Computing Resource Allocation in Three-Tier IoT Fog Networks: A Joint Optimization Approach Combining Stackelberg Game and Matching

Huaqing Zhang, *Student Member, IEEE*, Yong Xiao, *Senior Member, IEEE*, Shengrong Bu, Dusit Niyato, *Fellow, IEEE*, F. Richard Yu, *Senior Member, IEEE*, and Zhu Han, *Fellow, IEEE*

***Abstract*—Fog computing is a promising architecture to provide economical and low latency data services for future Internet of Things (IoT)-based network systems. Fog computing relies on a set of low-power fog nodes (FNs) that are located close to the end users to offload the services originally targeting at cloud data centers. In this paper, we consider a specific fog computing network consisting of a set of data service operators (DSOs) each of which controls a set of FNs to provide the required data service to a set of data service subscribers (DSSs). How to allocate the limited computing resources of FNs to all the DSSs to achieve an optimal and stable performance is an important problem. Therefore, we propose a joint optimization framework for all FNs, DSOs, and DSSs to achieve the optimal resource allocation schemes in a distributed fashion. In the framework, we first formulate a Stackelberg game to analyze the pricing problem for the DSOs as well as the resource allocation problem for the DSSs. Under the scenarios that the DSOs can know the expected amount of resource purchased by the DSSs, a many-to-many matching game is applied to investigate the pairing problem between DSOs and FNs. Finally, within the same DSO, we apply another layer of many-to-many matching between each of the paired FNs and serving DSSs to solve the FN-DSS pairing problem. Simulation results show that our proposed framework can significantly improve the performance of the IoT-based network systems.**

***Index Terms*—Fog computing, Internet of Things (IoT), matching theory, Stackelberg game.**

I. Introduction

With the rapid development of Internet of Things (IoT), the number of connected devices has increased at a unprecedented speed [1]. It is known that analyzing the big data generated by all kinds of IoT devices requires a large amount of computing resources. In order to meet the demand of the data computing services, a large number of large-scale data centers has been deployed. In addition, cloud computing has been proposed recently to provide flexible and efficient services to the data service subscribers (DSSs). In cloud computing, the data service operator (DSO) is able to organize a shared pool of configurable computing resources (such as servers, storage, networks, applications, and services), which can be easily accessed by DSSs on demands.

Generally speaking, large-scale data centers or clouds are typically built in remote areas far from the DSSs. This results in high transmission cost, transmission congestions, and service latency, which can be intolerable for the applications that require real-time interaction or mobility. Accordingly, it is beneficial and necessary to pull the cloud closer to the users. In wireless radio access network, the concept of mobile edge computing is developed by European Telecommunications Standards Institute, aiming to bring computation power into mobile radio access network. In mobile edge computing, the network edge is able to run in an isolated environment from the rest of the network and creates access to resources in the local neighborhood [5]. Moreover, in IoT, fog computing, put forward by Cisco, is proposed as a promising solution. In fog computing, multiple low-power computing devices, commonly referred to as the fog nodes (FNs), at the edge of the networks are deployed to offload the data computing services from the cloud. With the property of small-scale, low construction cost, and mobility support, the FNs are generally deployed much closer to the DSSs and therefore can provide low-latency, fast-response, and location-awareness service [6]. With different purpose and preferences, the DSSs at the network edge are able to receive data services from the FNs in the neighborhood or from remote data centers in remote areas.

In the fog computing network, the concept of network virtualization is also applied. As the large number of FNs and their computing resources are invisible for the DSSs, the DSSs can only contact and purchase the data services from the DSOs. Therefore, there is a virtualized network between DSOs and DSSs. When receiving the service requests from all DSSs, each DSO is able to collect the computing resources from the FNs and provide virtual data services for the DSSs. Based on the network requirements, each DSO is able to

Manuscript received December 1, 2016; revised March 3, 2017; accepted March 15, 2017. Date of publication March 29, 2017; date of current version October 9, 2017. This work was supported in part by the National Science Foundation under Grant CNS-1646607, Grant ECCS-1547201, Grant CCF-1456921, Grant CNS-1443917, and Grant ECCS-1405121, and in part by Singapore MOE Tier 1 (RG18/13). *(Corresponding author: Huaqing Zhang.)*

H. Zhang and Z. Han are with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77204 USA (e-mail: markasjunior@gmail.com).

Y. Xiao is with the Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ 85721 USA.

S. Bu is with the School of Engineering, University of Glasgow, Glasgow G12 8QQ, U.K.

D. Niyato is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798.

F. R. Yu is with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada.

Digital Object Identifier 10.1109/JIOT.2017.2688925

allocate different amount of computing resources from different FNs to different DSSs. Thus, the computing resource can be efficiently and effectively utilized by the nearby DSSs.

Fog computing networks can consist of a large number of FNs deployed by different DSOs at different locations to provide various data services and applications to the DSSs. When DSSs can choose their DSOs as well as the corresponding FNs to further enhance their quality-of-experience, how to allocate the limited computing resources of all the FNs to the DSSs is still an open problem. In this paper, we further extend our previous work [7] and focus on the resource selection and allocation problem between the FNs, DSOs, and DSSs. We propose a joint optimization framework for all FNs, DSOs, and DSSs in a distributed fashion. In the framework, we first formulate a Stackelberg game to model the interaction between DSOs and DSSs, where the DSOs set their service price first, and the DSSs purchase the optimal number of computing resource blocks (CRBs). Once the prices of DSOs and the purchased resources of DSSs have been obtained, each DSS can know how many CRBs are required and can then try to compete for the CRBs owned by the nearby FNs. Thus, we propose a many-to-many matching game to investigate the interaction between DSOs and FNs, where each DSO has a set of CRBs to offload, and each FN has many vacant CRBs to sell. After all the DSOs decide their DSO-FN pairs, FNs will compete with each other to allocate their CRBs to the DSSs of the DSOs. We also adopt another layer of many-to-many matching framework to solve the FN-DSS pairing problem within the same DSO. Simulation results show that our proposed framework can significantly improve the performance of the fog computing networks.

This paper is organized as follows. We describe the system model in Section II and formulate the problems in Section III. Based on the formulated problem, we analyze the system with the proposed framework in Section IV, where the interaction between DSOs and DSSs is considered in Section IV-A, the interaction between FNs and DSOs is analyzed in Section IV-B, and the interaction between FNs and DSSs is discussed in Section IV-C. Finally, we evaluate the performance of this paper in Section V, show related works in Section VI, and summarize this paper in Section VII.

II. System Model

Consider a fog computing network where each DSS can submit its data computing service to a set of neighboring FNs deployed by a set of DSOs as illustrated in Fig. 1. Accordingly, we consider a three-tier fog network, where the DSOs locate in the middle layer, managing the FNs in the upper layer and serving DSSs in the bottom layer. For clear understanding, we list the notations of this paper in Table I. Without loss of generality, we assume there are M DSOs, labeled as $\Psi = \{d_1, d_2, \ldots, d_M\}$ and N DSSs, denoted as $\Upsilon = \{s_1, s_2, \ldots, s_N\}$. Let λ_j be the workload arrival rate of DSS s_j, $\forall s_j \in \Upsilon$. We assume each DSS has a normalized preference list, denoted as $\mathbf{L}_j^s$ over all DSOs. Moreover, K FNs, labeled as $\Omega = \{f_1, f_2, \ldots, f_K\}$, locate in the area of consideration. We define the unit amount of computing resources that can be distributed by each FN as the CRB [7], each of which can provide computing service at the rate of μ. The physical data transmission network between FNs and DSSs satisfies the SecondNet topology [8], where the network facilities can provide the guaranteed quality-of-service (QoS) for

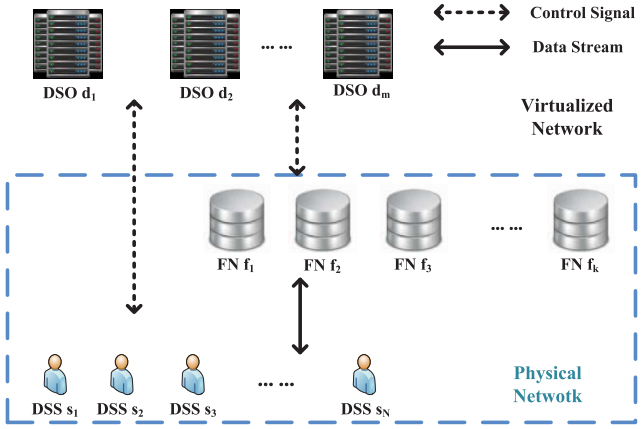

Fig. 1. System architecture.

