A Multi-tier Cost Model for Effective User Scheduling in Fog Computing Networks

Zening Liu*†, Yang Yang*, Yu Chen‡, Kai Li*, Ziqin Li*, Xiliang Luo*

*Shanghai Institute of Fog Computing Technology (SHIFT), SIST, ShanghaiTech University, China
†University of Chinese Academy of Sciences, China
‡School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, China

***Abstract*—In this paper, we investigate a cost model and the resulted cost-minimization user scheduling problem in multi-tier fog computing networks. For a typical multi-tier fog computing network consisting of one fog control node (FCN), multiple fog access nodes (FANs) and user equipments (UEs), how to model the cost paid to FANs for motivating resources sharing, and how to effectively schedule UEs to minimize the overall cost for FCN, are still problems to be resolved. To solve these problems, a unified multi-tier cost model, including the service delay and a linear inverse demand dynamic payment scheme, is proposed, and a cost-minimization user scheduling problem is formulated. Further, the user scheduling problem is reformulated as a potential game and proven to possess a nash equilibrium (NE) solution. The corresponding distributed algorithm, namely Cost-Oriented User Scheduling (COUS) algorithm, is developed to achieve an NE of the user scheduling game. Analytical and simulation results show that the COUS can offer near-optimal performance in terms of the overall cost. Besides, the dynamic payment scheme can achieve a win-win outcome for both FANs and FCN, but an unfair workload distribution among FANs, compared with the fixed payment scheme.**

***Index Terms*—Fog computing, cost model, user scheduling, potential game.**

I. Introduction

With the explosion of smart devices and the popularity of low-latency applications, such as streaming of videos, current wireless networks have been suffering from the unprecedented data traffic burst and stringent demands on service delay. To cope with this challenge, fog computing has emerged as a promising architecture for Internet of Things (IoT) and future wireless networks [1]. Fog computing shifts part of the communication, computation, and caching resources from the remote cloud to the network edge, along the cloud-to-things continuum. It empowers end user equipments (UEs) with multi-tier computing or service [2], [3]. In such an architecture, data can be processed, or services can be provided, flexibly at different tiers, which are closer to UEs. Thus, both the traffic load and the service delay can be effectively reduced.

Without loss of generality, let us consider a multi-tier fog computing network consisting of one fog control node (FCN), multiple fog access nodes (FANs) and UEs, as shown in Fig. 1. With the help of FANs, UEs can be served with reduced service delay and enhanced quality of service (QoS). For example, delay-tolerant services can be provided by remote FCN, while delay-sensitive applications can be processed at neighboring FANs. Through effective user scheduling, the traffic load and service delay can be greatly reduced.

A number of research efforts have been dedicated to the user scheduling problem in multi-tier fog computing networks [4]–[7]. Shah-Mansouri *et al.* [4] developed a user scheduling algorithm to maximize each UE's quality of experience, based on potential game theory. Liu *et al.* [5] derived a convex optimization based joint user scheduling and resource allocation algorithm to minimize the total system cost for a cloud-fog computing network with non-orthogonal multiple access. Zhao *et al.* [6] utilized the Lyapunov optimization techniques to design an online joint user scheduling and resource allocation algorithm to maximize the average network throughput for a fog-enabled content delivery network. Liu *et al.* [7] proposed a distributed file placement and user scheduling algorithm for a mobile fog-caching service network, utilizing matching theory.

Although different aspects of user scheduling in multi-tier fog computing networks have been discussed in literatures, effective user scheduling scheme still faces challenges, especially when the cost model is considered. Generally, the FCN is operated by a telecom operator, who signs a service contract with UEs, while the FANs are belongs to different individuals. To better motivate the FANs to share resources and anticipate in caching, the cost model, especially for FANs, should be taken into consideration.

In this paper, a unified multi-tier cost model, including the service delay and a linear inverse demand dynamic payment scheme, and the resulted cost-minimization user scheduling problem, are investigated, in a multi-tier fog computing network consisting of one FCN, multiple FANs and UEs. The main contributions of this paper are summarized as follows.

- A unified multi-tier cost model, including the service delay and a linear inverse demand dynamic payment scheme, is proposed for a multi-tier fog computing network consisting of one FCN, multiple FANs and UEs. Besides, the resulted cost-minimization user scheduling problem is formulated.
- A potential game is formulated to model the cost-minimization user scheduling problem and the existence of nash equilibrium (NE) is proven. Also, the corresponding distributed algorithm, namely Cost-Oriented User

978-1-7281-1878-9/19/$31.00 ©2019 IEEE

Scheduling (COUS) algorithm, is developed to achieve an NE of the game.

