Received November 26, 2018, accepted December 12, 2018, date of publication December 27, 2018, date of current version January 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2888869

Fog Load Balancing for Massive Machine Type Communications: A Game and Transport Theoretic Approach

SARDER FAKHRUL ABEDIN [1], (Student Member, IEEE), ANUPAM KUMAR BAIRAGI [1], (Member, IEEE), MD. SHIRAJUM MUNIR [1], NGUYEN H. TRAN [2], (Senior Member, IEEE), AND CHOONG SEON HONG [1], (Senior Member, IEEE)

[1]Department of Computer Science and Engineering, Kyung Hee University, Yongin-si 17104, South Korea
[2]School of Computer Science, The University of Sydney, Sydney, NSW 2006, Australia

Corresponding author: Choong Seon Hong (cshong@khu.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) funded by the Korea Government (MSIT) under Grant NRF-2017R1A2A2A05000995.

ABSTRACT The emerging fog computing and narrow-band Internet of Things (NB-IoT) wireless technologies are indispensable for the next-generation massive machine-type communication (mMTC) applications. However, the communication capacity of NB-IoT is limited compared with the ever-growing number of NB-IoT devices. Furthermore, the optimal assignment of different computational jobs ignites the issue of load balancing in the fog network to ensure a well-balanced computational resource allocation. Therefore, in this paper, we formulate a fog load balancing problem considering the communication and computation constraints, where the objective is to minimize the load balancing cost of the fog computing network empowered with the NB-IoT. First, we model the time resource scheduling problem in NB-IoT as a bankruptcy game. Within the game framework, we enforce the Shapley value-based strategic policy for the NB-IoT devices to perform uplink scheduling for mMTC applications while calculating the transmission costs of the computational jobs. We also propose greedy iterative time scheduling (GITS), complementary to the Shapley value-based scheduling but with less computational complexity. Second, we decompose the fog load balancing problem into a Hitchcock–Koopmans transportation problem that defines the overutilized and underutilized fog computing nodes based on the computational resource utilization. Subsequently, we solve the transportation problem by applying Vogel's approximation method (VAM), which finds a feasible load balancing solution to ensure optimal job assignment in the fog computing network. The simulation results illustrate that the average job load balancing cost with our approach is significantly reduced compared with the baseline methods.

INDEX TERMS Load balancing, bankruptcy game, transport theory, Hitchcock-Koopmans transportation problem, mMTC, fog computing, NB-IoT.

I. INTRODUCTION

In recent years, the massive utilization of the Fifth-generation wireless (5G) applications [1], [2] has become prominent in different spheres of life, making the role of heterogeneous IoT devices vital. However, in the 5G paradigm [3], most IoT devices for emerging machine-type communication (MTC) applications [4] are less expensive and resource constrained in terms of energy, computation, communication, and storage capability [5]. Therefore, lightweight IoT devices demand not only low power and extended cellular connectivity technologies, but also computational resource accessibility at the edge of the network. In fact, this is the reason why fog computing [6] and Narrow-Band Internet of Things (NB-IoT) [7] have become prevalent in next-generation Internet of Things (IoT) applications and devices. Unlike cloud computing, fog nodes [8] are able to respond promptly to dynamically meet the demand of IoT-enabled massive machine type communication (mMTC) applications with a diverse range of features such as communication, computation, caching, and control [9]. On the other hand, NB-IoT is proposed by the 3rd-Generation Partnership Project (3GPP), providing massive cellular connectivity and ultra-low power consumption [10]. Therefore, the fog network empowered with NB-IoT technology provides a uniform operating environment in which fog nodes are considered to be an alternate solution to the cloud radio access network (CRAN), thus ensuring long-range cellular connectivity for massive machine type communication (mMTC) applications and devices [11].

Most *resource allocation* and *task offloading* schemes [12]–[16] for fog-based computational task offloading defer the excess amount of task load the cloud environment. However, there still remain two major challenges in the task offloading and resource allocation problem, which are:

- First, the full potential of fog networks remains vastly unexplored, which leads toward significant dissipation of computational resources. In fact, the task loads at different *non-cooperative* fog servers are often disproportionately allocated, whereas the computational cost in terms of latency becomes significantly higher than that of the *cooperative* fog networks.
- Second, there is also some research on balancing the workload among the fog nodes where communications between the computational entities are exploited. The existing models do not consider joint radio and computational resource allocation for the cooperative workload-balancing problem. In fact, the communication and computational capacities of both the NB-IoT and fog networks, respectively, are limited compared to the ever-growing demand of the mMTC applications. Therefore, to ensure an efficient resource allocation and task offloading in NB-IoT empowered fog networks, optimizing both radio and computational resources is a possible way to tackle these capacity constraints.

Considering the above circumstances, in this paper we exploit the inter-fog communication between neighboring fog computing nodes and provide an efficient solution for performing the *computational job load balancing* among themselves. In addition, we also consider *transmission time scheduling over NB-IoT* technology so that the fog nodes can acquire application specific data that are sufficient for energy efficiency and transmission reliability for energy-constrained IoT devices. As a result, the local computational cost of the fog nodes can be minimized by ensuring fairness during radio resource allocation.

The main contributions of the paper are summarized as follows:

- First, we investigate the problems of joint uplink transmission time scheduling and load balancing in fog computing networks, where the main goal is to minimize the computational and communication costs for the jobs.
- Second, due to the limited resources of NB-IoT, the bankruptcy game is a qualified model for this scenario. Therefore, we formulate an appropriate bankruptcy game for transmission time scheduling that ensures energy efficiency and reliable data acquisition from resource-constrained NB-IoT devices. The proposed approach fairly allocates limited transmission time resources provided by the fog computing nodes to a large number of NB-IoT devices.
- Third, we propose a lightweight greedy iterative time scheduling (GITS) algorithm that can perform transmission time scheduling, even when the computational complexity is much lower than that of computing the Shapley value in the bankruptcy game.
- Fourth, using uplink scheduling, we calculate the transmission cost for computational jobs and decompose the fog load balancing problem into a Hitchcock-Koopmans transportation problem. In the transportation problem, the job load balancing cost includes the computational cost as well as the transmission cost. Then, we apply Vogel's Approximation Method (VAM) to optimize the job load balancing cost by generating feasible neighboring underloaded fog nodes and overloaded fog nodes to initiate the inter-fog job load offloading. The proposed approach iteratively minimizes the job load balancing cost of the fog network and provides optimal computational job assignment. As a result, the computational job load in the fog network becomes well-balanced while also ensuring efficient computational resource allocation.
- Finally, we perform extensive numerical analysis to evaluate the performance of the proposed approach to solve the fog job load balancing problem. The simulation results depict that the VAM can significantly reduce the average inter-fog job load distribution cost by 71.27% and 83.69% compared to that of the two baseline methods.