TABLE I
List of Notations

Symbol	Definition
M	Total number of DSOs
N	Total number of DSSs
K	Total number of FNs
Ψ	The set of DSOs
Υ	The set of DSSs
Ω	The set of FNs
μ	Service rate of CRBs
λ_j	Workload arrival rate for the DSS s_j
r_i	Price of unit virtualized CRB set by the DCO o_i
$\mathbf{L}_j^s$	The preference list of DSSs over DSOs
t_j	Total cost due to the delay of DSS s_j
h_j	Cost due to network delay from the physical service provider to DSS s_j
o_j	Cost due to queuing delay at the servers
q_j	Total number of CRBs purchased by the DSS s_j
l_{kj}	Distance between the FN f_k and the DSS s_j
W_j^s	Utility function of DSS s_j
W_i^d	Utility function of DSO d_i
W_k^f	Utility function of FN f_k
τ_{ij}	The boolean variable determining whether the DSO d_i serves DSS s_j or not.
$\alpha_j, \beta_j, \gamma_j$	Weight factors in the utility function of DSS s_j
t_{th}	The maximum tolerance of service delay for DSS s_j
c_{kj}	Transmission cost for unit CRB from FN f_k to DSS s_j
e_i	Increment of the energy cost in the massive data center for DSO d_i
η_{ki}^f	Normalized preference from the FN f_k to the DSO d_i
p_k	Rent of unit CRB set by the FN f_k
r_{th}	Upper bound of total delay cost
q_{kj}^{fs}	Number of CRBs allocated from the FN f_k to the DSS s_j
q_{ik}^{fd}	Number of CRBs allocated from the FN f_k to the DSO d_i
$\mathbf{L}_i^{df}$	Preference list of DSO d_i on all FNs
$\mathbf{L}_j^{sf}$	Preference list of DSS s_j on all FNs
$\mathbf{L}_k^{fs}$	Preference list of FN f_k on all DSSs

the DSSs. Accordingly, in order to reduce the risk of potential network congestion and achieve real-time fast-response interaction, each DSO tries to offload the data services submitted by the DSSs to the large-scale data centers to the local FNs. However, as the DSSs cannot have the authorization to access the CRBs directly, the DSSs are required to receive the virtualized services from the DSOs, and with the management of DSOs, the CRBs of the FNs can finally be allocated to the DSSs. We assume that different DSOs offer data services with different requirements. Based on the preference list $\mathbf{L}_j^s$, the DSS s_j, $\forall s_j \in \Upsilon$ is required to subscribe to at most one DSO. The network architecture is illustrated in Fig. 1.

Assume the DSSs apply real-time interactive applications, where QoS is measured by the service delay. In this paper, for DSS s_j and $\forall s_j \in \Upsilon$, we measure the cost of the service delay as

$$t_j = h_j + o_j \tag{1}$$

which consists of the cost incurred by the queuing delay o_j at the servers as well as the cost incurred by the network delay h_j from the sensors to the physical service provider and from the physical service provider to DSS s_j. We suppose the workload of each DSS s_j follows Poisson arrival process. According to the M/G/1 queuing delay model, the mean response time for unit transmitted data is $[1/(\mu - (\lambda_j q_j))]$ [9], [11]. Thus, the cost of queuing delay when serving DSS s_j is

$$o_j = \frac{\lambda_j}{\mu - \frac{\lambda_j}{q_j}} \tag{2}$$

where q_j is the number of CRBs purchased by DSS s_j. Moreover, as the network delay is related to the transmission distance, traffic condition in the network and many other unpredicted factors, in practice, we suppose the network delay can be evaluated from training data periodically sent from sensors to the physical service provider and from the physical service provider to the DSS. In this paper, we set the distance between the farthest sensor to the physical service provider plus the distance between the physical service provider (e.g., FN f_k) to DSS s_j to be l_{kj}. For simplicity, we assume the network resource from the physical service provider to DSS s_j is always sufficient, and the cost incurred by the network delay h_j generally follows a linear function of the distance from the sensor to the physical service provider plus the distance from the physical service provider to the DSS s_j, i.e., $h_j = \theta l_{kj}$, where θ is a scalar.

As the DSSs in the network pay DSOs for the service, following the structure of [7], the utility of DSS s_j, $\forall j \in \Upsilon$, can be denoted as the revenue received from the workload data minus both the cost of service delay and payment to the DSOs, which can be expressed as follows:

$$W_j^s = \sum_{i=1}^{M} \tau_{ij}\left(\alpha_j \lambda_j - \beta_j q_j r_i - \gamma_j t_j\right) \tag{3}$$

where α_j denotes the revenue that DSS s_j can obtain for unit received data rate. $\alpha_j \lambda_j$ indicates the total revenue for receiving the data with workload rate λ_j. r_i is the price set by DSS d_i for each unit of the virtualized CRB. Thus, when the DSS s_j purchases q_j CRBs from the DSO d_i, the total payment from the DSS s_j to the DSO d_i can be expressed as $q_j r_i$. β_j is the weight factor indicating the importance of payment in the utility function of DSS s_j. γ_j is weight factor indicating the importance of data service delay in the utility function of DSS s_j. We set τ_{ij} to be the Boolean variable determining whether DSO d_i serves DSS s_j or not. If $\tau_{ij} = 1$, DSS s_j is served by the DSO d_i. Otherwise, DSS s_j prefers to be served by the other DSO. The value of τ_{ij} follows the preference list $\mathbf{L}_j^s$ of DSS s_j, and each DSS can at most choose one DSS, i.e., $\sum_{i=1}^{M} \tau_{ij} = 1$, $\forall j \in \Upsilon$. For each DSS, we assume there is an upper bound t_{th} for the service delay. When the service delay is larger than the threshold, the DSS will regard it as an unsuccessful connection. Corresponding, we set q_j^{th} as the lower bound of CRBs required for DSS s_j to guarantee the service delay within the acceptable thresholds.

Based on the number of virtualized CRB purchased by serving DSSs, the utility of each DSO is the revenue received from the DSSs minus the payment to the facilities that are able to provide the physical CRBs. Here, each DSO prefers to offload its services to the FNs nearby. However, if there are not sufficient available CRBs from the FNs which can meet the requirements of all DSSs. Some DSSs will be served by the remote data centers, which are located far away from the DSSs. We suppose the increment of the energy cost in the remote data center is e_i. Therefore, for the DSO d_i, if q_{ik}^{fd} CRBs are offloaded to the FN f_k, and q_i^o CRBs are still served by the remote data centers, the utility function of DSO d_i, $\forall i \in \Psi$, can be denoted as

$$W_i^d = \sum_{j=1}^{N} \tau_{ij}\left(r_i q_j\right) - \sum_{k=1}^{K} p_k q_{ik}^{fd} - e_i q_i^o \tag{4}$$

where $\sum_{j=1}^{N} \tau_{ij}(r_i q_j)$ refers to the total revenue that DSO d_i receives from DSSs for its data services. As p_k is the price set by the FN f_k, which is determined by the cost and current traffic of FN f_k, $\sum_{k=1}^{K} p_k q_{ik}^{fd}$ denotes the total payment from the DSO d_i to all FNs. $e_i q_i^o$ is the total cost for DSO d_i to serve DSSs by itself.

For FN f_k in the network, the utility is the payment received from the DSOs minus the transmission cost. We set c_{kj} as the transmission cost for each unit CRB, which has a linear relationship with the distance l_{kj}. Moreover, we set η_{ki}^f as the normalized preference to the DSO d_i. Accordingly, considering the preference to different DSOs, the discounted utility function for each DSO is

$$W_k^f = \sum_{k=1}^{K} \eta_{ki}^f \left(p_k - c_{kj}\right) q_{kj}^{fs} \tag{5}$$

where $\sum_{i=1}^{M} \eta_{ki}^f p_k q_{kj}^{fs}$ is the expected rewards received from DSOs, and $\sum_{i=1}^{M} \eta_{ki}^f c_{kj} q_{kj}^{fs}$ is the expected costs for serving DSSs in fog computing.

III. Problem Formulation

According to the modeled architecture of fog network, with tradings between FNs and DSOs and between DSOs and

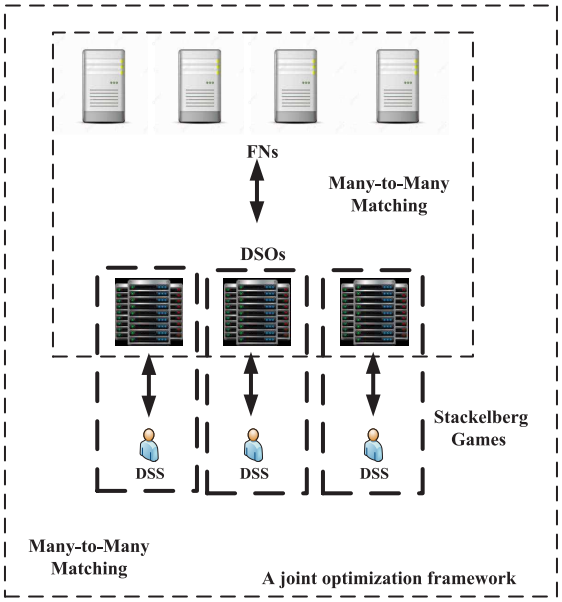

Fig. 2. Joint optimization framework.