- Extensive simulations are conducted to demonstrate the performance of the COUS algorithm and the features of the proposed cost model. The theoretical proofs and simulation results show that the COUS algorithm can offer near-optimal performance in terms of the overall cost, and both the FCN and FANs can benefit from the dynamic payment scheme, compared with the fixed payment scheme. However, the dynamic payment scheme will incur unfair workload distribution among FANs.

The rest of this paper is organized as follows. The system model of multi-tier fog computing networks is given in Section II, together with the mathematical formulation of the cost model and the resulted cost-minimization user scheduling problem. Based on potential game, the user scheduling game is developed and analyzed in Section III. The NE of this game is proven to exist and the corresponding distributed user scheduling algorithm, i.e., COUS algorithm, is proposed. Then, Section IV evaluates the performance of developed algorithm and the features of proposed cost model via simulation. Finally, Section V concludes this paper.

II. System Model and Problem Formulation

A. System Model

As shown in Fig. 1, a multi-tier fog computing network consisting of one FCN, M FANs and N UEs is considered. The FCN is operated by a telecom operator, which provides services to N UEs, i.e., service subscribers, while the FANs are belongs to different individuals. To reduce service delay and improve QoS, the FCN is willing to pay money to FANs if they provide services to UEs. For the ease of expression, we take caching as an example in the following context [1]. In the fog-enabled caching network, the FCN can allocate files to FANs during off-peak time, i.e., file placement, and thus the UEs can be associated with proper FANs or FCN to download files during peak time, i.e., user scheduling.

Denote the set of N UEs and the set of M FANs by $\mathcal{N} = \{1, ..., N\}$ and $\mathcal{M} = \{1, ..., M\}$, respectively. We further denote the library of F files as $\mathcal{F} = \{1, ..., F\}$. Without loss of generality, all files are assumed to have a uniform size with L bits. Define the association vector of UE n as $\mathbf{a}_n = (a_{n,0}, ..., a_{n,M})$, where $a_{n,x} \in \{0, 1\}$, $x \in \{0\} \cup \mathcal{M}$, with $\sum_{x\in\{0\}\cup\mathcal{M}} a_{n,x} = 1$, is an association indictor between UE n and FCN, FANs. To be specific, $a_{n,m} = 1$ indicates that UE n is associated with FAN m; otherwise, $a_{n,m} = 0$. Especially, $a_{n,0}$ is the association indictor between UE n and FCN. We further define the association profile as $\mathbf{A} = (\mathbf{a}_1^T, \mathbf{a}_2^T, ..., \mathbf{a}_N^T)^T$. Since the FANs usually have constrained storage size and communication capability, and thus can cache limited files and serve limited UEs. We further introduce a matrix $\mathbf{B} = (\mathbf{b}_1^T, \mathbf{b}_2^T, ..., \mathbf{b}_N^T)^T$, with $\mathbf{b}_n = \{b_{n,1}, b_{n,2}, ..., b_{n,M}\}$, $b_{n,m} \in \{0, 1\}$, $\forall n \in \mathcal{N},\ m \in \mathcal{M}$,

[1]Our model and algorithm also apply to other services, such as computing.

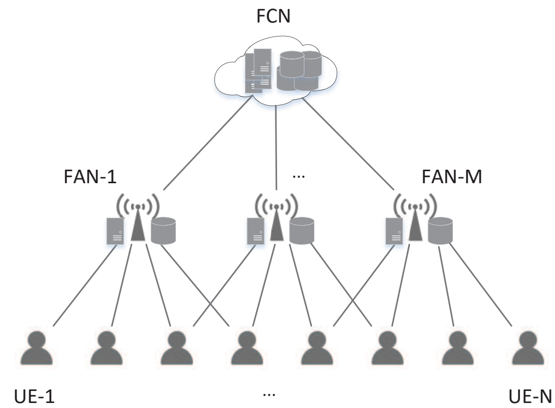

Fig. 1. An illustration of multi-tier fog computing networks.

to denote the availability of UEs' requested files at FANs, and matrix $\mathbf{C} = (\mathbf{c}_1^T, \mathbf{c}_2^T, ..., \mathbf{c}_N^T)^T$ to denote the connectivity between UEs and FANs, where $\mathbf{c}_n = \{c_{n,1}, c_{n,2}, ..., c_{n,M}\}$, $c_{n,m} \in \{0, 1\}$, $\forall n \in \mathcal{N},\ m \in \mathcal{M}$. Specifically, if $b_{n,m} = 1$, the requested file of UE n is available at FAN m; otherwise, $b_{n,m} = 0$. If $c_{n,m} = 1$, UE n can connect to FAN m; otherwise, $c_{n,m} = 0$.