The rest of the paper is organized as follows. In Section II, we present an extensive literature of the current research. In Section III, we present the communication model, the computational model, and problem formulation. Sections IV, V, and VI describe how to solve the proposed problem using game theory, greedy approach and transportation theory, respectively. In Section VII, we present an extensive numerical analysis to validate the performance and efficiency of our proposed approach. Finally, in Section VIII, we conclude the discussion.

II. LITERATURE REVIEW

In this section, we discuss some of the significant related works and challenges, which are grouped into four categories: (i) task offloading in cloud computing, (ii) task offloading in edge computing, (iii) load balancing in edge computing, and (iv) fog computing empowered with NB-IoT wireless technology.

A. TASK OFFLOADING IN CLOUD COMPUTING

The topic of task offloading to the cloud has been studied for over a decade where the resource and energy constrained

IoT devices offload the heavy computation jobs to the remote cloud. In [17], Altamimi *et al.* have mainly focused on communication costs for effective task offloading from smartphones to the remote cloud. Since smartphones are energy-constrained, an accurate energy estimation model for WLAN is proposed that enables smartphones to decide which tasks should be offloaded when considering the communication activities. In [18], Zhang *et al.* proposed an elastic model to ensure seamless and transparent use of cloud resources to support resource-constrained mobile devices. Reference [19] provides an overview of the ''Circus Cloud'', where the intermittent connectivity feature is focused, while explaining the spectrum of computational contexts for remote computation at the cloud system in mobile environments. However, task offloading at the cloud is proven to be inefficient largely due to limitations of the user devices in terms of energy and low range connectivity.

B. TASK OFFLOADING IN EDGE COMPUTING

To mitigate the remote offloading limitations at the cloud, the mobile cloud computing [20], fog computing [21], and mobile edge computing [22] have been widely used to perform different compute-intensive tasks at the network edge. In [23], Lin *et al.* proposed a novel linear-time rescheduling algorithm that includes a minimal-delay solution to ensure energy reduction by migrating tasks among the local cores and the cloud within the Mobile cloud computing (MCC) environment. In [24], Alam *et al.* proposed a reinforcement learning-based code offloading mechanism within the fog computing paradigm to ensure low-latency service delivery towards mobile service consumers. In [25], Wang *et al.* proposed an optimal resource allocation scheme in mobile edge computing that effectively minimizes the access points' total energy consumption subject to the users' individual computational constraints. However, if the computational task loads at the edge systems are too intensive, the cloud-based offloading mechanism is widely applied to increase the computational service utility without exhausting the possibility of task load balancing among edge computing entities. Therefore, in our proposed model, we consider collaboration among the fog computing nodes so that the computational resources of the fog network can be utilized more efficiently. As a result, the task load of the fog computing nodes is well-balanced and the computational cost is significantly reduced compared to that of the cloud-based offloading mechanism.

C. LOAD BALANCING IN EDGE COMPUTING

Several existing works have investigated task-load balancing among edge computing nodes. In [26], Jia *et al.* investigated the cloudlet placement problem as well as mobile user allocation to the cloudlet in a wireless metropolitan area network (WMAN) and proposed two heuristic algorithms. The proposed algorithms also balance the workload between the cloudlets so that the service response time for the tasks can be minimized. In [27], Oueis *et al.* considered a multi-user computation offloading scenario, where the focus was to improve the users' quality of experience (QoE) by enabling load balancing in fog computing. Therefore, low-complexity small cell clustering is proposed for efficient resource management in the fog computing environment. In [28], Verbelen *et al.* proposed a component-based cyber foraging framework to optimize application specific metrics, where both the offloading and application configurations at runtime lower the execution time and energy consumption. Unlike existing works on task load balancing and offloading, in our proposed offloading cost model, we not only consider the computation costs of the offloading and load balancing, but also the communication costs for data acquisition from different IoT devices. Jointly taking into account these two costs, we solve the overall offloading cost of the fog network.

D. FOG COMPUTING WITH NB-IOT WIRELESS TECHNOLOGY

The issue of establishing IoT empowered green communication [29] and a computing network [30] is currently considered to be one of the key challenges, especially when considering the existing issues of heterogeneity, scalability, interoperability, and security, and privacy [31]. Therefore, the integration of NB-IoT communication technology has become inevitable for the edge computing paradigm. In [32], Yu *et al.* proposed a novel uplink link adaptation scheme where data transmission reliability is guaranteed, and the throughput of the NB-IoT systems is also improved significantly. In [33], Oh and Shin proposed a control plane (CP) solution to enable idle state devices to transmit small data packets without the radio resource control connection setup. In [34], Yang *et al.* addressed the challenge of low-cost and low-power the massive IoT devices in the case of enabling robust network acquisition and extended coverage. However, none of the existing works have considered the role of NB-IoT wireless technology with fog computing. In fact, both the computation and communication cost of the task offloading can be minimized, where the IoT devices can efficiently transmit a small amount of data more reliably using the NB-IoT carrier to enable edge level task execution at the fog computing nodes.

III. SYSTEM MODEL

In Fig. 1, the F-RAN slave nodes (FSNs) are capable of performing different offloading jobs based on the received sensory observations from the NB-IoT devices. We consider all the NB-IoT devices use the licensed NB-IoT carrier inside the existing LTE carrier. The data acquisition from different NB-IoT devices should be reliable enough to perform any application specific computation job. Therefore, the fog system model can be decomposed in two parts, the communication and computational models, which are discussed in detail in the later section.

A. COMMUNICATION MODEL

In the communication model, we consider, the NB-IoT devices are resource-constrained devices and thus it is

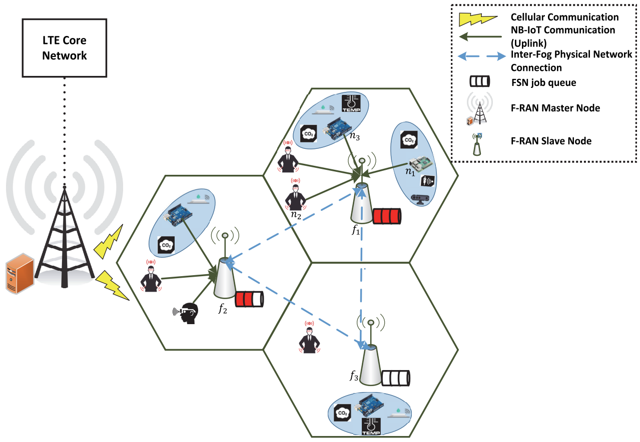

FIGURE 1. Fog system model with NB-IoT devices.

