School of Computer Science and Technology, Donghua University, Shanghai, China*
[4]*School of Software and Electrical Engineering, Swinburne University of Technology, Hawthorn, Australia*
Email: gbzou@shu.edu.cn, qhe@swin.edu.au

***Abstract*—Nowadays, the world is witnessing a rapid development of edge computing. As an important issue in the edge computing paradigm, the edge user allocation (EUA) problem has attracted considerable attention. EUA aims at allocating the end-users in a specific area to the edge servers in that area, and ensure end-users' low-latency access to app vendor's services deployed on those edge servers. However, existing approaches simply assume that each edge server has a specific coverage and neglect the complexity of wireless signal transmission. To ensure end-users' low latency, an EUA approach must take into account the distance between end-users and their nearby edge servers, as it significantly impacts their Quality of Experience (QoE). Accordingly, EUA must maximize the overall QoE of the app vendor's users. To tackle this new distance-aware EUA problem, we propose two novel approaches, namely DEUA-O and DEUA-H. DEUA-O aims to find the optimal solution while DEUA-H aims to find the sub-optimal solution in large-scale scenarios efficiently. Four series of experiments are conducted on a real-world dataset to evaluate DEUA-O and DEUA-H. The results demonstrate the substantial gains of our approaches over the state-of-the-art.**

***Keywords*-Edge computing; Edge User Allocation; Signal Strength; Quality of Service; Quality of Experience; Edge Service**

I. INTRODUCTION

In recent years, the Internet of Things (IoT), such as smartphones, wearables and tablets, has been proliferated worldwide, and then it was widely applied to many fields such as healthcare, home, environment and transports [1]. The rapid development of IoT devices gives the IT industry a tremendous boost. However, this also has put forward an enormous challenge for ensuring low response time for end-users and long battery life for their IoT devices [2]. For example, a smart vehicle can generate two petabytes of data per second, and it requires real-time processing to make a timely decision. Sending all the data to the remote cloud server to be processed often results in high and unpredictable latency, leading to unexpected accidents. App vendors are facing the challenge of maintaining low-latency connections for their users.

To attack this challenge, edge computing, a new distributed computing paradigm, has emerged to allow computing resources such as CPU, memory and storage to be distributed to edge servers at the edge of the cloud [3]. Each edge server is powered by one or more physical servers and deployed at base stations that are geographically close to end-users. In this way, app users can be allocated to edge servers to guarantee the low-latency and reliable connection. Powered by 5G, edge computing thus plays an extraordinarily significant role in the IT industry where low latency and energy optimization are extremely desired in real-world service-oriented application scenarios.

Offering a variety of new opportunities, edge computing also raises many new problems. The edge user allocation (EUA) problem as one of those has attracted a lot of attention very recently [4]–[6]. With the consideration of uneven user distribution in an area and the constrained computing resources on edge servers, the general idea of EUA is to allocate users to edge servers to achieve a specific optimization goal, e.g., to fulfill the users' needs of computing resources measured by their Quality of Service (QoS), to minimize the number of edge servers needed [4] and to maximize the overall user satisfaction measured by their Quality of Experience (QoE) [5]. However, existing approaches simply ignore the complexity of wireless signal transmission and assume that every edge server has a specific coverage radius [4]–[6]. This is unrealistic in the real world for twofold reasons: 1) The signal strength, or data rate, may not immediately drop to zero when edge users are just out of the coverage. Instead, it declines as the distance between the edge servers and users increases [7]; 2) The user's QoE attained by the corresponding server largely depends on the data rate, and thus it would be also attenuated during the wireless transmission. Therefore, it is critical to consider the users' distances from edge servers when allocating them to the nearby edge servers.

We refer to this problem as the *distance-aware edge user allocation* (distance-aware EUA) problem. To solve this problem, we propose two novel approaches that maximize users' overall QoE, taking into account their distances from nearby edge servers. To the best of our knowledge, this is the first attempt to study the distance-aware EUA problem from the app vendor's perspective. The main contributions of this paper are as follows:

[*] Guobing Zou (gbzou@shu.edu.cn) and Qiang He (qhe@swin.edu.au) are the corresponding authors of this work.

978-1-7281-8786-0/20/$31.00 ©2020 IEEE
DOI 10.1109/ICWS49710.2020.00016

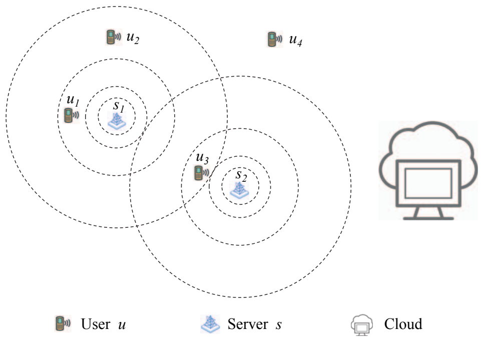


Figure 1. An example of distance-aware EUA problem.

- We formally define and model the distance-aware EUA problem, and prove its $\mathcal{NP}$-hardness;
- We propose an optimal approach, named DEUA-O, based on Integer Linear Programming (ILP), and propose a heuristic approach, named DEUA-H, for finding sub-optimal solutions in large-scale scenarios.
- Extensive experiments have been conducted on a real-world dataset to demonstrate the effectiveness and efficiency of the two proposed approaches against the state-of-the-art.