DSSs, it is impossible to reach the maximum utilities for all FNs, MDCOs, and DSSs simultaneously. Accordingly, we consider a sequential decision making process. During the process, the DSOs first predict the total number of CRBs purchased from their servings DSSs and set their service prices to their DSSs based on the prediction. Observing the behaviors of the DSOs, each DSS determines the optimal number of CRBs to purchase to achieve the maximum utility. Furthermore, knowing the total amount of required CRBs, the DSOs try to allocate CRBs to the FNs in the neighborhood. Finally, all FNs selected by the same DSO competes for the DSSs.

We can write the optimization problem can be formulated as follows:

$$\max_{q_j} \quad W_j^s(q_j|\mathbf{r}) \qquad \forall j \in \Upsilon$$
$$\text{s.t.} \quad \begin{cases} \sum_{i=1}^{M} \tau_{ij} = 1 \\ t_j \le t_{\text{th}} \\ \tau_{ij}\alpha_j\lambda_j \le \tau_{ij}\left(\beta_j q_j r_i + \gamma_j t_j\right) \\ q_j \ge 0 \\ \tau_{ij} \in \{0,1\} \quad \forall i \in \Psi \end{cases} \tag{6}$$

where $\mathbf{r}$ is the pricing profile of all DSOs observed by DSS s_j.

Predicting the behaviors of the DSSs, each DSO is required to set the service price to each serving DSS and compete with other DSOs to choose FNs in the neighborhood. Thus, the formulated problem of the DSO is

$$\max_{r_i} \quad W_i^d(r_i|\mathbf{q}^*, \mathbf{p}, \mathbf{r}_{-i}^*) \qquad \forall i \in \Psi$$
$$\text{s.t.} \quad \begin{cases} \sum_{j=1}^{N} \tau_{ij}(r_i q_j) \ge \sum_{k=1}^{K} p_k q_{ik}^{fd} + e_i q_i^o \\ r_i \ge 0 \end{cases} \tag{7}$$

where $\mathbf{q}^*$ denotes the optimal number of CRBs purchased by all DSSs. $\mathbf{p}$ is the profile of rent for all FNs. $\mathbf{r}_{-i}^*$ is the profile of optimal service prices set by other DSOs.

Moreover, each FN in the network would like to choose its preferred DSOs and serve its DSSs with low distance. Thus, competing with other FNs, it is required to determine the number of CRBs allocated to DSOs and its serving DSSs, respectively. The optimization problem is denoted as

$$\max_{q_{ik}^{fd}, q_{kj}^{fs}} \quad W_k^f\left(q_{ik}^{fd}, q_{kj}^{fs} \middle| \mathbf{q}^*, \mathbf{q}_{i-k}^{fd*}, \mathbf{q}_{-kj}^{fs*}, \mathbf{p}_{-k}\right) \qquad \forall k \in \Omega$$
$$\text{s.t.} \quad \begin{cases} \sum_{i=1}^{M} q_{ik}^{fd} \le q_k^{f\text{th}} \\ \sum_{j=1}^{N} q_{kj}^{fs} \le q_k^{f\text{th}} \\ \sum_{k=1}^{K} q_{ik}^{fd} \le q_i^{d\text{th}} \quad \forall i \in \Psi \\ \sum_{k=1}^{K} q_{kj}^{fs} \le q_j^{s\text{th}} \quad \forall j \in \Upsilon \end{cases} \tag{8}$$

where $\mathbf{q}_{i-k}^{fd}$ is the optimal number of CRBs rent to DSO d_i for all other FNs, $\mathbf{q}_{-kj}^{fs*}$ is the optimal number of CRBs allocated to DSS s_j for all other FNs. During the service, the total CRBs distributed to all DSOs or all DSSs cannot exceed its total available CRBs $q_k^{f\text{th}}$ for FN f_k. Furthermore, the total CRBs purchased from DSO d_i or DSS s_j should not exceed its demand $q_i^{d\text{th}}$ or $q_j^{s\text{th}}$, respectively.

In summary, following the relationships among all FNs, DSOs, and DSSs, we focus on the following problems.

1) *Resource Purchasing Problem for the DSSs:* In the network, as the DSSs can only access to the DSOs in a virtualized fashion, they are required to purchase the optimal number of CRBs from the DSOs. Following the system model, as different DSSs have different tolerance of service delay, when the upper bound of service delay is high, the DSSs are able to purchase a small number of CRBs to achieve satisfying services. However, when the upper bound of service delay is low, the DSSs have to purchase a large number of CRBs to guarantee the service delay is within the tolerated region. Moreover, the service price set by the DSOs also affects the utility of DSSs. When the price is in high value, even though the large number of CRBs is able to improve the quality of data services, the DSSs have to make a large payment to the DSOs for their services. The revenue may not be satisfying. Therefore, considering both the delay tolerance and setting prices of DSOs, the optimal number of CRBs should be determined for high utilities.
2) *Pricing Problem for the DSOs:* In the data service with fog, the DSOs are required to provide virtualized CRBs to the DSSs and try to rent the CRBs from the FNs to serve DSSs in the physical network. Therefore, how to do the pricing for the DSOs is a problem. Considering the announced rent from all FNs, the DSOs need to set a price which can bring profits for themselves. However, if the price is set too high, the serving DSSs will not purchase a large number of CRBs. Therefore, predicting the reactions of CRBs and observing the rent of FNs, the DSOs are required to determine its service price so as to receive the maximum revenues.
3) *DSO-FN Pairing Problem:* As FNs may be private computing devices, which are small-scale and unable to communicate with DSSs directly, the FNs are accessible to the DSOs only. In the multi-DSO scenario, as the FNs

are accessible to all DSOs, it is a problem for all DSOs in the network to pair FNs distributedly so as to serve their DSSs with low latency. With different relations and trading history, each FN has different preference on all DSOs. Observing the rent of all FNs, each DSO also has a preferences on each FN. Based on the preferences of all DSOs' and FNs', it is required to reach a stable DSO-FN pairing results, where any DSO or FN is able switch its current pairing result for a higher utility.

4) *FN-DSS Pairing Problem:* After the pairing between DSOs and FNs, each FN has allocated its CRBs to all DSOs. However, within one DSO, it is still a problem for the FNs to allocate their CRBs to all DSSs. As the distance between each FN and each DSS is various, with a longer transmission distance, the FNs have to pay more on the transmission cost. Thus, based on the transmission distance, each FN has a preference over all DSSs. Moreover, each DSS also have preference over FNs based on the rent. Therefore, following the preference of all FNs' and DSSs', a stable FN-DSS pairing result should be achieved.

According to the formulated problems, all FNs, DSOs, and DSSs are rational and autonomous individuals, who observe the behaviors of others and make decisions to improve their own utilities. Therefore, in order to reach the optimal and stable solutions for all FNs, DSOs, and DSSs, we model a three-stage joint optimization framework, as shown in Fig. 2. In the framework, we first model the Stackelberg games to solve the pricing problem for the DSOs and resource purchasing problem for the DSSs. When the DSOs know the expected amount of resource purchased by the DSSs, a many-to-many matching is proposed between the DSOs and the FNs to deal with the DSO-FN pairing problem. Finally, within the same DSO, we apply another many-to-many matching between its paired FNs and serving the DSSs to solve the FN-DSS pairing problem.

IV. System Analysis

In this section, we analyze the optimal strategies for FNs, DSOs, and DSSs. Based on the analysis of the formulated framework, in the following sections, we first investigate the interactions between the DSOs and DSSs to determine how many CRBs are required during the service. Given the optimized behaviors of the DSOs and DSSs, we analyze the interactions between the FNs and DSOs based on different preferences. Finally, with the obtained results, we discuss the interactions between the FNs and DSSs within the same DSO for better services.

A. Interaction Between DSOs and DSSs

In the virtualized network, the DSOs provides CRBs for the DSSs. Following the formulated problems for both DSOs and DSSs, there is a Stackelberg game, where the DSOs act as leaders and DSSs act as followers. In the game, when all DSSs choose their serving DSOs with their preferences, the DSO first sets the service price. Then, based on the price all DSSs determine optimal number of CRBs to purchase. Accordingly, considering the optimization problem of DSSs, we have the following lemma.