essential for such devices to conserve energy for uplink data transmission. For example, in Fig. 1, we can consider two IoT-empowered massive machine type communication (mMTC) applications within the smart city scenario: the air pollution measurement service and personalized fitness tracking application [35]. To transmit sensor observations both reliably and energy-efficiently, all the NB-IoT gateway devices (i.e., n_1, and n_3) and the wearable-device (i.e., n_2, such as a smart band, smartwatch, or smart glass) users utilize the same licensed NB-IoT spectrum. In the case of the air pollution measurement service, two NB-IoT gateway devices n_1 and n_3 are statically deployed to collect the environmental data from difference sensors (e.g., CO2 emission detector, temperature sensor, gas detector) so that the air quality of a specific geographic area can be measured and mapped at a nearby FSN f_1. Since the spectrum resource provided by FSN f_1 is limited due to the user density, FSN f_1 will require efficient spectrum resource allocation to ensure fairness. Therefore, in our scenario, we propose an efficient game theoretic approach for scheduling the transmission time for data acquisition of the energy constrained NB-IoT devices so that the network resources at the FSNs can be utilized efficiently.

Let us consider a set of Fog-RAN slave nodes (FSN) $\mathcal{F} = \{1, \cdots, F\}$ managed by the geographically closely located Fog-RAN master node (FMN) controller. At each FSN $f \in \mathcal{F}$, the computational operation is likely to be implemented as one or more multi-core CPUs [36]. Therefore, we assume each $f \in \mathcal{F}$ has a specific number of CPUs, called computing resource blocks (CRBs). There is also a set of NB-IoT devices $\mathcal{N} = \{1, \cdots, N\}$ where each $n \in \mathcal{N}$ is either a gateway device or stand-alone NB-IoT device. The set of application-specific computational jobs in the FSNs are denoted as $j \in \mathcal{J}$. Therefore, the observations from different NB-IoT devices are used for the same job $j \in \mathcal{J}$, where there is a one-to-many mapping relation between a job and a set of NB-IoT devices. We assume each $f \in \mathcal{F}$ has one dedicated LTE resource block that is used as an NB-IoT resource block.

1) UPLINK TRANSMISSION TIME AND ENERGY CONSUMPTION

The uplink transmission capacity between FSN $f \in \mathcal{F}$ and NB-IoT device $n \in \mathcal{N}$ for the corresponding jobs $j \in \mathcal{J}$ is calculated using the Shannon's capacity formula as [37]

$$\beta_{f,n} = b_{f,n} \cdot \log(1 + \gamma_{f,n}), \tag{1}$$

where $b_{f,n}$ is the fixed bandwidth provided by the FSN $f \in \mathcal{F}$ to the NB-IoT devices $n \in \mathcal{N}$. The signal-to-noise-ratio (SNR) is denoted as $\gamma_{f,n} = \frac{|h_f|^2 \kappa^j_{f,n}}{\sigma^2}$, where h_f is the channel gain, σ is the noise factor, and $\kappa^j_{f,n}$ is the uplink transmission power of the NB-IoT devices $n \in \mathcal{N}$ for the corresponding job $j \in \mathcal{J}$.

The NB-IoT transmission power over the distance between $n \in \mathcal{N}$ and $f \in \mathcal{F}$ for collecting data for job $j \in \mathcal{J}$ is defined as [38]

$$\kappa^j_{f,n}(\tau^j_{f,n}, d_{f,n}) = \beta_{f,n} \cdot \sigma \cdot d_{f,n} \cdot \left(2^{\left(\frac{\mathcal{L}^j_{f,n}}{\tau^j_{f,n} \cdot \beta_{f,n}}\right)} - 1\right), \tag{2}$$

where the number of transmitted bits for job $j \in \mathcal{J}$ by NB-IoT device $n \in \mathcal{N}$ to the corresponding FSN $f \in \mathcal{F}$ is denoted as $\mathcal{L}^j_{f,n}$ which utilizes the NB-IoT uplink transmission capacity $\beta_{f,n}$ as defined in (1). Unlike traditional cellular communication models, the transmission time $\tau^j_{f,n}$ is one of the key parameters that need to be optimized for the NB-IoT wireless communication model [39]. In fact, for a job $j \in \mathcal{J}$, energy consumption of the NB-IoT devices $n \in \mathcal{N}$ is incurred because of data acquisition over a period of uplink duration $\tau^j_{f,n}$. Since most NB-IoT devices are battery-powered, it is essential for NB-IoT devices to conserve residual energy as much as possible to last longer in the network [40]. Therefore, (2) captures the effect of transmission time on the battery life of NB-IoT devices, where much of the energy is dissipated due to the uplink transmission power.

The total energy consumption of the NB-IoT devices for transmitting $\mathcal{L}^j_{f,n}$ bits over distance $d_{f,n}$ can be calculated as [41]

$$\mathcal{E}^j_{f,n}(\tau^j_{f,n}) = \tau^j_{f,n} \cdot \kappa^j_{f,n}(\tau^j_{f,n}, d_{f,n}), \tag{3}$$

where the transmission energy $\mathcal{E}^j_{f,n}$ of the NB-IoT devices depends on the uplink transmission time $\tau^j_{f,n}$ and the distance between $n \in \mathcal{N}$ and $f \in \mathcal{F}$. Here, $\mathcal{E}^j_{f,n}$ is a monotonically increasing function of $\tau^j_{f,n}$ for the fixed number of transmitted bits $\mathcal{L}^j_{f,n}$ and bandwidth $\beta_{f,n}$.

2) PACKET ERROR RATIO (PER)

In the case of uplink transmission between the NB-IoT devices and the FSNs for the corresponding jobs, the packet errors in the outage probability can be significantly higher due to Rayleigh fading, especially in the urban environments [42]. Therefore, to ensure reliable data transmission, the NB-IoT devices re-transmit the data packets and the FSNs calculate the packet error probability while decoding $\mathcal{L}^{j}_{f,n}$ bits, given as

$$p^{j}_{f,n} = Pr\left[\gamma_{f,n} \leq \Gamma_f\right] = 1 - e^{\mathcal{L}^{j}_{f,n}\log(1-p_b)}, \tag{4}$$

where p_b is the bit error rate probability and Γ_f is the threshold of the signal-to-noise-ratio (SNR) for correct demodulation.