The remainder of this paper is organized as follows. Section II motivates this research using an example. Section III introduces the basic notations, defines the problem, and then presents the proposed approaches as well as proves the $\mathcal{NP}$-hardness of the distance-aware EUA problem. Section IV experimentally evaluates the proposed approaches on a real-world dataset. Section V reviews the related work. Section VI concludes the paper and points out future works.

II. Motivating Example

Figure 1 presents an example of distance-aware EUA scenario with four users $\{u_1, u_2, u_3, u_4\}$ and two edge servers $\{s_1, s_2\}$ hired by an app vendor to accommodate those users' requests. In addition, a remote server is available in the cloud to accommodate the requests of the users who cannot be allocated to any edge servers, e.g., user u_4.

In the existing work [4]–[6], it is assumed that each edge server has a specific coverage radius and the users covered by the same edge server will have the same data rate. However, this is unrealistic. In the real world, when a user communicates with an edge server, the wireless transmission between them strictly follows a slow attenuation pattern [8]. In general, the wireless signal strength relies on the distance between the user and the edge server (referred to as *distance* for short hereafter), the closer the stronger. Take Figure 1 for example. Users u_1, u_2 and u_3 are covered by edge server s_1. Since u_1 is close to s_1, u_1 will receive a stronger wireless signal from s_1 than u_2 and u_3, and consequently a higher data rate. Being far away from s_1, u_3 might not be able to receive a satisfactory data rate, which lowers u_3's QoE. However, this is not considered by the existing approaches [4]–[6]. Aiming to minimize the number of edge servers needed, existing approaches [4] will allocate u_1, u_2 and u_3 to s_1. As a matter of fact, u_3 might have to be allocated to s_2 to ensure a satisfactory data rate for u_3. Thus, when allocating users to edge servers, the distance between the users and the edge servers must be considered. Otherwise, the app vendor's objective cannot be achieved, i.e., to maximize its users' overall QoE [5].

Table I
Notations

Notation	Description
T = {CPU, RAM, Storage, Bandwidth}	Set of resource types
n	Total number of users
m	Total number of edge servers
q	Total number of of QoS / QoE levels
u_i	i-th user in U
s_j	j-th edge server in S
$d_{i,j}$	Geographical distance between user u_i and edge server s_j
$\gamma(d_{i,j})$	Attenuation coefficient caused by distance $d_{i,j}$
c_j^k	Capacity of k-th resource type of edge server s_j
$U = \{u_1, u_2, ..., u_n\}$	Set of users
$S = \{s_1, s_2, ..., s_m\}$	Set of edge servers
$C_j = \{c_j^1, c_j^2, ..., c_j^h\}$	Capacity of edge server s_j
$W = \{w_1, w_2, ..., w_q\}$	Set of QoS level l
$E = \{e_1, e_2, ..., e_q\}$	Set of QoE with QoS level l

In the real world, the scale of this distance-aware EUA problem can be much larger than the one presented in Figure 1. The EUA problem was proven to be $\mathcal{NP}$-hard [4]. The distance awareness further increases the complexity of the distance-aware EUA problem. Finding an optimal solution in a large-scale distance-aware EUA problem is not trivial.

III. Approach

A. Problem Definition

With the consideration of distance, we define the distance-aware EUA problem as follows:

Definition 1. *Given a set of edge servers denoted as S and app users denoted as U, The distance-aware EUA problem aims to find an allocation $f : U \to S$, which maximizes the overall QoE of users while fulfils all the users' resource requirements, including that data rate that are impacted by wireless signal attenuation.*

Similar to many studies of edge computing [4], [5], [9], we investigate the distance-aware EUA problem in quasi-static scenarios where the user distributions remain unchanged during the allocation, e.g., their resource needs and locations. The notations used in the paper are summarized in Table I.

To solve the distance-aware problem, we propose two approaches, namely DEUA-O and DEUA-H. DEUA-O is an optimal approach for finding optimal solutions to small-scale distance-aware EUA problem. Then, we prove the $\mathcal{NP}$-hardness of the distance-aware EUA problem based on the optimization model. To solve the distance-aware EUA problem efficiently in large-scale scenarios, DEUA-H employs a heuristic to find sub-optimal solutions.

B. Optimal Approach

Given a set of users and a set of edge servers in a particular area, the app vendor's objective is to maximize these users' overall QoE [5]. In this research, we measure a user's QoE in the same way as [5], which highly depends on the Quality of the Service (QoS) delivered to the user. The correlation between QoS and QoE is application-specific. For example, a YouTuber user's QoE mainly relies on video resolution and frames per second while a Uber user is more sensitive to service latency. In the general EUA scenario, a user's QoE relies on the computing resources it receives from the edge server, e.g., CPU, RAM, storage and data rate [5]. In this research, the distance impacts the data rate, but not the CPU, RAM or storage. Thus, a user's QoE is measured by its data rate. As reported in [10], in general, as the QoS increases, a user's QoE does not obey a linear open-ended increase. Instead, it starts to increase slowly at first, then speeds up, and finally converges. Take YouTube for example. A mobile YouTube user's QoE increases slightly when the video resolution increases from 240p to 360p. As the video resolution continues to increase to 720p or 1080p, the user's QoE increases significantly. However, as video resolution further increases to 2k, the user's QoE increases only slightly or does not even increase as it cannot tell the difference between a 1080p video and a 2k video on a mobile device. The same applies to a user's QoE measured by data rate. In general, the correlation between QoE and QoS can be generalized with a sigmoid function [11], also known as logistic function, which is also employed in [5]. Formally, it is represented as follows:

$$e_l = \frac{L}{1 + e^{-\alpha(w_l - \beta)}} \tag{1}$$

where e_l and w_l denote QoE and QoS, L is the maximum value of QoE, α is the growth rate of the curve, and β is the value at the middle point of the curve. In this way, a user's QoE can be measured quantitatively and integrated into a domain-specific distance-aware EUA problem.