Lemma 1: In the modeled Stackelberg game between DSO d_i and DSS s_j, when the DSO announces its service price r_i, the optimal number of CRBs q_j purchased by the DSS is

$$q_j^* = \frac{\lambda_j}{\mu\sqrt{r_i\frac{\beta_j}{\gamma_j}}} + \frac{\lambda_j}{\mu}. \tag{9}$$

Proof: According to the utility function of DSS s_j (3), the second derivative of W_j^s with respect to q_j is

$$\frac{\partial^2 W_j^s}{\partial q_j^2} = -\frac{2\lambda_j^2\mu}{\left(\mu q_j - \lambda_j\right)^3}. \tag{10}$$

As $(\partial^2 W_j^s/\partial q_j^2) < 0$, W_j^s is a quasi-concave function with respect to q_j. Furthermore, the first derivative of W_j^s with respect to q_j is

$$\frac{\partial W_j^s}{\partial q_j} = \left(\frac{\lambda_j}{\mu b_j - \lambda_j}\right)^2 - r_i\frac{\beta_j}{\gamma_j}. \tag{11}$$

We set the first derivative equal to zero and obtain the optimal number of CRBs to purchase so as to achieve the maximum utility, that is,

$$q_j^* = \frac{\lambda_j}{\mu\sqrt{r_i\frac{\beta_j}{\gamma_j}}} + \frac{\lambda_j}{\mu}. \tag{12}$$

■

Therefore, considering the reactions of the DSSs, we adjust the optimization problem for the DSO d_i, $\forall i \in \Psi$, as follows:

$$\begin{aligned} \max_{r_i} \quad & \widetilde{W}_i^d\left(r_i|\mathbf{q}^*, \mathbf{p}, \mathbf{r}_{-i}^*\right) \qquad \forall i \in \Psi \\ \text{s.t.} \quad & \begin{cases} \sum_{j=1}^N \tau_{ij}\left(\frac{\lambda_j}{\mu}\sqrt{\frac{\beta_j}{\gamma_j}}r_i + \frac{\lambda_j}{\mu}\sqrt{\frac{\beta_j}{\gamma_j}r_i}\right) \geq \sum_{k=1}^K p_k q_{ik}^{fd} + e_i q_i^o \\ r_i \geq 0 \end{cases} \end{aligned} \tag{13}$$

where

$$\widetilde{W}_i^d = \sum_{j=1}^N \tau_{ij}\left(\frac{\lambda_j}{\mu}\sqrt{\frac{\beta_j}{\gamma_j}}r_i + \frac{\lambda_j}{\mu}\sqrt{\frac{\beta_j}{\gamma_j}r_i}\right) - \sum_{k=1}^K p_k q_{ik}^{fd} - e_i q_i^o. \tag{14}$$

In the formulated problem (13), we take the first derivative of $\widetilde{W}_i^d$ with respective to r_i and discover it is a monotonous increasing function with respective to r_i. Furthermore, as the service delay cannot surpass t_{th} for the DSSs. The CRB purchased by DSSs has the following low bound:

$$q_j \geq \frac{\lambda_j t_{\text{th}}}{\mu t_{\text{th}} - \lambda_j}. \tag{15}$$

Thus, following the relation in (9), the maximum and optimal price for the DSO d_i to DSS s_j is, $\forall i \in \Psi$, $\forall j \in \Upsilon$:

$$r_i = \frac{\gamma_j}{\beta_j}\left(\frac{\mu t_{\text{th}} - \lambda_j}{\lambda_j}\right)^2. \tag{16}$$

B. Interaction Between FNs and DSOs

According to the predicted number of CRBs ordered by the DSSs, the DSOs try to offload the services from the massive data centers to the DSSs nearby. Observing the service prices set by all FNs, DSO d_i, $\forall i \in \Psi$, has a preference list $\mathbf{L}_i^{df} = [L_{i1}^{df}, \ldots, L_{iK}^{df}]$ on all FNs in the neighborhood. As the DSOs prefer to choose the FNs with a low price, we set

$$L_{ik}^{df} = -p_k \quad \forall i \in \Psi;\ \forall k \in \Omega. \tag{17}$$

Furthermore, each FN also has different preferences over DSOs. Thus, we set the preference list of FN f_k, $\forall k \in \Omega$, on all DSOs as $\mathbf{L}_k^{fd} = [L_{1k}^{fd}, \ldots, L_{Mk}^{df}]$, satisfying

$$L_{ik}^{fd} = \eta_{ki}^{f} \quad \forall i \in \Psi;\ \forall k \in \Omega. \tag{18}$$

Considering the preference lists of FNs and DSOs, i.e., L_{ik}^{df} and L_{ik}^{fd}, respectively, we design a many-to-many matching algorithm for the DSO-FN pairing problem. As shown in Algorithm 1, after the preference lists are constructed, we set a pointer for each FN in its preference list. Initially, we set the pointer at the first DSO in the list. During the each round of matching, each FN first propose to the DSO in the pointer of the preference list. Based on behaviors of the FNs, each DSO chooses its most preferred FNs in its preference list until the required CRBs are all supplied by the FNs. If the FNs supply more CRBs than the DSO requires, the DSO will reject the CRBs from less favorite FNs. If the FNs supplies less CRBs than the DSO requires, the DSO will choose massive data centers for the rest of the services. At the end of each round, if all of the CRBs from the FN have been allocated to the DSOs, the pointers of the FN will keep unchanged. Otherwise, the pointers of the rejected FNs will move to the next DSO in the preference list. In the next round, the FNs which still have available CRBs will propose to the new DSOs according to their pointers. Specifically, if the CRBs of FN are chosen by the DSO in the last round, but discarded in the current round, we suppose the pointer of the FN does not change its position, considering the pointed DSO in the current round may need more CRBs from the FN. The matching repeats in circulations until all the pointers of the FNs have moved out their preference list. According to the algorithm, we have the following lemmas.

Algorithm 1 Many-to-Many Matching Algorithm for DSO-FN Pairing Problem

```
for FN f_k do
    Construct the preference list L_k^fd on all DSOs according to (18);
    One pointer is set as the indicator pointing at the first DSO in the preference list.
end for
for DSO d_i do
    Construct the preference list L_i^df on all FNs according to (17);
end for
We set flag_k, ∀k ∈ Ω, as the indicator to show if the CRBs of FN f_k were chosen by the DSO in the last round, but discarded in the current round. Initially, flag_k = 1;
while the pointers of all FNs have not scanned all the DSOs in their preference list do
    FNs propose to DSOs with their service price;
    for FN f_k who still have available FNs to purchase do
        if flag_k = 1 then
            The pointer keep current position in the list;
        else
            The pointer moves to the next position in the list;
        end if
        The FN proposes to pointed DSO in its preference list with its available FNs;
        We set flag_k = 0;
    end for
    DSOs determines which FNs to choose;
    for DSO d_i do
        if The total available number of CRBs proposed by the FNs exceed the requirements then
            The DSO d_i chooses the most preferred number of CRBs from FNs, and rejects the rest;
            For CRBs of the FN f_k which is chosen by the DSO in the last round, but rejected in the current round, we set flag_k = 1;
        end if
    end for
end while
```

Lemma 2: For each FN in the matching algorithm, the pointer of the FN in its preference list moves in one direction. In other words, when its pointer has moved to the next DSOs in the preference list, whatever the matching results of other FNs, the FN cannot achieve a higher utility by moving the pointer back.

Proof: As shown in Algorithm 2, when the DSOs determines which FNs to choose, they choose the CRBs from the most preferred amount. If some CRBs from FN f_k is discarded by DSO d_i, the current accepted CRBs belong to the FNs which is more preferred than FN f_k. In the future rounds, when there are new FNs proposing to DSO d_i, if the DSO would like to change its current accepted FNs, the DSO can only choose the FNs that is even better than the FNs in the current accepted list. Therefore, for FN f_k which has been rejected once from DSO d_i, it is impossible for it to be accepted by the same DSO in the future rounds. ■

Lemma 3: Following Algorithm 2, the DSO-FN pairing problem will ultimately converge and achieve a stable matching result.