B. COMPUTATIONAL MODEL

In the computational model, first we consider the NB-IoT devices use the uplink channel to upload sensory observation data to the FSNs, and then the FSNs start processing the collected data for application specific jobs. The FSNs can also communicate and cooperate with each other in the case of performing computationally intensive jobs using the inter-fog physical communication channel, where an F-RAN master node (FMN) controller acts as the fog network coordinator. For example, in Fig. 1, the computational job load of FSN f_1 is comparatively higher than that of the neighboring FSNs f_2 and f_3. Therefore, f_1 exploits the inter-fog physical network connection so that a fraction of the computational jobs can be offloaded to the relatively underloaded neighboring FSNs f_2 and f_3. As a result, the computational resources at the edge computing nodes will be utilized more efficiently; in fact, the job load of the FSNs will be well-balanced, where the computational offloading costs among the neighboring FSNs are much lower than that of the job offloading cost to the remote cloud.

For a given set of FSNs $\mathcal{F}$, we consider the inter-fog offloading scenario, which is comprised of two parts. First, each FSN f needs to determine the jobs $j \in \mathcal{J}' \subset \mathcal{J}$ that need to be offloaded to a neighboring FSN $f' \in \mathcal{F}$. Second, an optimal set of neighboring FSNs $\mathcal{F}'$ should be determined by the FSN $f \in \mathcal{F}$ so that offloading cost of the offloaded job $j \in \mathcal{J}$ is minimized.

We model the FSNs as M/M/k queues [43]–[45], where each FSN $f \in \mathcal{F}$ has k_n parallel CRBs. We assume that the arrival and service processes of the jobs follow a Poisson distribution, where the arrival rate and service rate are denoted as $\lambda_f = \sum_{j\in\mathcal{J}} \lambda_{f,j}$ and μ_f, respectively. Each CRB k_n serves at most one job $j \in \mathcal{J}$ at a time. Moreover, all the arriving jobs at the corresponding FSN $f \in \mathcal{F}$ join a single queue Q_f. Therefore, the utilization factor for FSN $f \in \mathcal{F}$ is calculated as

$$u_f = \frac{\lambda_f}{k_f \times \mu_f}. \tag{5}$$

The queuing delay is calculated as

$$q_f = \frac{C(k_f, u_f)}{k_f\mu_f - \lambda_f}, \tag{6}$$

where $C(k_f, u_f)$ is known as the Erlang's C formula [46].

If FSN $f \in \mathcal{F}$ performs job $j \in \mathcal{J}$ locally (i.e., f_2 in Fig. 1), the computational delay can be calculated as [47]

$$\mathcal{C}_f(\tau^{j}_{f,n}) = \frac{\mathcal{L}^{j}_{f,n}}{c_f} + \sum_{n\in\mathbb{A}_f} \tau^{j}_{f,n} + q_f \tag{7}$$

Here, $\mathcal{L}^{j}_{f,n}$ is the job data size is, and $\tau^{j}_{f,n}$ is the data transmission time for job $j \in \mathcal{J}$ for the corresponding NB-IoT device $n \in \mathbb{A}_{f,j}$. Data processing depends on the computational cycle c_f (cycles/second) of FSN $f \in \mathcal{F}$ and the queuing delay q_f that jobs in $\mathcal{J}$ experience at FSN $f \in \mathcal{F}$.

For heavily loaded FSNs (f_1 in Fig.1), the jobs are offloaded to the neighboring FSN $f' \in \mathcal{F}'\backslash\{f\}$ (e.g., f_3 in Fig.1). The total computing delay depends on the data transmission time to the neighboring FSN, computing time, and queuing time as

$$\hat{\mathcal{C}}_f(j, \mathcal{F}') = \frac{\mathcal{L}^{j}_{f',n}}{c_{f'}} + \frac{\mathcal{L}^{j}_{f,n}}{R_{f,f'}(\beta_f)} + q_{f'} + D_{f,f'} + S_{f'}(j), \quad \forall f' \in \mathcal{F}'\backslash\{f\}. \tag{8}$$

In (8), $R_{f,f'}(\beta_f)$ is the capacity of the inter-fog uplink communication channel, where β_f is the fixed bandwidth. The queuing delay of $f' \in \mathcal{F}$ for the jobs in $\mathcal{J}$ is denoted as $q_{f'}$ in which $\mathcal{J}$ includes the local processing jobs, as well as, the incoming jobs from the other neighboring FSNs. The propagation delay between two FSNs $f \in \mathcal{F}$ and $f' \in \mathcal{F}$ is denoted as $D_{f,f'}$. $S_{f'}(j)$ is the time required for the FSN $f' \in \mathcal{F}$ to fetch the necessary application from the FMN for performing job $j \in \mathcal{J}$, which is offloaded from the parent FSN $f \in \mathcal{F}$.

The computation cost for job $j \in \mathcal{J}$ of $n \in \mathcal{N}$ is

$$\mathcal{C}_{n,j}(\tau^{j}_{f,n}, \Psi_{f,j}, \mathcal{F}') = \Psi_{f,j}\hat{\mathcal{C}}_f(j, \mathcal{F}') + (1 - \Psi_{f,j})\mathcal{C}_f(\tau^{j}_{f,n}), \tag{9}$$

where $\Psi_{f,j}$ is the binary indicator variable, which is equal to 1 when FSN $f \in \mathcal{F}$ offloads job $j \in \mathcal{J}$ to the neighboring FSN $f' \in \mathcal{F}$, otherwise the FSN performs the job locally and $\Psi_{f,j} = 0$.

C. PROBLEM FORMULATION

The data collection and inter-fog load balancing cost at each FSN $f \in \mathcal{F}$ for each job $j \in \mathcal{J}$ depend on the energy consumption of the NB-IoT devices $n \in \mathcal{N}$ for reliable data transmission and the computational delay of jobs $j \in \mathcal{J}$ of the NB-IoT devices $n \in \mathcal{N}$ at FSNs $f \in \mathcal{F}$. Therefore, the objective of the optimization problem **P1** is to minimize the computational cost for the jobs, which is accomplished by optimizing the transmission time and fog job assignment as

$$\underset{\tau^{j}_{f,n}, \Psi_{f,j}, \mathcal{F}'}{\text{Min}} \quad \mathbf{P1} = \sum_{f\in\mathcal{F}, n\in\mathcal{N}, j\in\mathcal{J}} \Phi^{j}_{f,n}\mathcal{C}_{n,j}(\tau^{j}_{f,n}, \Psi_{f,j}, \mathcal{F}')$$

$$\text{subject to } \mathcal{E}^{n}_{max} - \Phi^{j}_{f,n}\mathcal{E}^{j}_{f,n}(\tau^{j}_{f,n}) \geq 0, \quad \forall n \in \mathcal{N}, \quad \forall j \in \mathcal{J} \tag{10}$$

$$\Phi^{j}_{f,n}\tau^{j}_{f,n} \leq \tau^{th}_{f}, \quad \forall n \in \mathcal{N}, \ \forall j \in \mathcal{J} \tag{11}$$

$$\Phi^{j}_{f,n}(1-\rho^{j}_{f,n}) \geq \rho^{th}_{n}, \quad \forall n \in \mathcal{N},\ \forall j \in \mathcal{J} \tag{12}$$

$$\hat{\mathcal{C}}_{f'}(\Psi_{f,j}) \leq \mathcal{C}^{th}_{f}, \quad \forall j \in \mathcal{J},\ \forall f' \in \mathcal{F}',\ f \notin \mathcal{F}' \tag{13}$$

$$\sum_{f \in \mathcal{F}} \Psi_{f,j} \leq 1, \quad \forall j \in \mathcal{J} \tag{14}$$

$$\sum_{j \in \mathcal{J}} \Psi_{f,j} \leq Q_f, \quad \forall f \in \mathcal{F}. \tag{15}$$