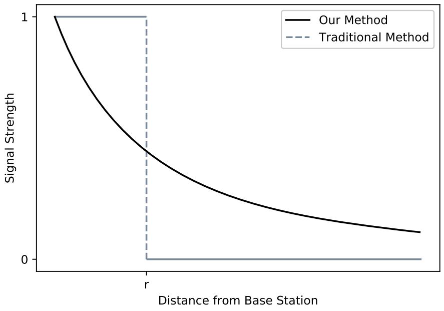


Figure 2. Quantitative correlation between distance and signal strength.

To solve the distance-aware EUA problem, the attenuation of data rate in wireless transmission must be taken into account. The Free Space Path Loss (FSPL) [8] model is part of the IEEE 802.11 standard, which is the world's most widely used wireless computer networking standards [7]. In this model, the received power is impacted by many factors, such as transmitted power, antenna gain and distance between the transmitter and the receiver. It ideally suggests that the electromagnetic wave spreads to all around based on the transmitter as the center. Thus, the basic idea of this model is that the received power decreases as the square of the distance. The signal power attenuation in a free space can be calculated as:

$$\gamma(d) = \frac{P_r}{P_t} = G_t G_r (\frac{\lambda}{4\pi d})^2 \tag{2}$$

where P_r and P_t are the received power and transmitted power, respectively, G_r and G_t are the receiver antenna gain and transmitter antenna gain, respectively, λ is the wavelength and d is the distance between transmitter and receiver. G_t and G_r are commonly set to 1.

In the edge computing environment, edge servers are attached to base stations. The transmitted power and antenna gain of the base stations are ensured and controlled by 5G network operators like T-Mobile and China Mobile. Thus, in this research, the distance is the key to finding the solution to a distance-aware EUA problem. As illustrated in Figure 2, a user's signal strength is attenuated by the increase in the distance between the user and the edge server. According to Equation 2, the attenuation coefficient can be simplified as:

$$\gamma(d_{i,j}) = (\frac{\xi}{d_{i,j}})^2 \tag{3}$$

where $\gamma(d_{i,j})$ is the attenuation coefficient for the communication between user u_i and edge server s_j. Given the attenuation coefficient, a user's data rate and QoE can be calculated.

Based on the model presented above, the distance-aware EUA problem can be modeled as an Integer Linear Programming (ILP) problem, formulated as follows:

$$\max: \quad \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{l=1}^{q} \gamma(d_{i,j}) \cdot e_l \cdot x_{i,j,l} \tag{4}$$

$$s.t.:$$

$$\sum_{i=1}^{n}\sum_{l=1}^{q} w_l \cdot x_{i,j,l} \leq C_j \tag{5}$$

$$\sum_{j=1}^{m}\sum_{l=1}^{q} x_{i,j,l} \leq 1 \tag{6}$$

$$x_{i,j,l} \in \{0,1\} \tag{7}$$

where $x_{i,j,l}$ is a binary variable indicating whether user u_i is allocated to edge server s_j with QoS level w_l.

The objective function (4) maximizes the total QoE of all the users. Note that the QoE level e_l can be pre-calculated by the predefined QoS level w_l. Constraint (5) indicates that the total computing resources required by the users allocated to an edge server must not exceed the available coputing resources on that edge server. Constraint (6) ensures that each user can only be allocated to one edge server at most.

The ILP model above can be solved with ILP solvers, such as Gurobi[1] or IBM CPLEX Optimizer[2]. The solution is the optimal allocation $f : U \rightarrow S$ defined in Section III-A.

C. Problem Hardness

Based on the optimization model built in Section III-B, we can prove the hardness of the distance-aware EUA problem. To do so, we first introduce a classic NP-hard problem, named Capacitated Facility Location problem (CFLP). Given the facility capacity set C, the facility set F, the demand set R and the cost metrix $Cost$. The CFLP problem can be formulated as follows:

$$\min : \sum_{i\in F}\sum_{j\in R} cost_{i,j} y_{i,j} + \sum_{i\in F} f_i x_i \tag{8}$$

$$s.t. :$$

$$\sum_{i\in F} y_{i,j} = 1 \tag{9}$$

$$\sum_{j\in R} r_j y_{i,j} \leq c_i x_i \tag{10}$$

$$x_i, y_{i,j} \in \{0,1\} \tag{11}$$

where the variable $y_{i,j}$ represents whether the demand r_j is fulfilled by facility i and the variable x_i means whether the facility i is open, while $cost_{i,j}$ is the cost to allocate the demand r_j to the facility i and u_i is the capacity of the facility i.

Now, we demonstrate that the distance-aware EUA problem is $\mathcal{NP}$-hard by proving Theorem 1.

