Digital Communications and Networks(DCN)

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/dcan

Mobility-Driven User-Centric AP Clustering in Mobile Edge Computing Based Ultra Dense Networks

Shuxin He, Tianyu Wang, Shaowei Wang

School of Electronic Science and Engineering, Nanjing University, Nanjing 210000, China

Abstract

Ultra dense network (UDN) has been envisioned as a promising technology to provide high quality wireless connectivity in dense urban areas, in which the density of access points (APs) is increased up to the point where it is comparable with or surpasses the density of active mobile users. In order to mitigate inter-AP interference and improve spectrum efficiency, APs in UDNs are usually clustered into multiple groups to serve different mobile users, respectively. However, as the number of APs increases, the computational capability within an AP group has become the bottleneck of AP clustering. In this paper, we first propose a novel UDN architecture based on mobile edge computing (MEC), in which each MEC server is associated with a user-centric AP cluster to act as a mobile agent. In addition, in the context of MEC-based UDN, we leverage mobility prediction techniques to achieve a dynamic AP clustering scheme, in which the cluster structure can automatically adapt to the dynamic distribution of user traffic in a specific area. Simulation results show that the proposed scheme can highly increase the average user throughput compared to the baseline algorithm using max-SINR user association and equal bandwidth allocation, while at the same time, guarantees low transmission delay.

KEYWORDS:
AP clustering, dynamic user traffic, mobile edge computing, mobility-driven, ultra dense networks.

1. Introduction

The new generation of mobile system, the fifth generation (5G), is expected to accommodate the extreme traffic load in crowded cities and hotspot areas [1]. A large amount of mobile devices will be connected to 5G, which leads to a dramatic growth of traffic demand [2, 3]. In order to address the contradiction between the ever increasing traffic demand of mobile users and limited radio resources, ultra dense networks (UDNs) are recently proposed as a promising technology to improve system capacity by leveraging an extremely dense deployment of access points (APs) [4, 5]. Compared with traditional cellular networks, UDN has an extremely high density of access points (APs) that is comparable with or surpass the density of active mobile users.

From the network-centric perspective, recent studies have shown that UDN can highly improve the system performance in terms of spectral efficiency, energy efficiency, system capacity and interference mitigation. In [6], a joint transmit power control and user scheduling scheme in a multi-cell scenario is proposed to optimize energy efficiency of UDNs, for which a dynamic stochastic game is formulated between small cell base stations and the drift plus penalty approach is utilized in the framework of Lyapunov optimization. In [7], the energy-efficient context-aware resource allocation problem is investigated in UDNs, which is decoupled and then reformulated as a one-to-one matching problem under two-sided preferences. In [8], an optimal design of UDN that balances user mobility and network densification is proposed, where the massive users are divided into different groups according to their moving speeds, and then served by different subnets. It is demonstrated that this approach can meet the traffic demand with high spectrum efficiency. In [9], a novel graph-based multi-cell

*Tianyu Wang (email: tianyu.alex.wang@nju.edu.cn).
[1]Shuxin He (email: mg1723062@smail.nju.edu.cn).
[2]Shaowei Wang (email: wangsw@nju.edu.cn).

scheduling framework is proposed to mitigate downlink inter-cell interference, in which a dynamic clustering method using channel-aware resource allocation is proposed to provide tunable quality of service.

Although network-centric methods have been well studied, the performance of cell-edge users has always been a challenging issue. Since mobile users in UDNs are much closer to access points compared to conventional cellular networks, the number of cell-edge users is greatly increased. Therefore, compared with the conventional network-centric methods, user-centric network management has achieved more attentions in the UDN literature. In [10], a joint AP clustering and resource allocation problem is formulated, and a distributed traffic-aware and user-centric clustering solution with overlapping clusters, is developed to maximize the spectral efficiency. In [11], the transmissions of UDN users are jointly optimized by introducing a number of predefined virtual cells, in which the transmit power of a virtual cell dedicated to each user is limited. A high cooperation gain is obtained by the proposed virtual cell scheme by avoiding inter-AP interference. In [12], a user-centric adaptive clustering method based on local measurements is proposed to maximize the goodput of UDN users using the coordinated multiple point transmission.

However, the the dynamic traffic distribution of UDNs can highly increase the computational burden for both network-centric and user-centric UDNs. Moreover, new service types, such as ultra-high definition video, wearable assistance, and augmented reality, not only require network operators to provide huge traffic support, but also to guarantee strict transmission delay [13, 14]. Therefore, mobile edge computing (MEC) is envisioned as a promising technology for UDNs to act as a powerful mobile agent that can provide high computational capability, large storage and online data analysis. MEC servers can be deployed at the network edge near end users to offload computational tasks and reduce transmission delay [15, 16, 17].