In optimization problem **P1**, the computational cost is comprised of two parts, the local computational cost (i.e., computational delay) and the inter-fog offloading cost of jobs in $\mathcal{J}$. The local computational cost depends on the uplink transmission time of the NB-IoT devices $\vec{\tau} = (\tau^{j}_{f,n}), \forall f, n, j$ at the corresponding FSN $f \in \mathcal{F}$. Therefore, first the transmission time is optimized for the given initial assignment vector, where the corresponding constraints are (10), (11), and (12). Second, the inter-fog load balancing cost is minimized by reducing the computational cost of task $j \in \mathcal{J}$, where the decision variables are $\Psi_{f,j}$ and $\mathcal{F}'$ with the corresponding constraints (13), (14), and (15).

The issue of maintaining energy efficiency during uplink data transmission is crucial for relatively energy-constrained NB-IoT devices. Hence, the constraint in (10) ensures sufficient energy efficiency for each of the NB-IoT devices for application-specific data acquisition. $\mathcal{E}^{n}_{max}$ is the maximum energy of each NB-IoT device and $\mathcal{E}^{j}_{f,n}(\tau_{f,n})$ is the energy consumption of the NB-IoT device for job $j \in \mathcal{J}$ over the allocated transmission time $\tau^{j}_{f,n}$.

The constraint in (11) ensures that the allocated transmission time $\tau^{j}_{f,n}$ for job $j \in \mathcal{J}$ of FSN $f \in \mathcal{F}$ to NB-IoT device $n \in \mathcal{N}$ does not exceed the threshold transmission time τ^{th}_{f} of FSN f. As a result, FSN $f \in \mathcal{F}$ will schedule the transmission time to the NB-IoT devices $n \in \mathcal{N}$ so that the data acquisition time for job $j \in \mathcal{J}$ remains within the maximum time resource capacity.

In (12), the packet transmission success rate for job $j \in \mathcal{J}$ should be greater than or equal to the packet success threshold ρ^{th}_{n} of the NB-IoT devices using FSN $f \in \mathcal{F}$ in the allocated transmission time $\tau^{j}_{f,n}$.

When job $j \in \mathcal{J}$ is processed primarily at FSN $f \in \mathcal{F}$, the local computational cost is only calculated based on the computational capacity, the transmission time for data acquisition from different data sources, and the queuing delay. However, if any specific job $j \in \mathcal{J}$ is offloaded to the neighboring FSN $f' \in \mathcal{F}$ from FSN $f \in \mathcal{F}$, the corresponding inter-fog offloading cost $\hat{\mathcal{C}}_{n,j}$ should be considered to ensure feasible and efficient inter-fog load-balancing. Therefore, in (13) of problem **P1**, the inter-fog offloading cost of job $j \in \mathcal{J}$ of $n \in \mathcal{N}$ should not exceed a threshold set by the FSN $\mathcal{C}^{th}_{f}$.

In (14), each job $j \in \mathcal{J}$ can only be scheduled to at most one CRB of $f \in \mathcal{F}$. In (15), The number of jobs at each FSN $f \in \mathcal{F}$ should not exceed the size of the queue Q_f (i.e., the workload of the FSN).

IV. TRANSMISSION TIME SCHEDULING ALGORITHM

First, the admission control is used to construct the initial assignment vector $\Phi^{j}_{f,n}$ for job $j \in \mathcal{J}$ at each FSN $f \in \mathcal{F}$, where the energy efficient constraint (10) and reliable transmission constraint (12) in **P1** are preserved (lines 3-8, Alg. 1). Second, transmission time scheduling is performed based on the given $\Phi^{j}_{f,n}$, where we consider the requested transmission time of the NB-IoT devices for any job in $\mathcal{J}$ and thus perform time slot allocation using the bankruptcy game-based scheduling algorithm. Since $\Phi^{j}_{f,n}$ is given for job $j \in \mathcal{J}$, the transmission time can only be allocated to NB-IoT devices where $\Phi^{j}_{f,n} = 1$. Therefore, for simplicity, we change the notation of the transmission time allocation variable from $\Phi^{j}_{f,n}$ to $\Phi_{f,n}$.

Each of the NB-IoT devices/agents cooperates with each other to provide optimal transmission time scheduling, where the maximum transmission time of the FSNs is limited. In such cases, the bankruptcy game framework provides a special type of N-person cooperative game to solve the transmission time scheduling problem in the fog network. In order to solve the transmission time scheduling problem in **P1**, the Bankruptcy game (BG) is applied considering the following:

- The maximum transmission time available for the NB-IoT subcarrier deployed in the existing LTE infrastructure is limited with respect to the ever-growing number of NB-IoT devices in the case of different mMTC applications. As a result, in practice, the total transmission time claimed by NB-IoT devices is not always less than the total transmission duration provided by different FSNs. Therefore, the bankruptcy game provides a suitable option in situations (e.g., the debt crisis in a bankrupt company) where the maximum transmission time provided by the FSNs (i.e., the company) is insufficient for a large number of the NB-IoT devices (e.g. the creditors).
- The main advantage of bankruptcy game-based time resource scheduling is that the approach considers resource management as a cooperative game where the players freely participate to form coalitions and obtain profits. During the data acquisition process for mMTC applications, a coalition can be formed with the corresponding NB-IoT devices via the FSNs by using this bankruptcy game to achieve better payoffs. Moreover, using the Shapley value for such coalitions within the Bankruptcy game framework guarantees relatively fair and stable allocation [48] of transmission times among the NB-IoT devices, unlike other state-of-the-art approaches.