Theorem 1. *The distance-aware EUA problem is $\mathcal{NP}$-hard.*

Proof: Now we prove that CFLP can be reduced to an instance of the distance-aware EUA problem. The reduction is done in the following steps. Firstly, we set the number of QoS levels to 1. Denote the total QoS obtained by the ILP model as QoS_{total}. Then we convert the objective (4) of the distance-aware EUA problem to $\min Q - QoS_{total}$, where $Q \rightarrow \infty$. As there is no any open cost for edge servers in the distance-aware EUA problem, the second part $\sum_{i\in F} f_i x_i$ in (8) can be ignored. Given an instance $CFLP(Cost, R, C, F)$, we can construct an instance $distance - awareEUA(D, U, T, S)$ with the reduction above in polynomial time while $|R| = |U|$ and $|F| = |S|$, where D is the distance matrix. In this case, any solution s satisfying objective (8) and constraint (11) also satisfies objective (4) and (7). Since the users who are not allocated to edge servers would be allocated to the cloud server, the solution s fulfills constraint (9) and constraint (6). In the distance-aware EUA problem, there are multiple kinds of resources required by app users. We can treat those resources as a whole, and it violates the constraint if any resource is over the capacity of that edge server. This way, we can project the constraint (5) to the constraint (10). In conclusion, any solution S satisfies the reduced distance-aware EUA problem if S satisfies the CFLP problem. Thus, the distance-aware EUA problem is $\mathcal{NP}$-hard. ■

D. Heuristic Approach

The density of edge servers is expected to reach up to 50 BSs per km2 in future 5G deployments [12]. Finding the optimal solution to the $\mathcal{NP}$-hard distance-aware EUA problem is intractable in large-scale scenarios. Thus, we propose a heuristic approach named DEUA-H for finding sub-optimal solutions to large-scale distance-aware EUA problems. Algorithm 1 presents the pseudo code.

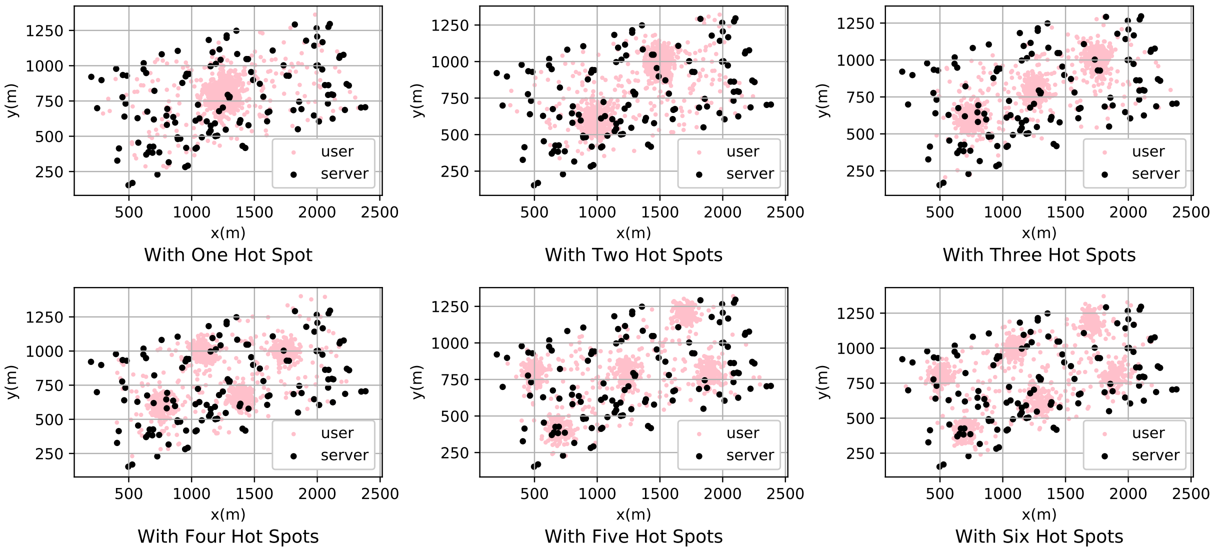
Algorithm 1 DEUA-H

```
Input: edge servers $S$; users $U$.
Output: allocation $f : U \rightarrow S$.
for $i = 1$ to $n$ do
    $S(u_i) \leftarrow \{s_j : c_j \geq w_1\}$
    if $S(u_i) \neq \varnothing$ then
        $j \leftarrow \text{argmax}_j\{\frac{c_j}{d_{i,j}} : s_j \in S(u_i)\}$
        $l \leftarrow \text{argmax}_l\{w_l : w_l \in W, w_l \leq C_j\}$
        allocate user $u_i$ to server $s_j$ with QoS level $l$
    end if
end for
```

DEUA-H goes through three main steps: 1) obtain the set of edge servers that have adequate computing resources to accommodate the user; 2) calculate the ratio of the remaining computing resources to the respective distance for each edge server, and find the edge server which has the highest ratio; 3) select the maximum QoS level the edge server can provide, and allocate the user to that edge server. Take Figure 1 as an example. DEUA-H first selects a users from u_1, u_2, u_3 and u_4, calculates the ratio of each edge server's computing capacity to its distance from the selected user. It then allocates the selected user to the edge server that has the highest ratio with the highest QoS / QoE level this edge server can provide. This process is iterated for all the four users until they are all allocated or there are no computing

[1] http://www.gurobi.com/

[2] https://www.ibm.com/analytics/cplex-optimizer



With One Hot Spot

With Two Hot Spots

With Three Hot Spots

With Four Hot Spots

With Five Hot Spots

With Six Hot Spots

Figure 3. An example of user and edge server distributions with different numbers of hot spots.

resources available on any of the edge servers. The final allocation is the solution to the distance-aware EUA problem presented in Figure 1. Please note that the ratio between each edge server's capacity to the distance from each user can be maintained in a matrix. This matrix can be partially updated in each iteration to reduce the overall computation time.