Moreover, current studies mainly focus on static AP clustering methods based on local wireless environment measurement without considering the real-time change of user mobility and traffic distribution. These static methods are facing challenges due to the irregular coverage and multifarious AP relationships of UDNs [18]. In this paper, we integrate a mobile edge computing (MEC) layer into the UDN, where a MEC server is connected to the neighborhood APs to provide integrated communication and computation service for a specific UDN user. In the context of MEC-based UDN, the MEC servers can provide extra computational capability to address real-time user mobility in the network. Therefore, we propose a mobility-driven dynamic user-centric AP clustering scheme to optimize the average user throughput in a MEC-based UDN, which automatically changes the cluster structure according to user mobility prediction and dynamic distribution of user traffic. The main contributions are summarized as follows:

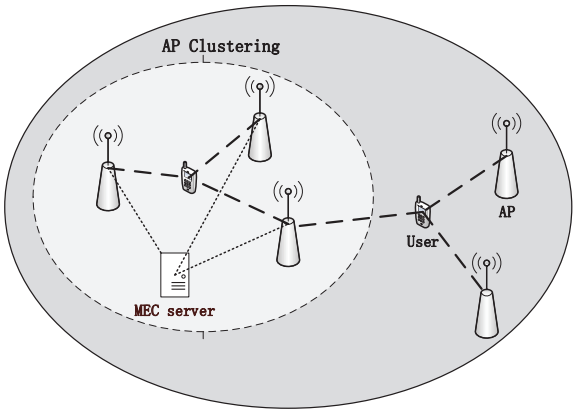

Fig. 1: AP clustering in a MEC-based UDN

- We propose a MEC-based UDN architecture to offload computational tasks from the core network.
- We formulate a $M/M/1/L$ queue to model the traffic of mobile users served by each AP. And we further formulate a combinatorial optimization problem that maximizes the average user throughput.
- The combinatorial optimization problem is solved by using a dynamic user-centric AP clustering scheme. First, a user-centric AP clustering algorithm is designed based on the real-time prediction of user mobility. Then, a resource allocation algorithm is introduced in which the average packet transmission delay of each user is limited.
- The performance of our proposed dynamic user-centric clustering scheme is analyzed and simulation results show that the proposed scheme offers superior clustering performance compared with the baseline algorithm using max-SINR user association and equal bandwidth allocation.

The rest of the paper is organized as follows. In Section II, we introduce the system model and formulate an optimization problem for average user throughput maximization. In Section III, we propose a dynamic user-centric AP clustering scheme. Simulation results are analyzed in Section IV and conclusions are drawn in Section V.

2. System Model

We consider an area served by a MEC-based UDN shown in Fig. 1, in which N APs, the set of which is denoted by $\mathcal{N} = \{1, 2, \ldots, N\}$, and K users, the set of which is denoted by $\mathcal{K} = \{1, 2, \ldots, K\}$, are uniformly distributed in the system. Note that we focus on active mobile users and we assume $N \geq K$. Each AP has a

Table 1: Notations

Notation	Definition
$B_{i,j}$	blocking probability of user j served by AP i
$B_{\mathcal{M}_j}$	blocking probability of user j served by AP group $\mathcal{M}_j$
$\overline{C}$	average user throughput
C_j	throughput of user j
D	threshold of average packet delay
d_0	reference distance
E	average packet size
e_j	packet size of user j
L	buffer size of an AP for a user
$l_{i,j}$	association indicator of user j and AP i
$\mathcal{M}_j$	set of APs serving user j
$\mathcal{N}$	set of APs
N_0	noise power spectral density
$\mathcal{K}$	set of users
$\mathcal{K}_i$	set of users associated with AP i
P	power budget
P_0	power allocated for a user
$r_{i,j}$	transmission rate between user j and AP i
W	bandwidth budget
$w_{i,j}$	bandwidth allocated for user j from AP i
α	path loss parameter
$\mu_{i,j}$	service rate between user j and AP i
$\rho_{i,j}$	queuing parameter of user j
$\tau_{i,j}$	average packet delay of user j
λ_j	packet arrival rate of user j

constant power and bandwidth budget, denoted by P and W, respectively.

We introduce the association indicator $l_{i,j}$ to indicate whether user j is served by AP i,

$$l_{i,j} = \begin{cases} 1 & \text{user } j \text{ is associated with AP } i, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

The set of users associated with AP i is given by,

$$\mathcal{K}_i = \{j \mid l_{i,j} = 1, j \in \mathcal{K}\}, \forall i \in \mathcal{N}. \tag{2}$$

The number of users associated with AP i is then given by $K_i = |\mathcal{K}_i|$. For any user $j \in \mathcal{K}$, we assume it can be served by a group of M APs and M is a predefined constant. The AP group that serves user j is then given by,

$$\mathcal{M}_j = \{i \mid l_{i,j} = 1, i \in \mathcal{N}\}, \forall j \in \mathcal{K}. \tag{3}$$

We assume that each AP uses a constant power P_0 for each user and the bandwidth allocated to user j by AP i is denoted by $w_{i,j}$. The power budget constraint of AP i is then given by,

$$P_0 K_i \le P, \forall i \in \mathcal{N}. \tag{4}$$

And the bandwidth budget constraint of AP i is given by,

$$\sum_{j \in \mathcal{K}_i} w_{i,j} \le W, \forall i \in \mathcal{N}. \tag{5}$$

The downlink transmission rate between user j and AP i is then given by,

$$r_{i,j} = w_{i,j} \log\left[1 + \frac{P_0 (d_0/d_{i,j})^\alpha}{N_0 w_{i,j}}\right], \tag{6}$$

where d_0 is the reference distance, $d_{i,j}$ is the distance between user j and AP i, α is the pathloss exponent, and N_0 is the noise power spectral density. Here, we consider the downlink transmission without inter-AP interference.