A. BANKRUPTCY GAME FRAMEWORK

Suppose, there is a bankrupt company that wants to allocate its resources among creditors. The demands/claims of the creditors are larger than the amount of money of the bankrupt company. Therefore, to ensure a fair allocation of company resources, an N-person game is applied to find an equilibrium point that divides the resource efficiently. Based on the bankruptcy game framework in Definition 1 below, we model the problem of transmission time scheduling, where the NB-IoT devices and FSNs are modeled as the players and the bankrupt company, respectively. The notations used in this game framework are discussed in Table 1.

TABLE 1. Notations for the Bankruptcy Game Framework for Transmission Time Scheduling.

Notation	Description
$\mathbb{A}_f$	set of players (i.e., NB-IoT devices) at FSN $f \in \mathcal{F}$
τ_f^{th}	available transmission time for each FSN $f \in \mathcal{F}$
τ_n	claimed transmission time for the NB-IoT devices $n \in \mathbb{A}_f$
$\tau_f \in \mathbb{R}^N$	transmission time allocation vector of each FSN $f \in \mathcal{F}$ for $n \in \mathbb{A}_f$
$\tau_{f,n}$	allocated transmission time for the NB-IoT device $n \in \mathbb{A}_f$

Definition 1: A *bankruptcy problem* is a pair (τ_f^{th}, τ_f) where the transmission time allocation vector $\tau_f = [\tau_{f,1}, \cdots, \tau_{f,N}]$ satisfies

(a) $\sum_{n \in \mathbb{A}_f \subset \mathcal{N}} \tau_n \geq \tau_f^{th}, \forall f \in \mathcal{F}$
(b) $0 \leq \tau_{f,n} \leq \tau_n, \forall n \in \mathbb{A}_f$
(c) $\sum_{n \in \mathbb{A}_f \subset \mathcal{N}} \tau_{f,n} = \tau_f^{th}, \forall f \in \mathcal{F}$.

In condition (a), a finite set of agents for each FSN $f \in \mathcal{F}$ is denoted as $\mathbb{A}_f$, which is a subset of the total number of NB-IoT devices in $\mathcal{N}$. For a standard bankruptcy game scenario, condition (a) implies that the total demand or claim of the NB-IoT devices should be greater than the total transmission time τ_f^{th} for each FSN $f \in \mathcal{F}$. Therefore, the minimum transmission time demand τ_n by each NB-IoT device $n \in \mathbb{A}_f \subset \mathcal{N}$ is calculated for transmitting $\mathcal{L}_{f,n}$ bits over the communication channel with additive white Gaussian noise σ, which is given as

$$\tau_n = \frac{\mathcal{L}_{f,n}}{\beta_{f,n} \log\left(1 + \frac{\kappa_{f,n}|h_n|^2}{\gamma(p_n) d_{f,n}\sigma}\right)} \tag{16}$$

In (16), the transmission power $\kappa_{f,n}$ is fixed, h_n is the channel fading coefficient, $\gamma(p_n)$ is the SNR margin that is used to meet the target packet error rate, and $\beta_{f,n}$ is the transmission capacity, which is calculated using (1). Condition (b) ensures that each NB-IoT device receives a non-negative transmission time allocation solution $\tau_{f,n}$ which should be less than or equal to the demand τ_n of each NB-IoT device. Condition (c) dictates that the available transmission time τ_f^{th} provided by the FSNs $f \in \mathcal{F}$ must be completely distributed among the NB-IoT devices $n \in \{\mathbb{A}_f \subset \mathcal{N}\}$. Based on Definition 1, the bankruptcy game framework for transmission time scheduling is proposed in Algorithm 1.

1) COALITION AND CHARACTERISTIC FUNCTION

A coalition $\mathbb{X}$ always exists in a bankruptcy game so that the agents can beneficially cooperate with each other.

Algorithm 1 Bankruptcy Game for Transmission Time Scheduling at Each FSN $f \in \mathcal{F}$

Input: τ_f^{th}
1 **Initialize:** Φ, $\mathbb{A}_f$
Result: $\vec{\tau_f}$
2 **Step 1: Admission Control**
3 **for** $n \in \mathcal{N}$ **do**
4   **if** *(10) and (12) are fulfilled* **then**
5     $\Phi_{f,n} = 1$
6     $\mathbb{A}_f = \mathbb{A}_f \cup \{\Phi_{f,n}\}$
7   **else**
8     $\Phi_{f,n} = 0$
9 **Step 2: Transmission Time Scheduling**
10 **for** $n \in \mathbb{A}_f$ **do**
11   Calculate coalition charactaristics function $\nu(\mathbb{X})$ using (17)
12   Calculate Shapley value for n, $\phi_n(\nu)$ using (20) complied with $\mathbb{I}$ and $\mathbb{C}$ in (18) and (19) repectively without violating constraints (11), and (12) in **P1**
13   $\phi_n(\nu)$ is rounded as $\tau_{f,n}$
14 $\vec{\tau_f} = \vec{\tau_f} \cup \tau_{f,n}$
15 return $\vec{\tau_f}$

Furthermore, a coalition formation game is formed, where the coalition is expressed by a characteristic function and the coalition of agents is denoted as $\mathbb{X} \subset \mathbb{A}_f$ (line 2 in Alg. 1). The empty coalition and the grand coalition are denoted as $\emptyset$ and $\mathbb{A}_f$, respectively. The characteristic function for the N-person game with the pair $(\mathbb{A}_f, \nu)$, where ν is the characteristic function, which is defined as

(a) $\nu(\emptyset) = 0$
(b) if $\mathbb{X} \cap \mathbb{X}' = \emptyset$, then $\nu(\mathbb{X}) + \nu(\mathbb{X}') \leq \nu(\mathbb{X} \cup \mathbb{X}')$.

In 1(a), if there is no coalition, the value of the characteristic function becomes 0. In 1(b), the condition describes the super-additivity property of the characteristic function and is defined as

$$\nu(\mathbb{X}) = \max(0, \tau_f^{th} - \sum_{n' \notin \mathbb{X}} \tau_{n'}), \quad \forall \mathbb{X} \subset \{\mathbb{A}_f - \emptyset\}, \tag{17}$$

for all possible coalitions $\mathbb{X}$.

2) THE CORE AND TRANSMISSION TIME SCHEDULING STABILITY

The goal of the core is to determine the stability region for the solution of an N-person cooperative game, where the solutions should have rationality constraints (i.e., desirable properties and existence conditions). The solution vector τ_f is consistent if the solutions of each $n \in \mathbb{A}_f \subset \mathcal{N}$ are both group rational and individually rational. Moreover, the individual NB-IoT devices will not stay in a coalition under the following terms:

(a) if individual NB-IoT devices receive a transmission time less than it could obtain without coalition,

(b) if the conditions (10)-(12) are violated during the transmission time allocation process.