The time complexity of all these steps is $O(m)$. Hence, the overall time complexity of Algorithm 1 is $O(nm)$, linear to the number of users times the number of edge servers. This indicates the high efficiency of DEUA-H. Its effectiveness will be experimentally evaluated in Section IV.

IV. EXPERIMENTS

In this section, we evaluate DEUA-O and DEUA-H through a series of experiments conducted on a widely-used real-world dataset. All the experiments are conducted on a platform equipped with Intel(R) Xeon(R) Gold 6130 CPU@2.10GHz.

A. Dataset and Experiment Settings

To comprehensively evaluate the effectiveness and efficiency of our approaches, we conduct a series of experiments on the EUA dataset[3]. It contains the locations of base stations in Australia and has been used widely used in research on edge computing [4]–[6], [9], [13]. We select the Melbourne Central Business District (CBD), with an area of $6.2km^2$ to conduct the experiments. There are a total of 125 base stations in this area, which corresponds to 125 edge servers. In addition, users are randomly and unevenly distributed in six ways in this area to simulate different EUA scenarios, as illustrated in Figure 3.

In the experiments, the parameters for existing approaches are tuned to be optimal. The coverage radius of edge server assumed by the existing approaches is set to $150m$. The users within edge servers' coverage areas can achieve a relatively high data rate. For the QoE model presented in SectionIII-B, we set $L = 5$, $\alpha = 1.5$, $\beta = 2$ and the possible QoS levels are set to be $W = \{< 1, 2, 1, 2 >, < 2, 3, 3, 4 >, < \ 5, 7, 6, 6 \ >\}$. For DEUA-O and DEUA-H, we set $\xi = 100m$, and $\gamma(d_{i,j})$ = 1 if $d_{i,j} < 100m$. Under the above settings, we conduct four sets of experiments to compare DEUA-O and DEUA-H with one baseline approach and three state-of-the-art approaches. In each set of experiments, we vary one of the four setting parameters to evaluate its impact on the performance of the approaches. Each time a setting parameter varies, the experiment is repeated for 100 times and the average results over the 100 runs of experiment are reported. The setting parameters are changed in the following ways:

1) User Distribution: We randomly distribute the users in the Melbourne CBD with distance from following the Gaussian distribution $\mathcal{N}(\mu, 50^2)$ with one to six hot spots, as illustrated in Figure 3.

2) Number of Users: The number of users varies from 100, 200 to 1,000 in steps of 100.

3) Number of Edge Servers: A certain percentage of the 125 base stations are randomly selected to be available for allocating users, i.e., 10%, 20%, ..., 100%.

4) Server Capacity: The available computing resources on edge servers also follow a Gaussian distribution, with $\sigma = 1$. The mean for the four types of computing resources, i.e., CPU, RAM, storage and bandwidth, varies from 15, 20, to 60 in steps of 5. Please note that in the experiments the

[3] https://github.com/swinedge/eua-dataset

computing resources are unitized, similar to [4], [5].

B. Competing Approaches and Evaluation Metrics

In the experiments, to demonstrate the performance of EDUA-O and EDUA-H, we compare them with one baseline approach and three state-of-the-art approaches:

- Random: This approach randomly allocates users to available edge servers.
- VSVBP [4]: This approach solves the EUA problem as a variable sized vector bin packing problem, with the aims to maximize the number of allocated users and to minimize the number of edge servers needed.
- DQoS [5]: This approach solves the EUA problem with the consideration of the correlation between users' QoE and required QoS. It aims to maximize the total QoE of all the users, the same as DEUA-O and DEUA-H.

Additionally, we employ three performance metrics to evaluate the approaches, two for effectiveness and one for efficiency:

- *QoE*: measured by users' total QoE produced by the approach, the higher the better.
- *Allocation Rate*: measured by the percentage of users allocated to edge servers, the higher the better.
- *CPU Time*: measured by the computation time taken to find the solution, the lower the better.

C. Results and Discussion

Table II summarizes the results of experiment set #1 where we use dark and light grey to mark the best and second-best value in each column respectively. DEUA-O achieves the highest overall QoE and the second-highest allocation rate among all the five approaches. Its advantages in QoE are significant, 24.60% over DQoS, 42.58% over DEUA-H, 139.02% over VSVBP and 146.80% over Random in user distribution type 1. However, such high performance comes at a price - DEUA-O takes the most time to find a solution. Instead, DEUA-H achieves the third highest overall QoE and the third highest allocation rate. Compared to DEUA-O, its advantage is its high efficiency as the second fastest approach, outperformed by Random only. Interestingly, Random achieves the highest allocation rate and the lowest CPU time. It is because Random does not consider users' QoE and randomly assign QoS levels to users. As users become more evenly distributed (from user distribution type 1 to 6), DEUA-H achieves higher overall QoE, outperforming DQoS in user distribution type 6.