We assume that the arrival of data packets for any user j is a Poisson process with arrival rate λ_j, and the AP serves each user with a limited buffer size L. The packet size of user j, denoted by e_j, follows a negative exponential distribution with mean E. If user j is associated with AP i, i.e., $l_{i,j} = 1$, we have the service rate $\mu_{i,j}$ given by,

$$\mu_{i,j} = \frac{r_{i,j}}{e_j} = \frac{w_{i,j}}{e_j} \log\left[1 + \frac{P_0 (d_0/d_{i,j})^\alpha}{N_0 w_{i,j}}\right], \tag{7}$$

Based on the above assumptions, we can formulate the serving process of APs for a user as a $M/M/1/L$ queuing process [19], where the first M indicates the arrival of user traffic follows a Poisson process with the parameter λ_j, the second M indicates the service time follows a negative exponential distribution with the parameter $\mu_{i,j}$, and L is the limited buffer of the AP for an associated user.

When a packet of user j arrives at AP i, the probability that there are n packets waiting in the buffer is given by,

$$\pi_{i,j}(n) = \rho_{i,j}^n \frac{1 - \rho_{i,j}}{1 - \rho_{i,j}^{L+1}}, \tag{8}$$

where

$$\rho_{i,j} = \frac{\lambda_j}{\mu_{i,j}} = \frac{\lambda_j e_j}{w_{i,j} \log\left[1 + \frac{P_0 (d_0/d_{i,j})^\alpha}{N_0 w_{i,j}}\right]}, \tag{9}$$

Note that $\rho_{i,j} < 1$ must always be satisfied.

When a new packet of user j arrives, if there are n packets waiting in the queue and $n < L$, the service time of total $n+1$ packets is given by $(n+1)/\mu_{i,j}$. And the average packet delay is equivalent to the average service time of $n+1$ packets in the queue, which is given by,

$$\begin{aligned} \tau_{i,j} &= \sum_{n=0}^{L-1} \pi_{i,j}(n) \frac{(n+1)}{\mu_{i,j}} \\ &= \frac{(1 - \rho_{i,j})}{(1 - \rho_{i,j}^{L+1}) \mu_{i,j}} \sum_{n=0}^{L-1} (n+1) \rho_{i,j}^n \\ &= \frac{1}{(1 - \rho_{i,j}^{L+1}) \mu_{i,j}} \left[\frac{1 - \rho_{i,j}^L}{1 - \rho_{i,j}} - L \rho_{i,j}^L\right]. \end{aligned} \tag{10}$$

We see that the average packet delay is decided by bandwidth allocation of AP i for its associated users from (9) and (10). And the packet delay constraint for user j is then given by,

$$\tau_{i,j} \leq D. \tag{11}$$

When a new packet of user j arrives, if the buffer is full, i.e. there are already L packets in the queue, the packets of user j would be dropped. Therefore, the blocking probability of AP i for user j, denoted by $B_{i,j}$, is given by,

$$B_{i,j} = \pi_{i,j}(L) = \rho_{i,j}^{L} \frac{1-\rho_{i,j}}{1-\rho_{i,j}^{L+1}}. \tag{12}$$

When user j is served by AP group $\mathcal{M}_j$, the packets of user j will enter the buffer of each AP in the group. Therefore, the arriving packets would be blocked only if all the buffers in the group are full. Thus, the blocking probability of user j served by AP group $\mathcal{M}_j$, denoted by $B_{\mathcal{M}_j}$, is given by,

$$B_{\mathcal{M}_j} = \prod_{i \in \mathcal{M}_j} B_{i,j}. \tag{13}$$

Then the throughput of user j is given by,

$$C_j = \lambda_j (1 - B_{\mathcal{M}_j}). \tag{14}$$

And the average user throughput in the system, denoted by $\overline{C}$, is given by,

$$\overline{C} = \frac{1}{K} \sum_{j \in \mathcal{K}} C_j. \tag{15}$$

We aim to optimize the AP group for each user to maximize the average user throughput in the system. Therefore, the optimization problem is formulated as follows:

$$\max_{\{l_{i,j}\},\{w_{i,j}\}} \overline{C} \tag{16a}$$

$$\text{s.t. } P_0 K_i \leq P, \forall i \in \mathcal{N}, \tag{16b}$$

$$\sum_{j \in \mathcal{K}_i} w_{i,j} \leq W, \forall i \in \mathcal{N}, \tag{16c}$$