As a result, the transmission time allocation vector τ_f must meet the additional constraints along with the other constraints (10)-(12) in **P1**, which are

$$\mathbb{I} = \{[\tau_{f,1}, \cdots, \tau_{f,N}] | \sum_{n \in \mathbb{A}_f} \tau_{f,n} = \nu(\mathbb{A}_f), \quad \text{and } \tau_{f,n} \geq \nu(n), \quad \forall n \in \mathbb{A}_f\}. \quad (18)$$

Here, $\sum_{n \in \mathbb{A}_f} \tau_{f,n} = \nu(\mathbb{A}_f)$ and $\tau_{f,n} \geq \nu(n)$ represent the group and individual rationality, respectively. Moreover, the allocation vector τ_f is considered to be stable if there is no allocation among the NB-IoT devices that violates conditions (10)-(12) in **P1** and either one of the rationality conditions in (18). Otherwise, the coalition will be unstable, which means the NB-IoT device is unsatisfied with the allocation. Therefore, to obtain a stable solution vector τ_f, we define the core as

$$\mathbb{C} = \{[\tau_{f,1}, \cdots, \tau_{f,N}] | \tau_f \in \mathbb{I}, \quad \text{and } \sum_{n \in \mathbb{X}} \tau_{f,n} \geq \nu(\mathbb{X}), \quad \forall \mathbb{X} \subset \mathbb{A}_f\}. \quad (19)$$

To obtain the core in (19), we utilize the Shapley value for the N-person cooperative game by considering the average marginal contributions of each NB-IoT device in each strategy solution (lines 10-14 in Alg. 1). The value function is defined as $\phi(\nu)$ to compute the Shapley value, which uses the value of NB-IoT device $n \in \mathcal{N}$ as

$$\phi_n(\nu) = \sum_{\mathbb{X} \subset \mathbb{A}_f, n \in \mathbb{A}_f} \frac{(|\mathbb{X}| - 1)!(N - |\mathbb{X}|)!}{N!} (\nu(\mathbb{X}) - \nu(\mathbb{X} - \{n\})), \quad (20)$$

where $|\mathbb{X}|$ is the number of elements in the set $\mathbb{X}$. In (20), the marginal contribution of NB-IoT device $n \in \mathbb{A}_f$ is captured effectively by averaging over all the sequences by which the grand coalition could be built up.

V. GREEDY ITERATIVE TIME SCHEDULING

Although the Shapley value for solving the transmission time scheduling problem is computationally less complex than the other methods such as τ-value and nucleolus [49], the computational complexity of the Shapley value is relatively still high enough to ensure fast convergence for the problem. Therefore, in this section, we provide an alternative heuristics-based solution to solve the transmission time scheduling problem which can converge much faster than when computing the Shapley value.

The objective of Greedy Iterative Time Scheduling (GITS) is to ensure reliable connectivity to a massive number of NB-IoT devices so that small data can be acquired from multiple sources for executing the jobs at the FSNs.

In the admission control phase (lines 2-8 in Alg. 2), each NB-IoT device $n \in \mathcal{N}$ checks whether or not the initial conditions are met for assignment at the FSNs $f \in \mathcal{F}$ (lines 3-4 in Alg. 2). If conditions (10) and (12) are satisfied, the initial assignment variable is set to $\phi_{f,n} = 1$, otherwise $\phi_{f,n} = 0$ (lines 5-8 in Alg. 2). After that, each FSN $f \in \mathcal{F}$ sorts the output of the initial assignment $\mathbb{A}_f$ in descending order based on the transmission time requirement τ_n of each $n \in \mathbb{A}_f \subset \mathcal{N}$ (line 10 in Alg. 2). As a result, each FSN $f \in \mathcal{F}$ can support a massive number of NB-IoT devices by prioritizing the NB-IoT devices that require less transmission time to transmit a small amount of job data. Subsequently, the initial residual transmission capacity τ_f^{res} of $f \in \mathcal{F}$ is set to the maximum transmission capacity of the corresponding FSN $f \in \mathcal{F}$ (line 11 in Alg. 2). If the initial residual capacity can support the requirements of $n \in \mathbb{A}_f$, the residual capacity is recalculated and the transmission time scheduling vector $\vec{\tau}_f$ is updated accordingly (lines 13-14 in Alg. 2). However, if the transmission time requirements exceed the residual capacity of the FSN $f \in \mathcal{F}$, the GITS procedure is initiated to iteratively schedule the transmission time for the NB-IoT devices (lines 16-24 in Alg. 2). In this process, the residual capacity is iteratively updated based on the transmission time requirements of the NB-IoT devices $n \in \mathbb{A}_f$. If the transmission time requirements of the NB-IoT device $n \in \mathbb{A}_f$ are met, the residual capacity of FSN $f \in \mathcal{F}$ is recalculated and the

Algorithm 2 Greedy Iterative Time Scheduling (GITS) at Each FSN $f \in \mathcal{F}$

Input: τ_f^{th}, $\mathcal{N}$
1 **Initialize:** Φ, $\mathbb{A}_f$, $\tau_f^{res} \leftarrow 0$
Result: $\vec{\tau}_f$
2 **Step 1: Admission Control**
3 **for** $n \in \mathcal{N}$ **do**
4   **if** *(10) and (12) are fulfilled* **then**
5     $\Phi_{f,n} = 1$
6     $\mathbb{A}_f = \mathbb{A}_f \cup \{\Phi_{f,n}\}$
7   **else**
8     $\Phi_{f,n} = 0$
9 **Step 2: GITS**
10 Sort $\mathbb{A}_f$ based on τ_n in descending order
11 $\tau_f^{res} \leftarrow \tau_f^{th}$
12 **if** $\tau_f^{res} \geq \sum_{n \in \mathbb{A}_f} \tau_{f,n}$ **then**
13   $\tau_f^{res} \leftarrow \tau_f^{th} - \sum_{n \in \mathbb{A}_f} \tau_{f,n}$
14   Update $\vec{\tau}_f$
15 **else**
16   $\mathbb{A}_f^{temp} = \emptyset$, $\mathbb{A}'_f = \emptyset$
17   **while** $\exists n \in \mathbb{A}$ *and* $\exists n \notin \mathbb{A}'_f$ **do**
18   Select $n \in \mathbb{A}$
19   Calculate τ_n using (16)
20   **if** $\tau_f^{res} \leq \tau_n$ *and* $\exists n \notin \mathbb{A}_f^{temp}$ **then**
21       $\tau_f^{res} \leftarrow \tau_f^{res} - \tau_n$
22   Update $\vec{\tau}_f$, $\mathbb{A}_f^{temp}$
23   **else**
24   Update $\mathbb{A}'_f$
25 return $\vec{\tau}_f$

transmission time vector $\vec{\tau}$ and $\mathbb{A}_f^{temp}$ are updated accordingly (lines 20-22 in Alg. 2). Otherwise, NB-IoT device $n \in \mathbb{A}_{temp}$ is not added to the list $\mathbb{A}_f^{temp}$, and is therefore added to the rejection list $\mathbb{A}_f'$ (lines 23-24 in Alg. 2). Finally, the GITS procedure terminates once the NB-IoT devices reside in either $\mathbb{A}_f^{temp}$ or $\mathbb{A}_f'$.