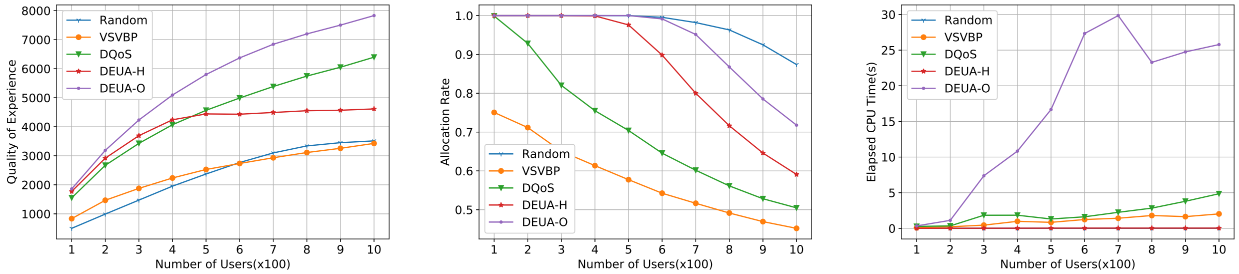
Figure 4 shows the results of experiment set #2, where the number of users increases. Figure 4(a) shows that as the number of users increases from 100 to 1,000, the total QoE achieved by every approach increases. DEUA-O achieves the most significant increase, by 5,972 from 1,859 to 7,831, outperforming DQoS's 4,847 increase from 1,551 to 6,398, DEUA-H's 2,842 increase from 1,773 to 4,615, VSVBP's 2,592 increase from 835 to 3,427 and Random's 3,014 from 501 to 3,515. As the number of users increases, the advantages of DEUA-O over the other approaches increase, indicated by the increasing gaps between them. DEUA-O achieves the second highest allocation rate, as shown in Figure 4(b), outperformed by Random. However, its advantages over the other approaches in allocation rate are also significant, i.e., by 9.61% over DEUA-H, 36.13% over DQoS an 63.32% over VSVBP on average. As the number of users increases, it is harder for all five approaches to allocate all the users, resulting in the decreases in their allocation rates, from 100.00% to 87.38% by 12.62% for Random, 100.00% to 71.78% by 28.22% for DEUA-O, 100.00% to 59.09% by 40.91% for DEUA-H, 99.88% to 50.48% by 49.46% for DQoS and 75.05% to 45.21% by 66.00% for VSVBP. Figure 4(a) shows that when the number of users is relatively small (between 100 and 400), DEUA-H achieves higher overall QoE than DQoS. However, as the number of users exceeds 400, DEUA-H is outperformed by DQoS. The rationale for that is the greedy heuristic employed by DEUA-H allows it to achieve promising results when the computing resources are ample (when the number of users is small). Something similar is observed in the results reported in [5]. The outstanding performance of DEUA-O in achieving high QoE and high allocation rates comes at the price of high computational overhead. As shown in Figure 4(c), DEUA-O takes much more time than the other approaches to find a solution. Its computation time increases significantly as the problem scales up in the number of users. This is expected because of the $\mathcal{NP}$-hardness of the DEUA problem proven in Section III-C. DEUA-H, the approach designed for finding sub-optimal solutions, are almost as fast as Random, outperforming DEUA-O by 99.58%, DQoS by 98.34%, and VSVBP by 96.73% on average. This is its most significant advantage over DQoS. An interesting phenomenon in Figure 4(c) is that, as the number of users exceeds 700, DEUA-O's computation starts to decreases. The reason is that the edge servers cannot accommodate too many users. A lot of users cannot be allocated. They are simply allocated to the remote cloud. This lowers the difficulty for DEUA-O to find a solution. However, this does not significantly help with DEUA-O's low efficiency in finding optimal solutions to large-scale distance-aware EUA problems.

Figures 5 and 6 show the results of experiment set #3 and set #4. The results are similar in general. In both experiment sets, DEUA-O again achieves the highest overall QoE and the highest allocation rate. DEUA-H achieves similar overall QoE and higher allocation rate compared with DQoS. Figure 5(a) shows that, as the number of edge servers exceeds 60, DEUA-H starts to outperform DQoS with an increasing advantage. Figure 6 shows something similar. The reason is the same as in Figure 4(a) - DEUA-H is more suitable in scenarios with ample computing resources, i.e., more servers as shown in Figure 5(a) and more available server capacity as shown in Figure 6(a). Figure 5(c) and Figure

Table II

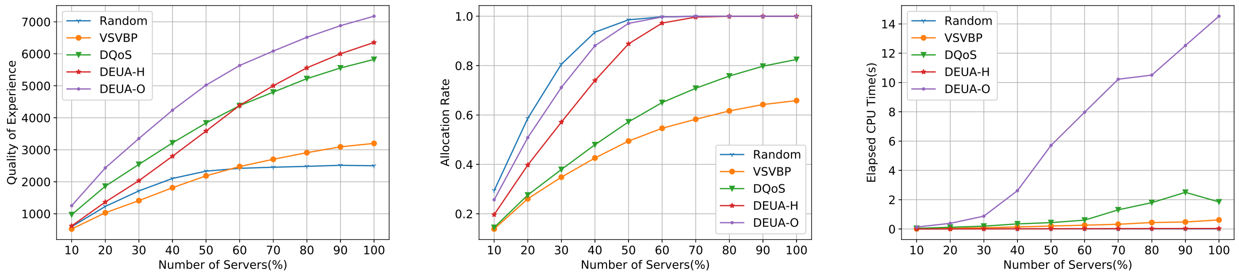
Experiment set #1: Performance results on disparate datasets.