$$l_{i,j} \in \{0,1\}, \forall i \in \mathcal{N}, j \in \mathcal{K}, \tag{16d}$$

$$\sum_{i \in \mathcal{N}} l_{i,j} = M, \forall j \in \mathcal{K}, \tag{16e}$$

$$\tau_{i,j} \leq D, \forall i \in \mathcal{N}, j \in \mathcal{K}, \tag{16f}$$

$$\rho_{i,j} < 1, \forall i \in \mathcal{N}, j \in \mathcal{K}. \tag{16g}$$

(16b) is the constraint of total power of an AP, and (16c) is the constraint of total bandwidth of an AP. (16d) indicates two states of user association between user j and AP i, and (16e) indicates each user is served by M APs. (16f) is the packet delay constraint and D is a predefined constant. (16g) is the constraint of the queuing system. We summarize the notations in Table 1.

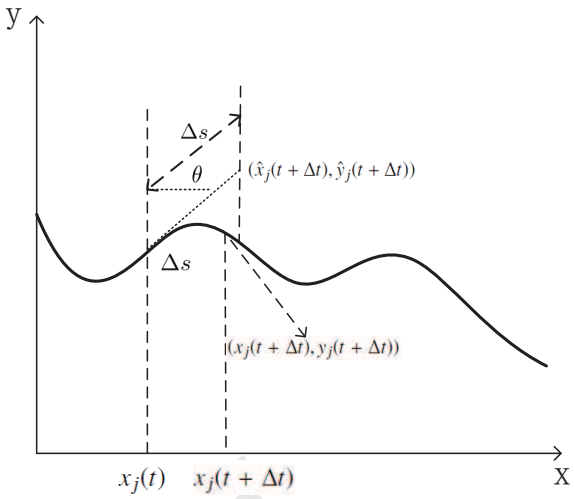

Fig. 2: Mobility Prediction of User j.

3. Dynamic User-Centric AP Clustering Scheme

Problem (16) is a combinatorial integer optimization problem, which is NP-hard in general. Here, we develop a heuristic algorithm consisting of user-centric AP clustering based on mobility prediction and greedy resource allocation with packet delay constraint.

3.1. AP Clustering Based on Mobility Prediction

From (9) and (12) we see that the average blocking probability $B_{i,j}$ increases with the distance between AP i and user j as the derivative of the blocking probability $B_{i,j}$ with respect to the distance $d_{i,j}$ is always above zero. Therefore, the idea of our algorithm is to cluster the APs close to user j into a group to reduce the average blocking probability of the traffic packet.

As shown in Fig. 2, we firstly generate the mobility trajectory of user j into a polynomial function $y = f_j(x)$ according to the user mobility model shown in [20, 21], where the location of user j is denoted by (x_j, y_j). We denote $(x_j(t), y_j(t))$ as the location coordinate of user j at time t. The initial location of user j is then denoted by $(x_j(0), y_j(0))$, and the initial AP group serving for user j is denoted by $\mathcal{M}_j(0)$.

We denote Δt as the prediction duration and trace the mobility of user j for every Δt. We assume that users move with a constant speed v, and the prediction distance Δs is then given by,

$$\Delta s = v \Delta t. \tag{17}$$

We further assume that the moving direction is tangent to the mobility trajectory $y = f_j(x)$, and the moving direction angle at time t is then given by,

$$\theta(t) = \arctan \frac{d f_j(x)}{dx}\Big|_{x=x_j(t)}. \tag{18}$$

Then the location of user j at time $t + \Delta t$, denoted by $(\hat{x}_j(t+\Delta t), \hat{y}_j(t+\Delta t))$, can be predicted by,

$$\hat{x}_j(t+\Delta t) = x_j(t) + \Delta s \cos\theta(t);$$
$$\hat{y}_j(t+\Delta t) = y_j(t) + \Delta s \sin\theta(t). \quad (19)$$

For any time instance t, we denote by $d_{j,i}(t)$ as the predicted distance between AP i and user j. For any user j, we sort the distances between user j and APs in descending order, i.e.,

$$d_{j,i_1}(t) < d_{j,i_2}(t) < \cdots < d_{j,i_N}(t). \quad (20)$$

We select the first M APs as the group serving user j, the set of which is given by

$$\mathcal{M}_j(t) = \{i_1, i_2, \ldots, i_M\}. \quad (21)$$

The AP group $\mathcal{M}_j(t)$ provides service for user j from time t to $t+\Delta t$. For any user j, a MEC server is connected to all the APs in the group $\mathcal{M}_j(t)$ to offload the computation tasks generated by user j.

When the packets of user j arrive with a rate $\lambda_j(t)$, the packets enter the buffers in the group $\mathcal{M}_j$ for service. The data packets will be blocked when all the APs' buffers in the group are full.