Proof: As proved in the Lemma 2, the pointer of the FNs can only move in one direction. Therefore, in the perspective of the FNs, when the pointer of each FN has moved to the end of the preference list, the FN has distributed its available CRBs to the DSOs in an optimized way. In other words, the FN is unable to change its pairing results unilaterally for higher utilities. Furthermore, in the perspective of the DSOs, when the pointers of all FNs have moved to the end of the preference lists, each DSO has evaluated all proposals from the available FNs. Therefore, it also cannot unilaterally change its

Algorithm 2 Many-to-Many Matching Algorithm for FN-DSS Pairing Problem

for DSS s_j **do**
  Construct the preference list $\mathbf{L}_j^{sf}$ on all DSOs according to (19);
  One pointer is set as the indicator pointing at the first FN in the preference list.
end for
for FN f_k **do**
  Construct the preference list $\mathbf{L}_k^{fs}$ on all FNs according to (20);
end for
We set $flag_j$, $\forall j \in \Upsilon$, as the indicator to show if the FNs allocate CRBs to the DSS s_j in the last round, but adjusted in the current round. Initially, $flag_j = 1$;
while the pointers of all DSSs have not scanned all the FNs in their preference list **do**
  DSSs propose to FNs for their services;
  for DSS s_j which has not been allocated required CRBs **do**
    if $flag_j = 1$ **then**
      The pointer keep current position in the list;
    else
      The pointer moves to the next position in the list;
    end if
    The DSS proposes to pointed FN in its preference list for its data services;
    We set $flag_j = 0$;
  end for
  Each FN determines which DSSs to choose;
  for FN f_k **do**
    if The total available CRBs requested by the DSSs exceed the available volume **then**
      The FN f_k allocate the CRBs to the most preferred DSSs, and rejects the rest;
      For CRBs allocated to the DSS s_j in the last round, but adjusted in the current round, we set $flag_j = 1$;
    end if
  end for
end while

pairing results, get accepted by the FNs and achieve a higher utility for itself. Furthermore, according to [10], when every agents preference list is substitutable, a pairwise stable matching always exists. Based on the above, the DSO-FN pairing will ultimately converge and achieve a stable matching result in Algorithm 2. ■

C. Interaction Between FNs and DSSs

When the CRBs from FNs have been rent to all DSOs, within each DSO, how to allocate the CRBs to all DSSs still remains a problem. According to the rent of all FNs, DSS s_j, $\forall j \in \Upsilon$, has a preference list $\mathbf{L}_j^{sf} = [L_{j1}^{sf}, \ldots, L_{jK}^{sf}]$ on all FNs in the neighborhood, satisfying

$$L_{kj}^{sf} = -p_k \quad \forall j \in \Upsilon;\ \forall k \in \Omega. \tag{19}$$

Furthermore, according to the utility function of the FN, the distance between the FN and its serving DSS affect the revenues the FN received. With a longer distance, the FN has to pay more for the data transmission in the network. Therefore, we set a preference list $\mathbf{L}_i^{fs} = [L_{j1}^{sf}, \ldots, L_{jK}^{sf}]$ for FN f_k, $\forall k \in \Omega$, over all DSSs, that is

$$L_{kj}^{fs} = -l_{kj} \quad \forall k \in \Omega;\ \forall j \in \Upsilon. \tag{20}$$

Considering the preference lists of DSSs and FNs, i.e., L_{kj}^{sf} and L_{kj}^{fs}, respectively, we design a many-to-many matching algorithm for the FN-DSS pairing problem within DSO d_i, $\forall i \in \Psi$. As shown in Algorithm 2, after the preference lists are constructed, we set a pointer for each DSS in its preference list. Initially, we set the pointer at the first FN in the list. During the each round of matching, each DSS first proposes to the FN in the pointer of the preference list. Based on behaviors of the DSSs, each FN chooses its most preferred DSSs in its preference list until the maximum CRBs available in the DSO d_i are reached. If the DSSs request more CRBs than the FN can supply, the FN will reject the less favorite DSSs. At the end of each round, if the DSSs have been allocated all of its requested CRBs from the FNs, the pointers of the DSS will keep unchanged. Otherwise, the pointers of the rejected DSS will move to the next FN in the preference list. In the next round, the DSSs which require CRBs will propose to the new FNs according to their pointers. Specifically, if some CRBs of FN are allocated to the DSS in the last round, but changed to other DSSs in the current round, we suppose the pointer of the DSS in the next round does not change its position, considering the pointed FN in the current round may be able to supply more CRBs to the DSS. The matching repeats in circulations until all the pointers of the DSSs have moved out their preference list. Following Lemmas 2 and 3 in a similar way, the FN-DSS pairing problem can ultimately achieve a stable matching result.

Based on the above, the optimal strategy of each DSO, FN, or DSS is a function with respect to the network information, e.g., channel state information and queue state information. When the network information changes dynamically, based on the function of optimal strategy, each DSO, FN, or DSS is able to adjust its strategy immediately to achieve optimal utility. However, in practice, the sensing process for network information may be performed periodically, while the time within each period is very short and we can assume the network information as constant. Accordingly, each DSO, FN, or DSS is able to iteratively update its optimal strategy to achieve high utility based on the observed network information within each period.

V. Simulation Results and Discussion

In this section, we present simulation results to evaluate our proposed framework with MATLAB. In the simulated IoT scenario, without specific explanation, there are 120 DSSs, 4 DSOs, and 20 FNs allocated randomly in a circle district with a diameter of 10 km. As we focus on the IoT scenarios, where DSSs are closely located with its sensors, without loss of generality for the methods in this paper, we suppose the

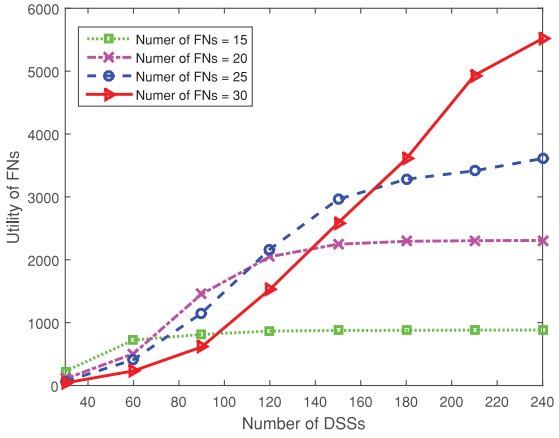

Fig. 3. Utility of all FNs versus the number of DSSs.

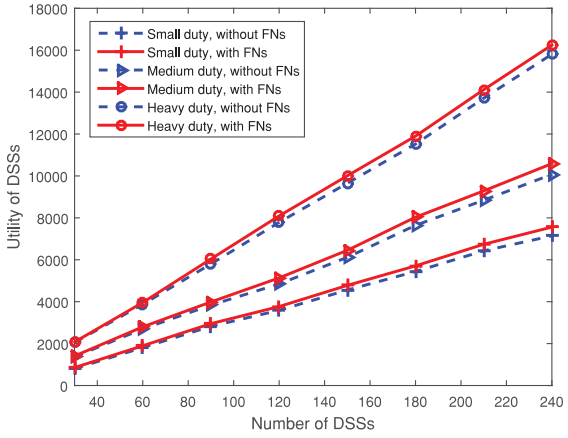

Fig. 4. Utility of all DSSs versus the number of DSSs.

sensors of each DSS are located at the same position with the DSS. We follow the settings in [11] and set the service rate of each CRB is 0.1 ms^{-1}. For each DSS, the workload arriving rate is a random number averaged 0.5 ms^{-1}. The data transmission speed is 50 km/ms. The delay tolerance of all DSSs is set to 60 ms. Furthermore, for each FN, we set its preference to each DSO as a random number satisfying the uniform distribution between [0, 1]. Based on the usage of its computing resources and its service cost, we set the announced rent as a random number satisfying the uniform distribution between (0, 10), and the amount of available CRB as a random number satisfying uniform distribution between (0, 100). The weight factors α, β, and γ are set as 50, 0.01, and 0.001, respectively.

As shown in Fig. 3, we evaluate the utility of all FNs when the number of DSSs increases. When the number of FNs is fixed, we discover with the number of DSSs increasing, the utility of all FNs generally increases, and the increasing speed first increases then gradually decreases to zero. The reason is that when the number of DSSs increases, but the number of FNs is fixed, the FNs will be able to serve more favorable DSSs with a low transmission distance. However, when all of the available CRBs of FNs are allocated to the DSSs nearby and the number of DSS keep increasing, the DSS will be allocated with CRBs from the massive data centers. Thus, the total utility of the FNs stop increasing. Nevertheless, when we increase the number of FNs, we discover that with more FNs, the utility can converge to a higher bound ultimately. Furthermore, because of the competition between FNs, when the number of DSS is small, with more FNs, the increasing speed is smaller.