Methods	User Distribution Type 1			User Distribution Type 2			User Distribution Type 3		
	QoE	Allocation Rate	CPU Time	QoE	Allocation Rate	CPU Time	QoE	Allocation Rate	CPU Time
Random	1,658	0.8822	0.0098	2,336	0.9754	0.0104	2,405	1.0000	0.0080
VSVBP	1,712	0.4588	0.8543	2,626	0.6059	0.8539	2,537	0.5778	1.4181
DQoS	3,284	0.5141	1.1749	4,581	0.7560	1.1595	4,581	0.7057	1.2516
DEUA-O	4,092	0.7498	5.3207	5,743	0.9330	18.548	5,799	1.0000	16.737
DEUA-H	2,870	0.6543	0.0194	4,205	0.8320	0.0253	4,443	0.9766	0.0292
Methods	User Distribution Type 4			User Distribution Type 5			User Distribution Type 6		
	QoE	Allocation Rate	CPU Time	QoE	Allocation Rate	CPU Time	QoE	Allocation Rate	CPU Time
Random	2,544	1.0000	0.0075	2,398	0.9986	0.0075	2,562	1.0000	0.0069
VSVBP	3,008	0.6542	1.2206	2,990	0.6541	2.2691	3,264	0.7018	3.0808
DQoS	5,377	0.8405	3.2848	5,507	0.8419	1.5504	6,314	0.9218	5.0674
DEUA-O	6,484	1.0000	21.774	6,572	0.9940	14.580	7,405	0.9966	23.232
DEUA-H	5,051	0.9508	0.0272	5,477	0.9833	0.0289	6,498	0.9962	0.0297



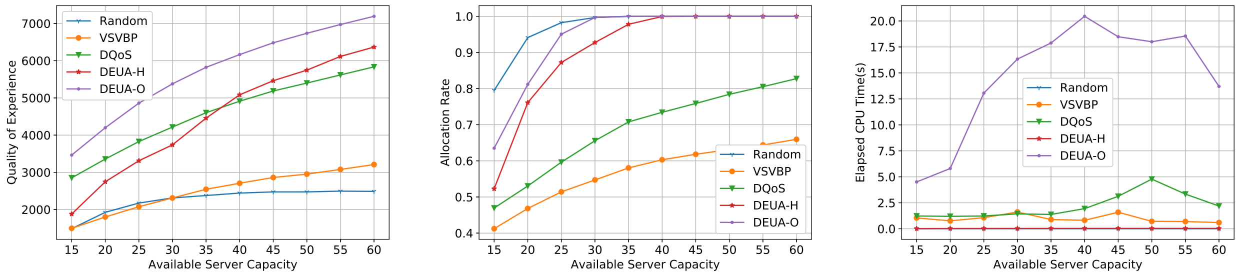
(a) QoE (b) Allocation Rate (c) CPU Time

Figure 4. Experiment set #2: Performance comparisons on various number of users.



(a) QoE (b) Allocation Rate (c) CPU Time

Figure 5. Experiment set #3: Performance comparisons on various number of servers.



(a) QoE (b) Allocation Rate (c) CPU Time

Figure 6. Experiment set #4: Performance comparisons on various server capacities.

6(c) illustrate DEUA-H's high efficiency and high scalability to the problem scale in terms of the number of users and the available server capacity. The performance of DEUA-O demonstrated in Figure 6(c) is similar to Figure 4(c). When the available server capacity exceeds 40, DEUA-O takes less time to find a solution. This is because when the computing resources become ample, all the users can be accommodated by the edge servers. This reduces the complexity of finding the optimal solution and starts the decrease in DEUA-O's computation time. DQoS also exhibits similar changes in its computation time in Figure 6(c).

The experimental results show that by taking into account the distance, our approaches outperform the state-of-the-art approaches in solving the distance-aware EUA problem. Overall, DEUA-O is the most suitable approach in small-scale scenarios for its outstanding performance in achieving high QoS and allocation rates. However, in large-scale scenarios, DEUA-H is the best choice for its effectiveness comparable to DQoS but much higher efficiency.

V. Related Work

With the push from cloud services and pull from the IoT, the edge computing paradigm has received increasingly attention in recent years [2]. The edge computing possesses the advantages of speed, security, cost saving, reliability and scalability, allowing the delivery of new applications and services especially for the future Internet. In terms of Cisco Global Cloud Index, 75% of the data produced by people and devices will be stored, processed, analyzed, and acted upon close to or at the edge of the network by 2021 [14]. As a novel paradigm, edge computing also poses many new challenges for app vendors in allocation-like problems, e.g., edge user allocation [4], [5], [13], [15], edge data caching [16], [17], edge server placement [18] and edge application deployment [19], [20].

Very recently, the edge user allocation problem has attracted a lot of attention. Lai et al. [4] made the first attempt to formulate the EUA problem. They model the EUA problem as a variable sized vector bin packing problem, and propose an approach that maximizes the number of allocated users while minimizing the number of edge servers. Afterwards, they investigated a more sophisticated problem by considering user satisfaction measured by users' Quality of Experience (QoE). Their aim was to maximize all users' overall QoE by assigning them suitable QoS levels [5]. Moreover, Peng et al. targeted at the mobile edge computing environment and modeled the EUA problem as a revolvable process. They propose a greedy algorithm based on the mobility of edge users to find an allocation solution [6]. However, the existing research on EUA ignores the complexity of wireless transmission in real-world EUA scenarios. It is simply assumed that each base station has a specific coverage area and every user in that coverage area will receive the same data rate. The correlation between the signal strength (as well as data rate) and the distance between users and edge servers is neglected. This often lowers users' overall QoE due to unsatisfactory data rates. In contrast, we studied the influence of the distance between users and edge servers on users' data rates and QoE and took it into account in the designs of our optimal approach DEUA-O and sub-optimal approach DEUA-H for solving the distance-aware EUA problem.