Moreover, the actual location $(x_j(t+\Delta t), y_j(t+\Delta t))$ on the mobility trajectory $f_j(x)$, can be calculated from

$$\int_{x_j(t)}^{x_j(t+\Delta t)} \sqrt{1+(\frac{df_j(x)}{dx})^2} = \Delta s. \quad (22)$$

The members of the AP group serving for user j are updated dynamically at each point where the mobility prediction is performed. Specifically, at each prediction point, users update their location, and send the location information to the MEC servers connected to their serving AP groups. Then the MEC servers perform the prediction of user location and inform users of the regrouped APs to provide service over the next prediction interval. Finally, the AP groups to serve the users are updated. In practical mobile communication systems, the APs within a group can exchange information with each other by the designed $X2$ interface [22]. The proposed AP clustering algorithm is shown in Table 2.

Table 2: AP Clustering Based on Mobility Prediction

Algorithm

Initiate $t = 0$, $\mathcal{M}_j = \mathcal{M}_j(0)$;
for $j = 1 : K$
  generate user mobility trajectory $y = f_j(x)$;
end for
while $t < T$
  do
  for $j = 1 : K$
    generate user traffic $\lambda_j(t)$;
    update $(\hat{x}_j(t), \hat{y}_j(t)$ by (19);
    for $i = 1 : N$
      calculate $d_{j,i}(t)$;
    end for
    decide AP group $\mathcal{M}_j(t)$;
  end for
  update $(x_j(t), y_j(t))$ by (22);
  $t = t + \Delta t$;
end while
return $\mathcal{M}_j$;

3.2. Resource Allocation with Delay Constraint

For any AP i and user $j \in \mathcal{K}_i$, we denote by $w_{i,j}^{min}$ as the minimum required bandwidth satisfying the delay constraint (16f), which is given by,

$$w_{i,j}^{min} = \min\{w_{i,j} | \tau_{i,j}(w_{i,j}) \le D\}. \quad (23)$$

The remained bandwidth of AP i, denoted by W_i^r, is given by,

$$W_i^r = W - \sum_{j \in \mathcal{K}_i} w_{i,j}^{min}, \forall i \in \mathcal{N}. \quad (24)$$

Table 3: Resource Allocation with Delay Constraint

Algorithm

Initiate $w_{i,j} = 0$, $j \in \mathcal{K}_i$, $W_i^r = W$;
for $i = 1 : N$
  for $j \in \mathcal{K}_i$
    calculate $w_{i,j}^{min}$ by (23);
  end for
  calculate W_i^r by (25);
  $w_{i,j} = w_{i,j}^{min}$;
  for $u = 1 : U_i$
    calculate j^* by (26);
    $w_{i,j} = w_{i,j} + w_u$;
  end for
end for
return $w_{i,j}$;

Then we divide the remained bandwidth of AP i into equal bandwidth units. Each bandwidth unit is denoted by w_u. The number of the remained bandwidth units of AP i is then given by,

$$U_i = \frac{W_i^r}{w_u}, \forall i \in \mathcal{N}. \quad (25)$$

In each iteration, one bandwidth unit is allocated to the user with the maximum packet delay, and the selected user is obtained by,

$$j^* = \underset{j \in \mathcal{K}_i}{\operatorname{argmax}}\, \tau_{i,j}, \forall i \in \mathcal{N}. \quad (26)$$

The iteration terminates after the remained U_i bandwidth units are used up. The resource allocation with delay constraint algorithm is shown in Table 3.

Table 4: Simulation Parameters

$P = 4$ W	Power budget of an AP
$W = 1$ MHz	Bandwidth budget of an AP
$P_0 = 0.4$ W	Received power at a user
$D = 200$ ms	Delay threshold
$d_0 = 1$ m	Reference distance
$\alpha = 3.76$	Path loss parameter
$N_0 = -184$ dBm/Hz	Noise power spectral density
$E = 10^3$ bit	Average packet size
$\lambda = 100$	Average packet arrival rate
$\delta^2 = 10$	Variance of packet arrival rate
$T = 60$ s	Observation time

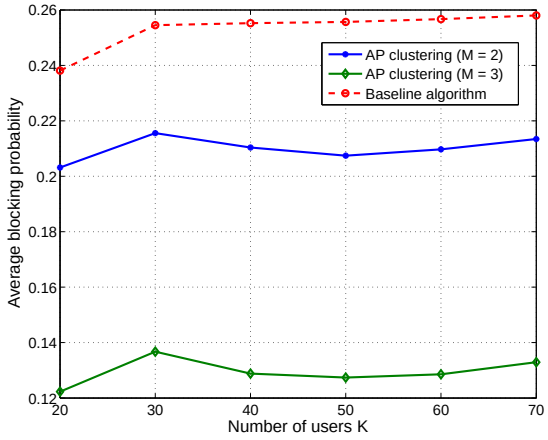

Fig. 3: Average blocking probability as a function of the number of users.

4. Simulation Results

We consider a square area with 2 km side length, where APs and users are uniformly distributed within the area. In traditional cellular networks, users are usually associated with only one base station which provides the max-SINR due to the limitation of the density of APs. Therefore, for comparison, given the real location on the mobility trajectory, the baseline algorithm employs the max-SINR user association without clustering and equal bandwidth allocation. The user reassociation is performed for every 1s in the baseline algorithm.