In Fig. 4, we consider the utility of all DSSs with the number of DSSs increasing. In the simulation, we compare the performance of the DSSs in our proposed framework with the performance of the DSSs which is served by the massive data centers only. With the same amount of workload, we discover that when the number of DSSs increases, the utility of DSS generally increases, and the utility with FNs achieve a higher value than the one without FNs. Furthermore, when the number of DSSs is fixed and the average workload for each DSS increases, the DSSs are able to receive more revenues from the services. Thus, the utility of DSSs increases.

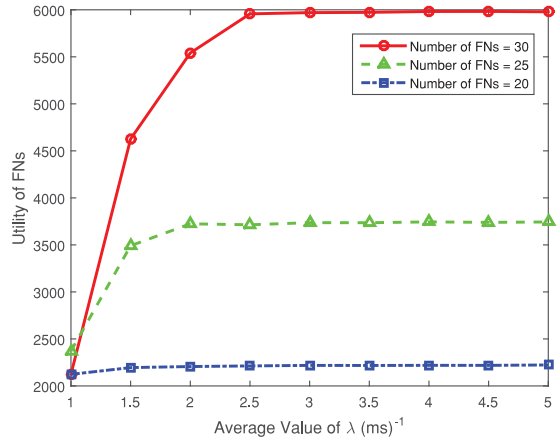

Fig. 5. Utility of all FNs versus the average value of λ.

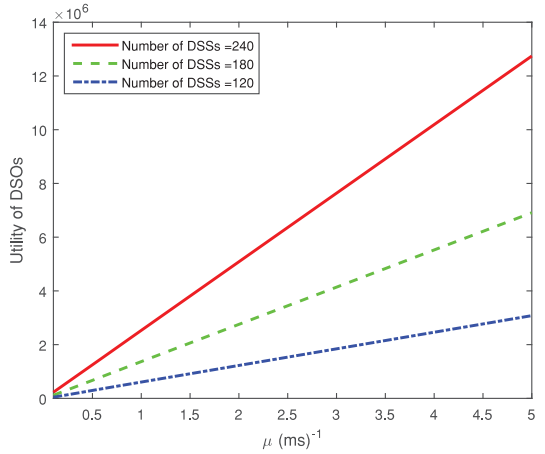

Fig. 6. Utility of all DCOs versus the value of μ.

In Fig. 5, we analyze the relation between the utility of FNs and average workload arrive rate for DSSs. As shown in the figure, when the number of FNs is fixed and the average value of workload λ increases, the utility of FNs first increases then gradually converge to a fixed value. The reason is that when the workload of all DSSs increases, the FNs are able to allocate more of its CRBs to the DSSs nearby. However, when all the available CRBs of FNs are allocated to the DSSs,

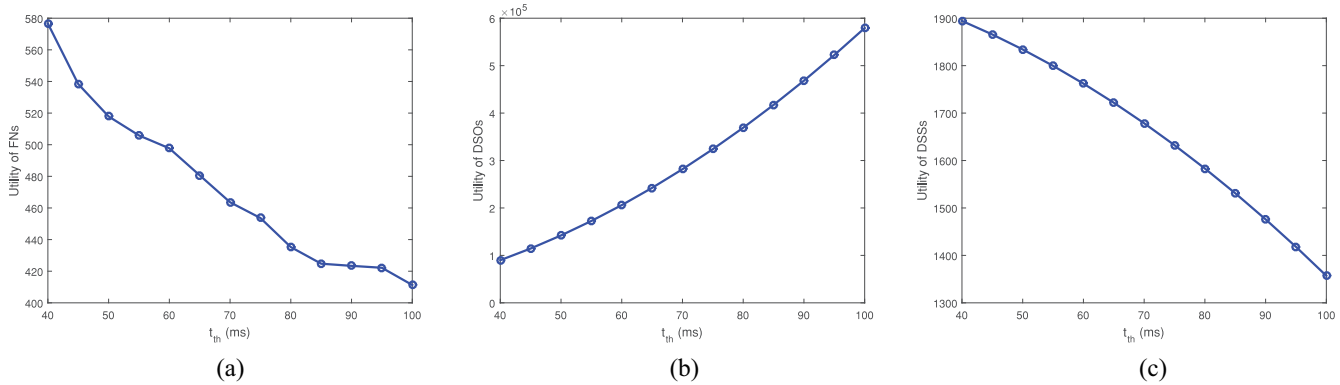

Fig. 7. Utilities versus t_{th}. (a) Utility of all FNs versus t_{th}. (b) Utility of all DSOs versus t_{th}. (c) Utility of all DSSs versus t_{th}.

the utility of the FNs stops improving and converges to one specific value. When the number of FNs increases, with the same value of λ, as the FNs are able to provide more CRBs to the DSSs, the converged value is higher.

In Fig. 6, we observe the utility of all DSOs when the value of μ increases. As shown in the figure, when μ increases, each DSS is able to be served with a less number of CRBs. Therefore, the DSO is able to set a higher price to the DSSs and receive high revenues. Moreover, when the number of DSSs increases, as the DSO is able to serve more DSSs at the same time, the total utility of DSOs also increases.

In Fig. 7, we evaluate the relationship of utilities and the tolerance service delay t_{th} of DSSs. As shown in Fig. 7(a), when the value of t_{th} increases, each DSS is able to be served with less CRBs. Thus, the FNs will supply less CRBs in the network, and the utility of the FNs generally decreases. In Fig. 7(a), as the DSO is able set a high price for its services, with the value of t_{th} increasing, the utility of DSO generally increases. Moreover, in Fig. 7(c), even though the DSS is able purchase less CRBs with high t_{th}, the price of CRBs set by the DSOs also increases. Furthermore, as the DSS suffers a lot with high delay, the utility of DSS generally decreases with the value of t_{th} increasing.

VI. Related Works

In the literature, fog computing has been advocated to be the promising future of the cloud. The concept of pulling the cloud closer to the users has been widely considered in previous work. Ahlgren *et al.* [12] put forward the concept of mist computing, aiming to distribute the cloud and its benefits deeply into the network. In [13], the deployment of edge cloud was proposed. From the DSSs' side, the edge cloud was able to surrogate the requirements and simplify the management of the network. From the servers' side, the edge cloud can exploit content and support service delivery in an efficient way. Without deploying massive data centers with high cost and latency, [14] on the other hand strengthened the importance of small distributed data center designs. The authors took email delivery as an example and showed the advantages of geo-diversity characteristics of micro data centers. In [15], a novel and distributed computing platform, called nano data centers was proposed. The authors evaluated the energy consumption of nano data centers and showed a significant improvement on energy efficiency, compared with the traditional data centers. Li *et al.* [16] proposed an intermediary framework, where there exists an intermediary between multiple cloud providers and users. The intermediary first rents the cloud service from cloud providers and then provides streaming processing service to users with low cost and delay. Li *et al.* [17] proposed a deduplication-based energy efficiency storage system for VM storage (EEVS) and implement it with existing cloud platform. In the EEVS, an online-deduplication mechanism is designed to decrease the redundant data without service interruptions, and a deduplication selection algorithm is introduced to minimize the storage energy consumption with limited computing resources for deduplication. In [18], considering the existing telecommunication and Internet service providers, the authors showed that it was required and beneficial to leverage the existing infrastructure and provide value-added services with FNs. Bonomi *et al.* [19] outlined the vision of fog computing and overviewed the important features of fog computing. In [20], the network optimization with fog computing was considered. As the data centers were aware of the locations of DSSs with fog computing, dynamic adaptation of computing resources to the DSSs' conditions was proposed. Firdhous *et al.* [21] compared the cloud computing with the fog computing and showed some significant characteristics of fog, which was required for current data services. Stojmenovic and Wen [22] elaborated the role of fog computing in six important scenarios and surveyed the security issues with fog computing.

Moreover, fog computing has been widely considered to be beneficial for the IoT. Chiang and Zhang [23] overviewed the opportunities and challenges of fog, especially the applications of fog computing in IoT. Aazam *et al.* [24] devised the method of media fog resource estimation, to provide resource estimation based on the service give-up ratio and to enhance QoS based on the previous QoE. Abedin *et al.* [25] addressed the utility-based pairing problem between the FNs and IoT devices with the Irving's matching algorithm. Giang *et al.* [26] proposed a distributed dataflow programming model for IoT devices to optimize resource allocation on computing infrastructures across the fog and the cloud. Aazam and Huh [27] considered the issues of resource prediction, customer type-based resource estimation and reservation, advance reservation, and pricing in the fog

computing for IoTs. Yannuzzi *et al.* [28] considered the requirements of mobility, scalability, reliable control, and actuation in some challenging scenarios of IoT to show the benefits and significance of fog computing. Considering the advantages of fog computing, Mei *et al.* [29] discussed and propose a procedure to be implemented in smart phones for UV measurement.