Similar to many previous works [4], [5], [8], [9], a quasi-static scenario, wherein the UEs remain unchanged during a user scheduling interval, is assumed. Besides, the user scheduling problem is the focus of this work, and thus the file allocation problem is ignored in this paper.

B. Cost Model

1) Service Delay: If UE n is associated with FAN m, the downloading delay of a file can be expressed as

$$t_{n,m} = \frac{L}{R_{n,m}}, \tag{1}$$

where $R_{n,m}$ is the transmission rate from FAN m to UE n [4], [7], [8].

If UE n can not obtain the requested file from neighboring FANs or it is more cost-effective to get the file from FCN, UE n will be associated with FCN. Similarly, the downloading delay of a file can be written as

$$t_{n,0} = \frac{L}{R_{n,0}}, \tag{2}$$

where $R_{n,0}$ is the transmission rate from FCN to UE n.

2) Payment: Just like the payment scheme for online advertisement, i.e., cost-per-click [10], the FANs are assumed to charge by usage amounts or downloads. To motivate more UEs to download files from it, and thus earn more revenues, the FANs set their price as an inverse demand function [11]. Assume a linear inverse demand function, and the price for single download or the payment for downloading a file from FANs is given by

$$\alpha_m - \beta_m \sum_{n=1}^{N} a_{n,m}, \forall m \in \mathcal{M}, \tag{3}$$

where α_m and β_m are two price-related constants set by FAN m. It is worth noting that FCN pays money to FANs for UEs downloading files from them.

If the UEs download files from FCN, FCN will pay extra for electric power consumption. Without loss of generality, a constant payment or cost γ for per download is considered here.

3) Cost Function: The overall cost function for FCN is defined as a combination of service delay and payment [4], [12], [13], which is given by

$$O_n(\mathbf{a}_n, \mathbf{A}_{-n}) = \lambda_n^T a_{n,0} \frac{L}{R_{n,0}} + \lambda_n^C a_{n,0}\gamma + \lambda_n^T \sum_{m=1}^{M} a_{n,m} \frac{L}{R_{n,m}} + \lambda_n^C \sum_{m=1}^{M} a_{n,m} \left(\alpha_m - \beta_m \sum_{n=1}^{N} a_{n,m} \right), \tag{4}$$

where $\lambda_n^T, \lambda_n^C \in [0,1]$ denote the weighting parameters of service delay and payment set by FCN, respectively, and $\mathbf{A}_{-n}$ is the association vectors of all UEs except n. If a UE is more sensitive to delay, or a UE subscribes to better services, λ_n^T is higher; otherwise, λ_n^T may be lower.

C. Problem Formulation

In this paper, we focus on the user scheduling problem to minimize the overall cost of FCN, i.e.,

$$\min_{\mathbf{A}} \sum_{n=1}^{N} O_n(\mathbf{a}_n, \mathbf{A}_{-n}) \tag{5a}$$

$$s.t.\ a_{n,0}, a_{n,m} \in \{0,1\}, \forall n \in \mathcal{N}, m \in \mathcal{M}, \tag{5b}$$

$$a_{n,0} + \sum_{m \in \mathcal{M}} a_{n,m} = 1, \tag{5c}$$

$$a_{n,m} \le b_{n,m}, \forall m \in \mathcal{M}, \tag{5d}$$

$$a_{n,m} \le c_{n,m}, \forall m \in \mathcal{M}. \tag{5e}$$

Constraints (5b) and (5c) ensure that each UE is associated with only one FAN or FCN. Constraint (5d) guarantees that each UE is associated with FANs which cache its requested file. Constraint (5e) assures that each UE is associated with FANs, to which it can connect. The optimization problem (5) is an NP-hard combinatorial problem, which has a high computational complexity.

In the following section, we will reformulate the problem into a user scheduling game, which can be proven to be a potential game, and thus can be effectively solved by a distributed algorithm.