We assume that an AP transmits to its associated users with the total power of 4 W and the total bandwidth of 1 MHz. Each AP transmits to a user with a constant power 0.4 W. The packet delay is constrained by the threshold of 20 ms. The packet arrival rate λ_j follows a Gaussian distribution with mean $\lambda = 100$, and variance $\delta^2 = 10$. We assume the buffer size of the AP for a user is equal and the user velocity is 2 m/s, and then present the performance of the proposed algorithm compared with the baseline algorithm. We summarize the simulation parameters in Table 4.

In Fig. 3, we show the average blocking probability as a function of the number of users, where the number of APs $N = 100$, the prediction duration $\Delta t = 1$ s and the AP buffer size $L = 20$. As we can see, our proposed algorithm outperforms the baseline algorithm by a 20% and 50% decrease of the blocking probability with the cluster size $M = 2$ and $M = 3$, respectively. When the number of users is above 30, we can see that the blocking probability first decreases a bit and then increases in the proposed algorithm. As the number of users increases, on the one hand, the distance between users and their serving APs decreases, and the clustered AP groups can provide better service so as to reduce the blocking probability. On the other hand, the average waiting time of the packets in the queue increases, which increases the blocking probability. When the number of users is from 30 to 50, the decrease of the distance between users and their serving APs predominates, and the blocking probability first decreases a bit. When the number of users is above 50, the increase of the waiting time in the queue predominates, and the average blocking probability then increases.

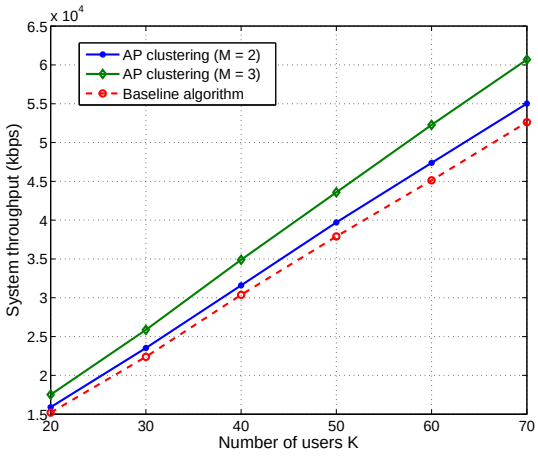

Fig. 4: System throughput as a function of the number of users.

In Fig. 4, we further show the system throughput as a function of the number of users, where the number of APs $N = 100$, the prediction duration $\Delta t = 1$ s and the AP buffer size $L = 20$. As we can see, the system throughput increases rapidly as the number of users increases in both algorithms. However, our proposed algorithm improves the total throughput by a 20% and 30% gain with the cluster size $M = 2$ and $M = 3$, respectively. As shown earlier in Fig. 3, when the number of users increases, the proposed algorithm shows a lower blocking probability compared with the baseline algorithm. Therefore, the proposed algorithm provides a throughput gain compared with the baseline algorithm.

In Fig. 5, we show the average packet delay as a function of the number of users, where the number of APs $N = 100$, the prediction interval $\Delta t = 1$ s and the buffer size $L = 20$. As we can see, the average packet delay of users increases as the number of users

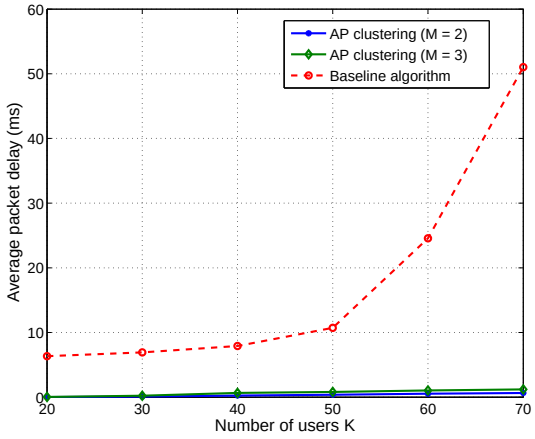

Fig. 5: Average packet delay as a function of the number of users.

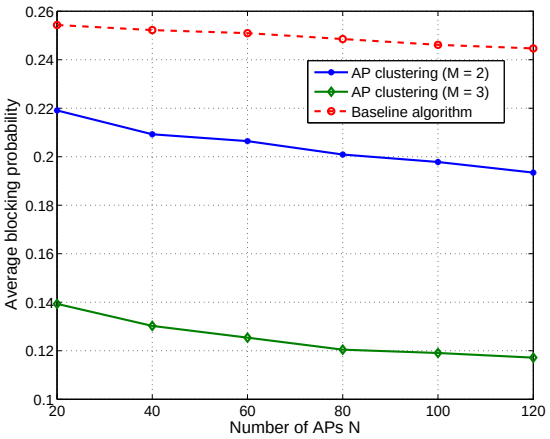

Fig. 6: Average blocking probability as a function of the number of APs.