In order to solve the resource management problems in a network system with a distributed fashion, game theory has been shown as a powerful tool [30]. In the literature, most of the cases, the network system is normally modeled as a bipartite or a multitier graph. Based on this model, in [31], a Stackelberg game theoretic model was shown for dynamic bandwidth allocation between virtual networks. Wu *et al.* [32] considered a Stackelberg game between data center and buses in the smart city, where each buses collect data along its route and compete with other buses for the reward forwarding to the data center. In the game, following the proposed heuristic algorithm, the Stackelberg equilibrium is shown to be achieved, where the data center and each bus are able to reach maximum utility. Wang *et al.* [33] formulated a Stackelberg game for power allocation of data centers in the cloud. In the game, the power grid controller acts as the leader and sets prices of the provided energy based on the current amount of renewable energy and costs. Observing the prices, the cloud controller, i.e., the follower, determines the optimal amount of energy to purchase and do resource allocation for its data centers. With backward induction, the near-optimal strategies of both players in the game can be achieved. Wang *et al.* [34] modeled the interaction between the monopolistic data center operator and the customers as a Stackelberg game. In the game, the pricing strategies of the monopolistic data center operator the corresponding behavior of data service customers is detailedly analyzed in both homogeneous and heterogeneous customer scenarios. Yang *et al.* [35] adopted the Stackelberg game to solve the problem of minimizing energy consumption in the data center networks. Zhang *et al.* [36] proposed a multileader multifollower Stackelberg game to address and cooperation problems among Wi-Fi, small cell, and macrocell networks. Zhang *et al.* [37] combined the Stackelberg game and the bargaining to design a resource allocation problem in a multitier LTE unlicensed network. Furthermore, the auction mechanisms are also powerful tools to solve the problem. In [38]–[40], the resource management problem could be perfectly optimized, but it requires high communication and computation overhead. Narayanam [41] adopted the generic game theoretic framework to identify important edges in the context of k-edge connectivity between certain pairs of nodes in a general given network. In [42], the graphical game was put forward to analyze the optimized behaviors of each node in a general graph.

VII. Conclusion

In this paper, we proposed a joint optimization framework in the multi-FN, multi-DSO, and multi-DSS scenario for IoT fog computing. In the framework, we first modeled the Stackelberg games to solve the pricing problem of the DSOs and resource purchasing problem of the DSSs. Then a many-to-many matching was proposed between the DSOs and the FNs to deal with the DSO-FN pairing problem. Finally, we applied another many-to-many matching between its paired FNs and serving DSSs to solve the FN-DSS pairing problem within the same DSO. For each stage of the problem, all participants were able to achieve the equilibrium or stable results, where no one was able to change its behavior unilaterally for a higher utility. Simulation results showed that all FNs, DSOs, and DCOs were able to reach optimal utilities for themselves, and high performance of the proposed framework could be achieved compared with the data services without FNs. For the future work, first the dynamic computing resource allocation problem can be considered in the three-tier IoT fog network with dynamic Stackelberg game, where each DSO is able to predict its future demands and to rent the computing resources of FNs in advance. Second, the analysis of cooperative and competitive behaviors among FNs may provide grouping strategies for FNs to achieve higher revenues. Correspondingly, the effective strategies are required for each DSO to prevent the severe competition for some FNs with other DSOs.

References

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

[2] M. F. Bari *et al.*, "Data center network virtualization: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 909–928, 2nd Quart., 2013.

[3] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: Research problems in data center networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 1, pp. 68–73, Jan. 2009.

[4] V. Pandey, R. Saha, and T. Chao, "Network virtualization for a virtualized server data center environment," U.S. Patent 12 937 206, Apr. 2009.

[5] *Broadband Application and Service Optimization: Mobile Edge Computing (MEC) and Fog Computing*. Accessed on May 2016. [Online]. Available: http://www.prnewswire.com/news-releases/broadband-application-and-service-optimization-mobile-edge-computing-mec-and-fog-computing-300262383.html

[6] I. Goiri, K. Le, J. Guitart, J. Torres, and R. Bianchini, "Intelligent placement of datacenters for Internet services," in *Proc. IEEE ICDCS*, Minneapolis, MN, USA, Jun. 2011, pp. 131–142.

[7] H. Zhang *et al.*, "Fog computing in multi-tier data center networks: A hierarchical game approach," in *Proc. IEEE ICC*, Kuala Lumpur, Malaysia, May 2016, pp. 1–6.

[8] C. Guo *et al.*, "SecondNet: A data center network virtualization architecture with bandwidth guarantees," in *Proc. 6th Int. Conf. ACM*, Philadelphia, PA, USA, Nov./Dec. 2010, p. 15.

[9] A. Gandhi, M. Harchol-Balter, R. Das, and C. Lefurgy, "Optimal power allocation in server farms," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 37, no. 1, pp. 157–168, Jun. 2009.

[10] A. E. Roth, "The evolution of the labor market for medical interns and residents: A case study in game theory," *J. Polit. Econ.*, vol. 92, no. 6, pp. 991–1016, Dec. 1984.

[11] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. H. Andrew, "Geographical load balancing with renewables," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 39, no. 3, pp. 62–66, Dec. 2011.

[12] B. Ahlgren *et al.*, "Content, connectivity, and cloud: Ingredients for the network of the future," *IEEE Commun. Mag.*, vol. 49, no. 7, pp. 62–70, Jul. 2011.

[13] S. Islam and J.-C. Grégoire, "Network edge intelligence for the emerging next-generation Internet," *Future Internet*, vol. 2, no. 4, pp. 603–623, Dec. 2010.

[14] K. Church, A. Greenberg, and J. Hamilton, "On delivering embarrassingly distributed cloud services," in *Proc. ACM HotNets*, Calgary, AB, Canada, Oct. 2008, pp. 55–60.