III. COUS Algorithm

A. Game Formulation

Define our user scheduling game as $G = \{\mathcal{N}, \{\mathcal{A}_n\}_{n\in\mathcal{N}}, \{O_n\}_{n\in\mathcal{N}}\}$, where $\mathcal{A}_n = \{\mathbf{a}_n | a_{n,0}, a_{n,m} \in \{0,1\}, a_{n,0} + \sum_{m\in\mathcal{M}} a_{n,m} = 1, a_{n,m} \le b_{n,m}, a_{n,m} \le c_{n,m}, \forall m \in \mathcal{M}\}$ is the association strategy space of UE n.

Definition 1. The *best-response function* $b_n(\mathbf{A}_{-n})$ of UE n to the given $\mathbf{A}_{-n}$ is a set of strategies for UE n such that

$$b_n(\mathbf{A}_{-n}) = \{\mathbf{a}_n | O_n(\mathbf{a}_n, \mathbf{A}_{-n}) \le O_n(\mathbf{a}'_n, \mathbf{A}_{-n}), \forall\, \mathbf{a}'_n \in \mathcal{A}_n\}. \tag{6}$$

Definition 2. An association profile $\bar{\mathbf{A}} = \{\bar{\mathbf{a}}_1^T, \bar{\mathbf{a}}_2^T, ..., \bar{\mathbf{a}}_N^T\}^T$ is a *pure-strategy nash equilibrium* of the user scheduling game G if and only if

$$\bar{\mathbf{a}}_n \in b_n(\bar{\mathbf{A}}_{-n}),\ \forall n \in \mathcal{N}. \tag{7}$$

At the NE point $\bar{\mathbf{A}}$, no UE can change its association strategy to further reduce its cost, while keeping other UEs' association strategies fixed.

B. Existence of NE

Theorem 1. The user scheduling game G possesses at least one pure-strategy NE and guarantees the *finite improvement property*.

Proof: We first prove that the game G is a *weighted potential game* [14] with potential function

$$\begin{aligned}\Phi(\mathbf{A}) = &\sum_{n=1}^{N} \frac{\lambda_n^T}{\lambda_n^C} a_{n,0} \frac{L}{R_{n,0}} + \sum_{n=1}^{N} a_{n,0}\gamma + \sum_{n=1}^{N} \frac{\lambda_n^T}{\lambda_n^C} \sum_{m=1}^{M} a_{n,m} \frac{L}{R_{n,m}} \\ &+ \sum_{n=1}^{N}\sum_{m=1}^{M} a_{n,m}\alpha_m - \frac{1}{2}\sum_{n=1}^{N}\sum_{k=1,k\ne n}^{N}\sum_{m=1}^{M} \beta_m a_{n,m} a_{k,m} \\ &- \sum_{n=1}^{N}\sum_{m=1}^{M} \beta_m a_{n,m}^2\end{aligned}, \tag{8}$$

such that

$$O_n(\mathbf{a}_n, \mathbf{A}_{-n}) - O_n(\mathbf{a}'_n, \mathbf{A}_{-n}) = w_n \left(\Phi(\mathbf{a}_n, \mathbf{A}_{-n}) - \Phi(\mathbf{a}'_n, \mathbf{A}_{-n}) \right), \quad \forall\ \mathbf{a}_n,\ \mathbf{a}'_n \in \mathcal{A}_n,\ \mathbf{A}_{-n} \in \prod_{m \ne n} \mathcal{A}_m, \tag{9}$$

where $(w_n)_{n\in\mathcal{N}}$ is a vector of positive numbers, i.e., weights.

$$\begin{aligned}&\frac{1}{\lambda_n^C} \left(O_n(\mathbf{a}_n, \mathbf{A}_{-n}) - O_n(\mathbf{a}'_n, \mathbf{A}_{-n}) \right) \\ =& \frac{\lambda_n^T}{\lambda_n^C} \left(a_{n,0} - a'_{n,0} \right) \frac{L}{R_{n,0}} + \left(a_{n,0} - a'_{n,0} \right) \gamma \\ &+ \frac{\lambda_n^T}{\lambda_n^C} \sum_{m=1}^{M} \left(a_{n,m} - a'_{n,m} \right) \frac{L}{R_{n,m}} \\ &+ \sum_{m=1}^{M} \left(a_{n,m} - a'_{n,m} \right) \left(\alpha_m - \beta_m \sum_{k=1,k\ne n}^{N} a_{k,m} \right) \\ &- \sum_{m=1}^{M} \beta_m \left(a_{n,m}^2 - a'^2_{n,m} \right)\end{aligned}. \tag{10}$$