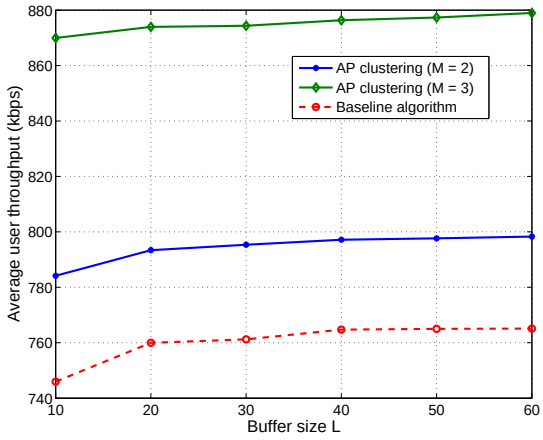

Fig. 7: Average user throughput as a function of the buffer size of an AP.

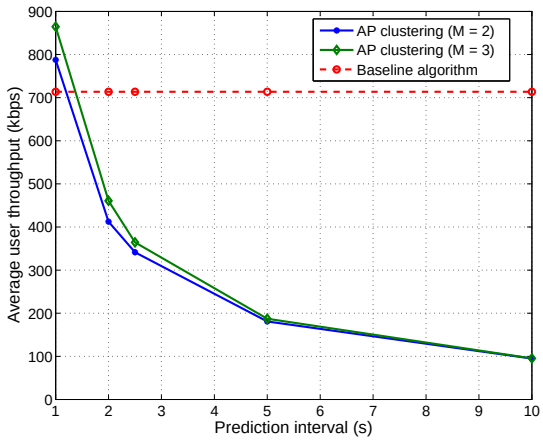

Fig. 8: Average user throughput as a function of the prediction duration, $N = 100$, $K = 20$, $L = 20$.

increases, since the average waiting time of the packets in the queue is increased. The proposed algorithm outperforms the equal bandwidth allocation algorithm by a 60% and 70% decrease of the average packet delay with the cluster size $M = 2$ and $M = 3$, respectively. Moreover, as the number of users increases, the proposed algorithm can guarantee the average packet delay of the users under a certain threshold 20 ms, while the packet delay continuously increases in the baseline algorithm.

In Fig. 6, we show the average blocking probability of the user packets as a function of the number of APs, where the number of users $K = 20$, the prediction interval $\Delta t = 1$ s and the buffer size $L = 20$. As we can see, the average blocking probability decreases as the number of APs increases, since a larger number of APs allow more users to access to the network, and the distances between the users and their serving APs are decreased. Moreover, the proposed algorithm outperforms the baseline algorithm by a 20% and 50% decrease of the blocking probability with the cluster size $M = 2$ and $M = 3$, respectively. It indicates that the proposed algorithm would perform better than the baseline algorithm on data transmission during the process of network densification, which is significant for the deployment of future mobile networks.

In Fig. 7, we show the average user throughput as a function of the buffer size, where the number of APs $N = 100$, the number of users $K = 20$ and the prediction interval $\Delta t = 1$ s. As we can see, the average user throughput increases as the buffer size increases since a larger buffer size means that the APs allow more data packets to be served. However, a marginal increase of the average user throughput is achieved when the buffer size is above 40. We can see that $L = 40$ is the optimal point where we formulate the queuing model, since a marginal performance gain can be obtained at the cost of a much larger buffer. Additionally, the proposed algorithm improves the average user throughput by a 20% and 25% gain compared to the baseline algorithm with the cluster size $M = 2$ and $M = 3$, respectively.

In Fig. 8, we show the average user throughput as a

function of the prediction interval, where the number of APs $N = 100$, the number of users $K = 20$ and the buffer size $L = 20$. As we can see, the average user throughput decreases rapidly as the prediction interval increases by using the proposed algorithm. Moreover, when the prediction interval is below 1*s*, the proposed algorithm shows a 20% and 30% enhancement of the user throughput compared with the baseline algorithm with the clustering size $M = 2$ and $M = 3$, respectively. However, when the prediction interval is above 1 s, it can be observed that our proposed algorithm shows a worse performance compared to the baseline algorithm, since the prediction of user mobility will be not accurate with a larger prediction distance. In contrast, if the prediction interval is below 1 s, the clustered AP groups will be frequently updated, which will increase the signalling burden of the core network. Thus, the optimal prediction interval is $\Delta t = 1$ s.

5. Conclusions

In this paper, we have investigated the user-centric AP clustering problem in a MEC-based UDN. A throughput maximization problem was formulated, where the service for a user by an AP was modeled as a $M/M/1/L$ queuing process. The problem was then solved by a user-centric AP clustering algorithm and a greedy bandwidth allocation algorithm. Simulation results show that the proposed scheme not only increases the average user throughput compared to the baseline algorithm using max-SINR user association and equal bandwidth allocation, but also guarantees low packet transmission delay. We find that the density of APs, prediction duration and cluster size have significant impacts on the performance of AP clustering, which requires careful design for practical UDN deployment.