[15] V. Valancius, N. Laoutaris, L. Massoulie, C. Diot, and P. Rodriguez, "Greening the Internet with nano data centers," in *Proc. ACM CoNEXT*, Rome, Italy, Dec. 2009, pp. 37–48.
[16] H. Li, M. Dong, K. Ota, and M. Guo, "Pricing and repurchasing for big data processing in multi-clouds," *IEEE Trans. Emerg. Topics Comput.*, vol. 4, no. 2, pp. 266–277, Apr./Jun. 2016.
[17] H. Li, M. Dong, X. Liao, and H. Jin, "Deduplication-based energy efficient storage system in cloud environment," *Comput. J.*, vol. 58, no. 6, pp. 1373–1383, Jun. 2015.
[18] M. B. Mobley, T. Stuart, and Y. Andrew. *Next-Generation Managed Services: A Window of Opportunity for Service Providers*. Accessed on Jul. 2009. [Online]. Available: https://pdfs.semanticscholar.org/cb25/fd8ba7ad190235d8e13dccec4132e45403e6.pdf
[19] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. 1st Ed. MCC Workshop Mobile Cloud Comput. ACM*, Helsinki, Finland, Aug. 2012, pp. 13–16.
[20] J. Zhu *et al.*, "Improving Web sites performance using edge servers in fog computing architecture," in *Proc. IEEE 7th Int. Symp. Service Orient. Syst. Eng. (SOSE)*, Redwood City, CA, USA, Mar. 2013, pp. 320–323.
[21] M. Firdhous, O. Ghazali, and S. Hassan, "Fog computing: Will be the future of cloud computing?" in *Proc. 3rd Int. Conf. Inf. Appl. (ICIA)*, Kuala Terengganu, Malaysia, Jul. 2014, pp. 8–15.
[22] I. Stojmenovic and S. Wen, "The fog computing paradigm: Scenarios and security issues," in *Proc. Federated Conf. Comput. Sci. Inf. Syst. (FedCSIS)*, Warsaw, Poland, 2014, pp. 1–8.
[23] M. Chiang and T. Zhang, "Fog and IoT: An overview of research opportunities," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854–864, Dec. 2016.
[24] M. Aazam, M. St-Hilaire, C.-H. Lung, and I. Lambadaris, "MeFoRE: QoE based resource estimation at fog to enhance QoS in IoT," in *Proc. 23rd Int. Conf. Telecommun. (ICT)*, Thessaloniki, Greece, May 2016, pp. 1–5.
[25] S. F. Abedin, M. G. R. Alam, N. H. Tran, and C. S. Hong, "A fog based system model for cooperative IoT node pairing using matching theory," in *Proc. 17th Asia-Pac. Netw. Oper. Manag. Symp. (APNOMS)*, Busan, South Korea, Aug. 2015, pp. 309–314.
[26] N. K. Giang, M. Blackstock, R. Lea, and V. C. M. Leung, "Developing IoT applications in the fog: A distributed dataflow approach," in *Proc. 5th Int. Conf. Internet Things (IOT)*, Seoul, South Korea, Oct. 2015, pp. 155–162.
[27] M. Aazam and E.-N. Huh, "Fog computing micro datacenter based dynamic resource estimation and pricing model for IoT," in *Proc. IEEE 29th Int. Conf. Adv. Inf. Netw. Appl.*, Gwangju, South Korea, Mar. 2015, pp. 687–694.
[28] M. Yannuzzi, R. Milito, R. Serral-Gracià, D. Montero, and M. Nemirovsky, "Key ingredients in an IoT recipe: Fog computing, cloud computing, and more fog computing," in *Proc. IEEE 19th Int. Workshop Comput. Aided Model. Design Commun. Links Netw. (CAMAD)*, Athens, Greece, Dec. 2014, pp. 325–329.
[29] B. Mei, W. Cheng, and X. Cheng, "Fog computing based ultraviolet radiation measurement via smartphones," in *Proc. 3rd IEEE Workshop Hot Topics Web Syst. Technol. (HotWeb)*, Washington, DC, USA, Nov. 2015, pp. 79–84.
[30] Z. Han, D. Niyato, W. Saad, T. Basar, and A. Hjorungnes, *Game Theory in Wireless and Communication Networks: Theory, Models and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2011.
[31] C. Wang, Y. Yuan, C. Wang, X. Hu, and C. Zheng, "Virtual bandwidth allocation game in data centers," in *Proc. IEEE Int. Conf. Inf. Sci. Technol.*, Hubei, China, 2012, pp. 682–685.
[32] M. Wu, D. Ye, S. Tang, and R. Yu, "Collaborative vehicle sensing in bus networks: A Stackelberg game approach," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Chengdu, China, Aug. 2016, pp. 1–6.
[33] Y. Wang, X. Lin, and, M. Pedram, "A Stackelberg game-based optimization framework of the smart grid with distributed PV power generations and data centers," *IEEE Trans. Energy Convers.*, vol. 29, no. 4, pp. 978–987, Dec. 2014.
[34] H. Wang, Y. Zhao, and H. Guan, "On pricing schemes in data center network with game theoretic approach," in *Proc. 23rd Int. Conf. Comput. Commun. Netw. (ICCCN)*, Shanghai, China, Aug. 2014, pp. 1–8.
[35] B. Yang, Z. Li, S. Chen, T. Wang, and K. Li, "Stackelberg game approach for energy-aware resource allocation in data centers," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 12, pp. 3646–3658, Dec. 2016.
[36] H. Zhang, M. Bennis, L. A. DaSilva, and Z. Han, "Multi-leader multi-follower Stackelberg game among Wi-Fi, small cell and macrocell networks," in *Proc. IEEE GLOBECOM*, Austin, TX, USA, Dec. 2014, pp. 4520–4524.
[37] H. Zhang *et al.*, "A hierarchical game approach for multi-operator spectrum sharing in LTE unlicensed," in *Proc. IEEE GLOBECOM*, San Diego, CA, USA, Dec. 2015, pp. 1–6.
[38] L. M. Ausubel, "An efficient dynamic auction for heterogeneous commodities," *Amer. Econ. Rev.*, vol. 96, no. 3, pp. 602–629, Jun. 2006.
[39] G. Demange, D. Gale, and M. Sotomayor, "Multi-item auctions," *J. Polit. Econ.*, vol. 94, no. 4, pp. 863–872, Aug. 1986.
[40] D. Mishra and R. Garg, "Descending price multi-item auctions," *J. Math. Econ.*, vol. 42, no. 2, pp. 161–179, Apr. 2006.
[41] R. Narayanam, "A game theoretic approach to identify critical components in networked systems," in *Proc. Annu. SRII Glob. Conf.*, San Jose, CA, USA, 2012, pp. 515–524.
[42] M. Kearns, "Graphical games," in *Algorithmic Game Theory*. Cambridge, U.K.: Cambridge Univ. Press, 2007, pp. 159–180.

Huaqing Zhang (S'14) received the B.S. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2013. He is currently pursuing the Ph.D. degree at the Department of Electronic and Computer Engineering, University of Houston, Houston, TX, USA.

His current research interest includes distributed strategies for resource allocation and security issues in the areas of wireless cellular communication and cloud/fog computing for IoT.

Yong Xiao (S'09–M'13–SM'15) received the B.S. degree in electrical engineering from the China University of Geosciences, Wuhan, China, in 2002, the M.Sc. degree in telecommunication from the Hong Kong University of Science and Technology, Hong Kong, in 2006, and the Ph.D. degree in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2012.

He is currently a Research Assistant Professor with the Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ, USA, where he is also the Center Manager of the NSF BWAC Center. His current research interests include machine learning and game theory and their applications in wireless networks.

Shengrong Bu received the Ph.D. degree in electrical and computer engineering from Carleton University, Ottawa, ON, Canada, in 2012.

She held a research position with Huawei Technologies Canada Inc., Ottawa, as a NSERC IRDF until 2014. She is currently a Lecturer (Assistant Professor equivalent) with the School of Engineering, University of Glasgow, Glasgow, U.K. Her current research interests include energy-efficient networks, smart grids, big data analytics, wireless networks, wireless network security, cloud computing, game theory, and stochastic optimization.

Dr. Bu was a recipient of three Best Paper Awards at IEEE International conferences. Highlights of her professional activities include duties as an Editor for the IEEE Technical Committee on *Green Communications & Computing* Newsletters, an Associate Editor for *Wireless Networks* (Springer), the TPC Co-Chair for six international workshops or conference symposiums, and duties as an N2Women Mentoring Co-Chair. She has sat on the TPC for over 20 leading international conferences and workshops and served as a Reviewer for over ten leading journals.

Dusit Niyato (M'09–SM'15–F'17) received the B.Eng. degree from the King Mongkuts Institute of Technology Ladkrabang, Bangkok, Thailand, in 1999, and the Ph.D. degree in electrical and computer engineering from the University of Manitoba, Winnipeg, MB, Canada, in 2008.

He is currently an Associate Professor with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His current research interests include energy harvesting for wireless communication, Internet of Things, and sensor networks.

F. Richard Yu (S'00–M'04–SM'08) received the Ph.D. degree in electrical engineering from the University of British Columbia, Vancouver, BC, Canada, in 2003.

From 2002 to 2006, he was with Ericsson, Lund, Sweden, and a start-up in California, USA. He joined Carleton University, Ottawa, ON, Canada, in 2007, where he is currently a Professor. His current research interests include cross-layer/cross-system design, security, green ICT, and QoS provisioning in wireless-based systems.

Dr. Yu was a recipient of the IEEE Outstanding Service Award in 2016, the IEEE Outstanding Leadership Award in 2013, the Carleton Research Achievement Award in 2012, the Ontario Early Researcher Award (formerly Premiers Research Excellence Award) in 2011, the Excellent Contribution Award at IEEE/IFIP TrustCom 2010, the Leadership Opportunity Fund Award from the Canada Foundation of Innovation in 2009, and the Best Paper Award at IEEE ICC 2014, Globecom 2012, IEEE/IFIP TrustCom 2009, and the International Conference on Networking 2005. He serves on the Editorial Boards of several journals, including as Co-Editor-in-Chief for *Ad Hoc and Sensor Wireless Networks*, Lead Series Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, and the IEEE COMMUNICATIONS SURVEYS & TUTORIALS. He has served as the Technical Program Committee Co-Chair of numerous conferences. He is a Registered Professional Engineer in the Province of Ontario, Canada. He serves as a member of the Board of Governors of the IEEE Vehicular Technology Society.

Zhu Han (S'01–M'04–SM'09–F'14) received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland at College Park, College Park, MD, USA, in 1999 and 2003, respectively.

From 2000 to 2002, he was a Research and Development Engineer with JDSU, Germantown, MD, USA. From 2003 to 2006, he was a Research Associate with the University of Maryland. From 2006 to 2008, he was an Assistant Professor with Boise State University, Boise, ID, USA. He is currently a Professor with the Department of Electrical and Computer Engineering and the Department of Computer Science, University of Houston, Houston, TX, USA. His current research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid.

Dr. Han was a recipient of the NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the EURASIP Best Paper Award for the *Journal on Advances in Signal Processing* in 2015, the IEEE Leonard G. Abraham Prize in the field of communications systems (Best Paper Award in IEEE JSAC) in 2016, and several Best Paper Awards at IEEE conferences. He is currently an IEEE Communications Society Distinguished Lecturer.