$$
\begin{aligned}
&\Phi(\mathbf{a}_n, \mathbf{A}_{-n}) - \Phi(\mathbf{a}'_n, \mathbf{A}_{-n}) \\
=&\frac{\lambda_n^T}{\lambda_n^C}\left(a_{n,0} - a'_{n,0}\right)\frac{L}{R_{n,0}} + \left(a_{n,0} - a'_{n,0}\right)\gamma \\
&+ \frac{\lambda_n^T}{\lambda_n^C}\sum_{m=1}^{M}\left(a_{n,m} - a'_{n,m}\right)\frac{L}{R_{n,m}} \\
&+ \sum_{m=1}^{M}\left(a_{n,m} - a'_{n,m}\right)\alpha_m \\
&- \frac{1}{2}\sum_{k=1,k\neq n}^{N}\sum_{m=1}^{M}\beta_m\left(a_{n,m} - a'_{n,m}\right)a_{k,m} \\
&- \frac{1}{2}\sum_{k=1,k\neq n}^{N}\sum_{m=1}^{M}\beta_m a_{k,m}\left(a_{n,m} - a'_{n,m}\right) \\
&- \sum_{m=1}^{M}\beta_m\left(a_{n,m}^2 - a_{n,m}^{'2}\right) \\
=&\frac{\lambda_n^T}{\lambda_n^C}\left(a_{n,0} - a'_{n,0}\right)\frac{L}{R_{n,0}} + \left(a_{n,0} - a'_{n,0}\right)\gamma \\
&+ \frac{\lambda_n^T}{\lambda_n^C}\sum_{m=1}^{M}\left(a_{n,m} - a'_{n,m}\right)\frac{L}{R_{n,m}} \\
&+ \sum_{m=1}^{M}\left(a_{n,m} - a'_{n,m}\right)\left(\alpha_m - \beta_m\sum_{k=1,k\neq n}^{N}a_{k,m}\right) \\
&- \sum_{m=1}^{M}\beta_m\left(a_{n,m}^2 - a_{n,m}^{'2}\right) \\
=&\frac{1}{\lambda_n^C}\left(O_n(\mathbf{a}_n, \mathbf{A}_{-n}) - O_n(\mathbf{a}'_n, \mathbf{A}_{-n})\right)
\end{aligned}
\quad . \tag{11}
$$

In conclusion, the user scheduling game G is a weighted potential game with the potential function as given in (8).

For finite potential games, there exists at least one pure-strategy NE. Furthermore, every sequence of better and best responses converges to an NE, regardless of its starting point, i.e., the finite improvement property.

■

C. Algorithm Design

By employing such a property stated in Theorem 1, we can design a distributed user scheduling algorithm called COUS, as [13]. To begin with, each UE chooses to be associated with FCN (line 2). Then, at each iteration t, each UE n will transmit the pilot signal to the FCN and FANs, which UE n can connect to and has the requested files of UE n, i.e., available FANs (line 5). The FCN and FANs measure the values of $L/R_{n,0}$, γ, $L/R_{n,m}$, and $\alpha_m - \beta_m$, and transmit these necessary information back to UE n (line 6). Accordingly, each UE n can calculate the best response $b_n(\mathbf{A}_{-n}(t))$ (line 7). If $\mathbf{a}_n(t) \notin b_n(\mathbf{A}_{-n}(t))$, i.e., UE n can reduce its cost, UE n sends a request to update (RTU) message to the FCN for contending for the association strategy update opportunity (line 9). Then, the FCN randomly decides a UE who sends the RTU message and sends the update-permission (UP) message to it. Once receiving the UP message, UE n updates its association strategy as $\mathbf{a}_n(t+1) \in b_n(\mathbf{A}_{-n}(t))$ at next iteration (line 11). For other UEs who do not receive the UP message, they maintain the current association strategy at next iteration (line 13). If no UEs want to update the current association strategy, i.e., no RTU messages are sent to FCN, the FCN will send the END message to all UEs and the algorithm terminates (line 18). The whole algorithm runs as Algorithm 1.