References

[1] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong and J. C. Zhang, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065-1082, Jun. 2014.

[2] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 74-80, Feb. 2014.

[3] S. Z. Chen and J. Zhao, "The Requirements, Challenges, and Technologies for 5G of Terrestrial Mobile Telecommunication," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 36-43, May 2014.

[4] A. Gotsis, S. Stefanatos and A. Alexiou, "Ultra Dense networks: The new wireless frontier for enabling 5G access," *IEEE Veh. Technol. Mag.*, vol. 11, no. 2, pp. 71-78, Jun. 2016.

[5] M. Kamel, W. Hamouda and A. Youssef, "Ultra-Dense Networks: A Survey," *IEEE Commun. Surveys Tuts.*, no. 4, vol. 18, pp. 2522-2545, May. 2016.

[6] S. Samarakoon, M. Bennis, W. Saad, M. Debbah and M. Latva-Aho, "Energy-efficient resource management in ultra dense small cell networks: A mean-field approach," in *Proc. IEEE GLOBECOM*, San Diego, CA, USA, Dec. 2015.

[7] Z. Zhou, M. Dong, K. Ota and Z. Chang, "Energy-efficient context-aware matching for resource allocation in ultra-dense small cells," *IEEE Access*, vol. 3, pp. 1849-1860, Sep. 2015.

[8] J. Zhu, M. Zhao and S. Zhou, "An Optimization Design of Ultra Dense Networks Balancing Mobility and Densification," *IEEE Access*, vol. 6, pp. 32339-32348, Jun. 2018.

[9] E. Pateromichelakis, M. Shariat, A. Quddus, M. Dianati and R. Tafazolli, "Dynamic clustering framework for multi-cell scheduling in dense small cell networks," *IEEE Commun. Lett.*, vol. 17, no. 9, pp. 1802-1805, Sep. 2013.

[10] Y. Lin, R. Zhang, C. Li, L. Yang and L. Hanzo, "Graph-Based Joint User-Centric Overlapped Clustering and Resource Allocation in Ultradense Networks," *IEEE Wireless Commun.*, no. 5, vol. 67, pp. 4440-4453, May. 2018.

[11] Y. Zhao, S. Bi and Y. A. Zhang, "User-Centric Joint Transmission in Virtual-Cell-Based Ultra-Dense Networks," *IEEE Trans. Veh. Technol.*, no. 5, vol. 67, pp. 4640-4644, May. 2018.

[12] V. Garcia, Y. Zhou and J. Shi, "Coordinated multipoint transmission in dense cellular networks with user-centric adaptive clustering," *IEEE Trans. Wireless Commun.*, vol. 13, no. 8, pp. 4297-4308, Aug. 2014.

[13] M. Assuncao, A. Veitha, and R. Buyyab, "Distributed data stream processing and edge computing: A survey on resource elasticity and future directions,"*Digit. Commun. Netw.*, vol. 103, pp. 77-86, Feb. 2018.

[14] J. Li, C. Shunfeng, F. Shu, J. Wu and D. Jayakody, "Contract-Based Small-Cell Caching for Data Disseminations in Ultra-Dense Cellular Networks," *IEEE Trans. on Mobile Computing.*, doi. 10. 1109/TMC.2018.2853746

[15] Q. La, M. Ngo, T. Dinh, T. Quek and H. Shin, "Enabling intelligence in fog computing to achieve energy and latency reduction,"*Digit. Commun. Netw.*, vol. 5, no. 1, pp. 3-9, 2019.

[16] Y. Ai, M. Peng and K. Zhang, "Edge computing technologies for Internet of Things: A primer," *Digit. Commun. Netw.*, vol. 4, no. 2, pp. 77-86, 2018.

[17] P. Wang, Z. Zheng, B. Di and L. Song, "HetMEC: Latency-optimal Task Assignment and Resource Allocation for Heterogeneous Mobile Edge Computing," https://arxiv.org/abs/1901.09307.

[18] S. Chen, F. Qin, B. Hu, X. Li and Z. Chen, "User-Centric Ultra-Dense Networks for 5G: Challenges, Methodologies, and Directions," *IEEE Wireless Commun.*, no. 2, vol. 23, pp. 78-85, Apr. 2016.

[19] I. Adan and J. Resing, *Queueing Theory*, Eindhoven University of Technology, Feb. 2001.

[20] M. Kim, D. Kotz and S. Kim, "Extracting a mobility model from real user traces," in *Proc. IEEE INFOCOM*, Barcelona, Spain, Apr. 2006.

[21] S. Chatterjee, D. Sarddar and J. Saha, "An Improved Mobility Management Technique for IEEE 802.11 based WLAN by Predicting the Direction of the Mobile Node," in *Proc. IEEE National Conf. Comput. Commun. Syst.*, Durgapur, India, Nov. 2012.

[22] X2 general aspects and principles, *3GPP TS 36.420, Version 12.1.0*, Feb. 2015.