VI. Conclusion

In this paper, we studied the distance-aware EUA problem. Specifically, we took into consideration the correlation between the users' signal strength (or data rate) and their distance from edge servers. To solve the distance-aware EUA problem, we proposed two novel approaches, one for finding the optimal solution that maximizes users' overall QoE, and the other for efficiently finding sub-optimal solutions in large-scale scenarios. The results of experiments conducted on a widely-used real-world dataset showed that our approaches significantly outperform the state-of-the-art approaches with their respective advantages. In the future, we plan to focus on the dynamics of the computation tasks and consider the behaviors of users.

Acknowledgment

This work was partially supported by National Key Research and Development Program of China (No. 2017YFC0907505), Shanghai Natural Science Foudation (No. 18ZR1414400), National Natural Science Foudation of China (No. 61772128, 61602109), and Shanghai Sailing Program (No.16YF1400300).

References

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[2] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[3] T. H. Luan, L. Gao, Z. Li, Y. Xiang, G. Wei, and L. Sun, "Fog computing: Focusing on mobile users at the edge," *arXiv preprint arXiv:1502.01815*, 2015.

[4] P. Lai, Q. He, M. Abdelrazek, F. Chen, J. Hosking, J. Grundy, and Y. Yang, "Optimal edge user allocation in edge computing with variable sized vector bin packing," in *International Conferences on Service-Oriented Computing*, 2018, pp. 230–245.

[5] P. Lai, Q. He, G. Cui, X. Xia, M. Abdelrazek, F. Chen, J. Hosking, J. Grundy, and Y. Yang, "Edge user allocation with dynamic quality of service," in *International Conferences on Service-Oriented Computing*, 2019, pp. 86–101.

[6] Q. Peng, Y. Xia, Z. Feng, J. Lee, C. Wu, X. Luo, W. Zheng, S. Pang, H. Liu, Y. Qin, and P. Chen, "Mobility-aware and migration-enabled online edge user allocation in mobile edge computing," in *IEEE International Conference on Web Services*, 2019, pp. 91–98.

[7] I. L. S. Committee *et al.*, "IEEE 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pp. 1–3534, 2016.

[8] T. S. Rappaport, "Wireless communications: principles and practice," *Prentice Hall*, vol. 2, 1996.

[9] X. Xia, F. Chen, Q. He, G. Cui, P. Lai, M. Abdelrazek, J. Grundy, and H. Jin, "Graph-based optimal data caching in edge computing," in *International Conference on Service-Oriented Computing*, 2019, pp. 477–493.

[10] M. Fiedler, T. Hossfeld, and P. Tran-Gia, "A generic quantitative relationship between quality of experience and quality of service," *IEEE Network*, vol. 24, no. 2, pp. 36–41, 2010.

[11] M. Hemmati, B. McCormick, and S. Shirmohammadi, "QoE-aware bandwidth allocation for video traffic using sigmoidal programming," *IEEE MultiMedia*, vol. 24, no. 4, pp. 80–90, 2017.

[12] X. Ge, S. Tu, G. Mao, C.-X. Wang, and T. Han, "5g ultra-dense cellular networks," *IEEE Wireless Communications*, vol. 23, no. 1, pp. 72–79, 2016.

[13] Q. He, G. Cui, X. Zhang, F. Chen, S. Deng, H. Jin, Y. Li, and Y. Yang, "A game-theoretical approach for user allocation in edge computing environment," *IEEE Transactions on Parallel and Distributed Systems*, 2019.

[14] C. V. N. Index, "Forecast and methodology, 2016–2021," *White paper, Cisco public*, vol. 6, 2017.

[15] P. Lai, Q. He, G. Cui, F. Chen, M. Abdelrazek, J. Grundy, J. Hosking, and Y. Yang, "Quality of experience-aware user allocation in edge computing systems: A potential game," in *40th IEEE International Conference on Distributed Computing Systems*. IEEE, 2020.

[16] Y. Liu, Q. He, D. Zheng, M. Zhang, F. Chen, and B. Zhang, "Data caching optimization in the edge computing environment," in *26th IEEE International Conference on Web Services*. IEEE, 2019, pp. 99–106.

[17] X. Xia, F. Chen, Q. He, G. Cui, P. Lai, M. Abdelrazek, J. Grundy, and H. Jin, "Graph-based optimal data caching in edge computing," in *International Conference on Service-Oriented Computing*. Springer, 2019, pp. 477–493.

[18] G. Cui, Q. He, X. Xia, F. Chen, H. Jin, and Y. Yang, "Robustness-oriented k edge server placement," in *2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. IEEE, 2020.

[19] Y. Chen, S. Deng, H. Zhao, Q. He, Y. Li, and H. Gao, "Data-intensive application deployment at edge: A deep reinforcement learning approach," in *26th IEEE International Conference on Web Services*. IEEE, 2019, pp. 355–359.

[20] S. Deng, Z. Xiang, J. Taheri, K. A. Mohammad, J. Yin, A. Zomaya, and S. Dustdar, "Optimal application deployment in resource constrained distributed edges," *IEEE Transactions on Mobile Computing*, 2020.