Algorithm 1 COUS Algorithm

1: **initialization**:
2: each UE n chooses to be associated with FCN, i.e., $\mathbf{a}_n(0) = [1, 0, ..., 0]$.
3: **end initialization**
4: **repeat** for each UE n and each iteration in parallel:
5: send the pilot signal to FCN and available FANs.
6: receive the necessary information from FCN and available FANs.
7: compute the best response $b_n(\mathbf{A}_{-n}(t))$.
8: **if** $\mathbf{a}_n(t) \notin b_n(\mathbf{A}_{-n}(t))$ **then**
9:   send RTU message to FCN for contending for the association strategy update opportunity.
10:   **if** receive the UP message from FCN **then**
11:     update the association strategy $\mathbf{a}_n(t+1) \in b_n(\mathbf{A}_{-n}(t))$ for next iteration.
12:   **else**
13:     maintain the current association strategy $\mathbf{a}_n(t+1) = \mathbf{a}_n(t)$ for next iteration.
14:   **end if**
15: **else**
16:   maintain the current association strategy $\mathbf{a}_n(t+1) = \mathbf{a}_n(t)$ for next iteration.
17: **end if**
18: **until** END message is received from FCN.

D. Price of Anarchy

In game theory, price of anarchy (PoA) is most often used to evaluate the efficiency of an NE solution. It answers the question that how far is the overall performance of an NE from the socially optimal solution. To be specific, let Γ be the set of NEs of the user scheduling game G and $\mathbf{A}^* = \{\mathbf{a}_1^{*T}, \mathbf{a}_2^{*T}, \cdots, \mathbf{a}_N^{*T}\}^T$ be the centralized optimal solution that minimizes the system cost. Then, the PoA is defined as

$$
PoA = \frac{\max\limits_{\mathbf{A}\in\Gamma}\sum\limits_{n\in\mathcal{N}} O_n(\mathbf{A})}{\sum\limits_{n\in\mathcal{N}} O_n(\mathbf{A}^*)}. \tag{12}
$$

For the user scheduling game G, we have the following theorem.

Theorem 2. For the user scheduling game G, the PoA of the system cost satisfies that

$$1 \leq PoA \leq \frac{\sum_{n=1}^{N} \min\{O_{n,0}, O_{n,m}^{\min}\}}{\sum_{n=1}^{N} \min\{O_{n,0}, O_{n}^{\min}\}}, \quad (13)$$

where $O_{n,0} \triangleq \lambda_n^T L/R_{n,0} + \lambda_n^T \gamma$, $\min_{m\in\mathcal{M}} \left(\lambda_n^T L/R_{n,m} + \lambda_n^C(\alpha_m - \beta_m)\right) \triangleq O_{n,m}^{\min}$, and $\min_{\mathbf{A}'\in\prod_{n\in\mathcal{N}}\mathcal{A}_n} O_n(\mathbf{A}') \triangleq O_n^{\min}$.

Proof: Due to space constraints, the proof is omitted here.

IV. Performance Evaluation

A. Simulation Setup

There are totally $F = 10$ files, each of which is 10 M bits, and $M = 10$ FANs, each of which can cache 4 random files. Assume that the UEs communicate with FCN and FANs via Long Term Evolution (LTE). As measured in [15], the average data rate of LTE is 5.85 Mbps, and thus the data rate between FANs and UEs, i.e., $R_{n,m}$, is randomly distributed in $[5.35, 6.35]$ Mbps. Furthermore, generally, we have $R_{n,0} < R_{n,m}$ [7], and thus the data rate between FCN and UEs, i.e., $R_{n,0}$, is randomly selected from $[4.35, 5.35]$ Mbps. Set $\gamma = 4$, while α_m is uniformly distributed over $[5.5, 6.5]$. To guarantee that the revenues of FANs can increase with the increasing of downloads, β_m is randomly chosen from $[0.05, 0.1]$ in our simulation. Besides, λ_n^T and λ_n^C are uniformly and randomly selected from $[0.5, 1]$ and $[0.1, 0.2]$, respectively. All numerical results are averaged over 500 simulation trials. In each simulation trial, the requested file of each UE is randomly determined.

B. Overall Cost

Fig. 2 compares our proposed COUS algorithm with the following baseline solutions in terms of the overall cost:

- optimal scheduling (Optimal): the near-optimal solution to the overall cost minimization is obtained, utilizing the Cross Entropy method [16].
- random scheduling (Random): each UE is randomly associated with FCN or one FAN.
- FCN scheduling (FCN): each UE is associated with FCN.

As demonstrated in Fig. 2, the overall cost increases as the number of UEs increases, and the COUS algorithm can always offer the near-optimal performance. The COUS algorithm shows a better performance than the random scheduling scheme and the FCN scheduling scheme, especially when the number of UEs is large.

C. Average Cost

Fig. 3 shows the average cost with different number of UEs, under different payment schemes. We compare the solution achieved by our payment scheme and algorithm, with the optimal solution (Fixed) under the fixed payment scheme, where

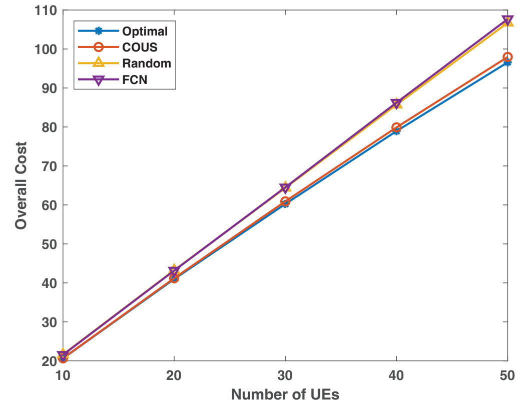

Fig. 2. Overall cost with different number of UEs.

the FANs set a fixed price for single download, regardless of downloads.

As shown in Fig. 3, the average cost of our dynamic scheme is smaller than that of fixed scheme. And, the average cost offered by the dynamic scheme, decreases with the increase of UEs. While, the average cost achieved by fixed scheme, almost maintains the same, in despite of the number of UEs.

D. Service Revenue

Fig. 4 demonstrates the revenue of all FANs with different number of UEs, under the COUS algorithm and the Fixed scheme, respectively. As demonstrated in Fig. 4, the revenue of all FANs offered by dynamic scheme is larger than that offered by fixed scheme. Moreover, both the revenues of all FANs offered by the dynamic scheme and fixed scheme increase as the number of UEs increases, while the revenue of all FANs offered by our proposed dynamic scheme shows a faster growth trend.

Fig. 3 and Fig. 4 show a win-win outcome of our proposed cost model, or dynamic payment scheme. To be specific, our proposed dynamic payment scheme can not only offer lower average cost for FCN, but also provide higher revenues for FANs, and this advantage becomes increasingly prominent as the number of UEs increases.

E. Workload Distribution

Fig. 5 illustrates the number of UEs served by different FANs under different payment schemes when $N = 50$. As illustrated in Fig. 5, our proposed dynamic payment scheme will incur unfair workload distribution among different FANs, compared with the fixed payment scheme.

V. Conclusions

In this paper, we investigated a unified multi-tier cost model and the resulted cost-minimization user scheduling problem in multi-tier fog computing networks consisting of one FCN, multiple FANs and UEs. A unified multi-tier cost model, including the service delay and a linear inverse demand dynamic payment scheme, was proposed, and a cost-minimization

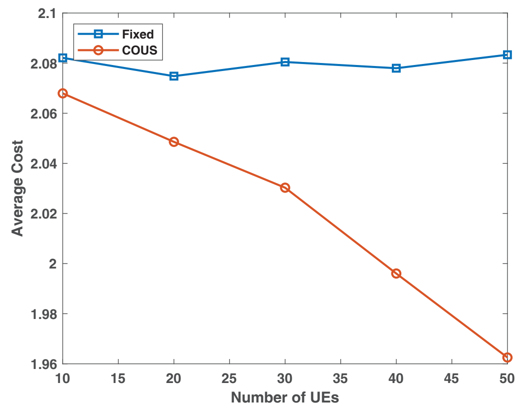

Fig. 3. Average cost with different number of UEs.

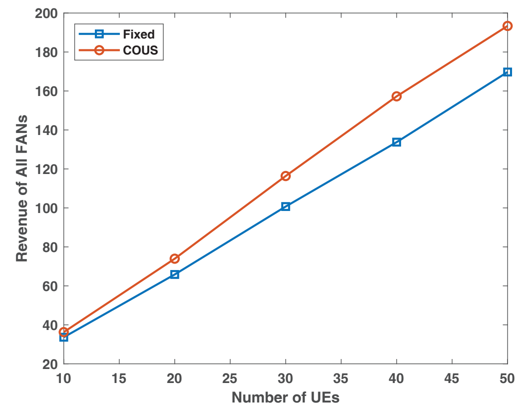

Fig. 4. Revenue of all FANs with different number of UEs.

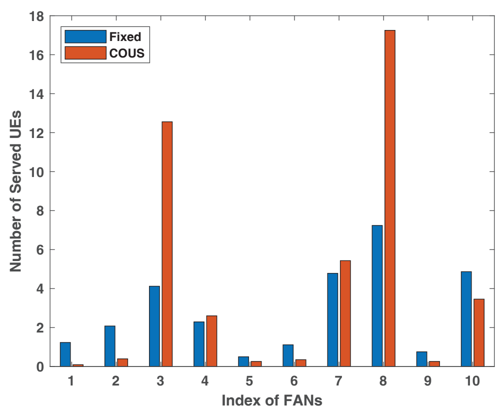

Fig. 5. Number of UEs served by different FANs ($N = 50$).

user scheduling problem was formulated. Further, the user scheduling problem was reformulated as a potential game and thus proven to possess an NE solution. Also, a distributed algorithm called COUS was developed to achieve an NE of the game. Analytical and simulation results showed that the COUS algorithm could offer near-optimal performance in terms of overall cost. Besides, the proposed dynamic payment scheme could achieve a win-win outcome for both FANs and FCN, but an unfair workload distribution among FANs, compared with the fixed payment scheme.

REFERENCES

[1] M. Chiang and T. Zhang, “Fog and IoT: An overview of research opportunities,” *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 854–864, 2016.

[2] Y. Yang, “Multi-tier computing networks for intelligent IoT,” *Nature Electronics*, vol. 2, no. 1, pp. 4–5, 2019.

[3] N. Chen, Y. Yang, T. Zhang, M.-T. Zhou, X. Luo, and J. K. Zao, “Fog as a service technology,” *IEEE Communications Magazine*, vol. 56, no. 11, pp. 95–101, 2018.

[4] H. Shah-Mansouri and V. W. Wong, “Hierarchical fog-cloud computing for IoT systems: A computation offloading game,” *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 3246–3257, 2018.

[5] Y. Liu, F. R. Yu, X. Li, H. Ji, and V. C. Leung, “Hybrid computation offloading in fog and cloud networks with non-orthogonal multiple access,” in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2018, pp. 154–159.

[6] S. Zhao, Y. Yang, Z. Shao, X. Yang, H. Qian, and C.-X. Wang, “FEMOS: Fog-enabled multitier operations scheduling in dynamic wireless networks,” *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1169–1183, 2018.

[7] T. Liu, J. Li, B. Kim, C.-W. Lin, S. Shiraishi, J. Xie, and Z. Han, “Distributed file allocation using matching game in mobile fog-caching service network,” in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2018, pp. 499–504.

[8] Z. Liu, X. Yang, Y. Yang, K. Wang, and G. Mao, “DATS: Dispersive stable task scheduling in heterogeneous fog networks,” *IEEE Internet of Things Journal*, in print, Dec. 2018.

[9] Y. Yang, K. Wang, G. Zhang, X. Chen, X. Luo, and M.-T. Zhou, “MEETS: Maximal energy efficient task scheduling in homogeneous fog networks,” *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 4076–4087, 2018.

[10] Y. Moon and C. Kwon, “Online advertisement service pricing and an option contract,” *Electronic Commerce Research and Applications*, vol. 10, no. 1, pp. 38–48, 2011.

[11] Q. D. Lã, Y. H. Chew, and B.-H. Soong, *Potential game theory: Applications in radio resource allocation*. Springer, 2016.

[12] X. Yang, Z. Liu, and Y. Yang, “Minimization of weighted bandwidth and computation resources of fog servers under per-task delay constraint,” in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[13] X. Chen, L. Jiao, W. Li, and X. Fu, “Efficient multi-user computation offloading for mobile-edge cloud computing,” *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795–2808, 2016.

[14] D. Monderer and L. S. Shapley, “Potential games,” *Games and economic behavior*, vol. 14, no. 1, pp. 124–143, 1996.

[15] J. Kwak, Y. Kim, J. Lee, and S. Chong, “DREAM: Dynamic resource and task allocation for energy minimization in mobile cloud systems,” *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 12, pp. 2510–2523, 2015.

[16] R. Y. Rubinstein and D. P. Kroese, *The cross entropy method: A unified approach to combinatorial optimization, monte-carlo simulation (Information Science and Statistics)*. Springer-Verlag New York, Inc., 2004.