VI. THE VOGEL's APPROXIMATION METHOD FOR INTER-FOG COMPUTATIONAL JOB ASSIGNMENT

The problem of inter-fog computational job assignment is an exponential time problem that cannot be solved in polynomial time. However, the problem **P1** can also be reduced to a base problem of the generalized assignment problem (GAP) [50], which is a classical generalization of a well-known $\mathcal{NP}$-hard problem, namely the bin packing problem [51]. Therefore, the solution of the computational job assignment problem in **P1** is three-fold. First, we determine the sets of underloaded and overloaded FSNs, which are denoted as $\mathcal{F}_u \subset \mathcal{F}$ and $\mathcal{F}_o \subset \mathcal{F}$, respectively. By using (5), the utilization of each FSN $f \in \mathcal{F}$ can be calculated, and if the utilization factor is $u_f > 1$, the corresponding FSN $f \in \mathcal{F}$ belongs to the overloaded set $\mathcal{F}_o \subset \mathcal{F}$. Otherwise, $f \in \mathcal{F}$ belongs to the underloaded set $\mathcal{F}_u \subset \mathcal{F}$. Second, at each FSN $f \in \mathcal{F}_o$, the offloading requests are formed as a binary offloading decision vector $\Psi_f = (\psi_{f,j}), f = 1, \cdots, F$, and $j = 1, \cdots, J$, which is comprised of off-loadable jobs $j \in Q_f$. Therefore, we determine the off-loadable jobs at the FSNs $f \in \mathcal{F}_o$ mentioned in **P1** as

$$\Psi_f = [\psi_{f,j}, \cdots, \psi_{f,J}] = \begin{cases} 1, & \text{if } \lambda_{max} < \lambda_f(j) \\ 0, & \text{otherwise.} \end{cases}, \quad \forall f \in \mathcal{F}_o \tag{21}$$

In (21), we assume that the maximum job computational capacity of the FSNs are limited within an job arrival rate of λ_{max}. Therefore, at each FSN $f \in \mathcal{F}_o$, if the computational load $\lambda_f(j)$ of the job $j \in Q_f$ is greater than the maximum job arrival rate λ_{max}, the job $j \in Q_f$ is off-loadable and added to the vector Ψ_f. Otherwise, the job can be processed locally at $f \in \mathcal{F}_o$. Finally, each overloaded FSN $f \in \mathcal{F}_o$ needs to find an appropriate neighboring underloaded FSN from the set $\mathcal{F}_u$, where the inter-fog job computational job assignment cost in (8) is minimum. From equation (8), we observe that, the inter-fog computational job assignment decision not only depends on the computational capacity of the neighboring underloaded FSNs in $\mathcal{F}_u$, but also on different factors such as the inter-fog job data transmission time, the queuing delay of the neighboring FSN for the existing jobs, and the time to fetch application for offloaded job execution. Moreover, in an online scenario, finding an appropriate neighboring FSN $f' \in \mathcal{F}_u$ for job offloading is challenging due to the fact that different FSNs in $\mathcal{F}$ have the different computational capacities (i.e., the number of residual CRBs), as well as with different incoming job demands from the overloaded FSNs $f \in \mathcal{F}_o$ and existing job demands of the neighboring underloaded FSN $f' \in \mathcal{F}_u$. Therefore, we introduce a new decision variable Θ for the inter-fog job offloading problem in **P1**, which is the job assignment vector used to model the problem as a Hitchcock-Koopmans problem [52]. The linear programming representation of a transportation problem is:

$$\underset{\Theta}{\text{Min}}\ \mathbf{P2} = \sum_{f \in \mathcal{F}_o} \sum_{f' \in \mathcal{F}_u} \theta_{f,f'} \hat{\mathcal{C}}_{f,f'} \tag{22}$$

$$\sum_{f \in \mathcal{F}_o} \theta_{f,f'} \leq \bar{k}_{f'}, \quad \forall f' \in \mathcal{F}_u \tag{23}$$

$$\sum_{f' \in \mathcal{F}_u} \theta_{f,f'} \geq \sum_{f \in \mathcal{F}_o} \Psi_f, \forall f \in \mathcal{F}_o \tag{24}$$

$$\theta_{f,f'} \geq 0, \quad \forall f' \in \mathcal{F}_u, \forall f \in \mathcal{F}_o. \tag{25}$$

The goal in **P2** is to find an $|\mathcal{F}_o| \times |\mathcal{F}_u|$ array of numbers $\Theta = (\theta_{f,f'}), f = 1, \cdots, |\mathcal{F}_o|, f' = 1, \cdots, |\mathcal{F}_u|$, that minimizes the inter-fog job assignment cost $\hat{\mathcal{C}}_{f,f'}$ of filling shortages of CRBs in the overloaded FSNs from the surplus CRBs at the underloaded FSN. In (23), $\bar{k}_{f'}$ is the residual number of idle CRBs at the underloaded FSNs in $\mathcal{F}_u$, which represents the supply constraint. Constraint (24) ensures the off-loadable jobs at the overloaded FSN will be assigned to the underloaded FSN where it represents the demand. In (25), $\theta_{f,f'}$ is a non-negative integer that represents the number of jobs offloaded from the supply point of $f \in \mathcal{F}_o$ (i.e., overloaded FSN) to the destination point of $f' \in \mathcal{F}_u$ (i.e., underloaded FSN). To solve the Hitchcock-Koopmans transportation problem for the computational job assignment, first we apply an iterative procedure known as Vogel's Approximation Method (VAM) [53], [54] which calculates the feasible and optimal solution of the problem. The basic steps of VAM for solving the computational inter-fog job assignment problem are given below.

Step 1: Initialization of the balanced opportunity cost matrix, $\mathbb{O}$ with dimensions $|\mathcal{F}_o| \times |\mathcal{F}_u|$

1(a) If the demand is greater than the supply, the opportunity cost matrix is balanced by adding dummy rows to $\mathbb{O}$.

1(b) If the supply is greater than the demand, the opportunity cost matrix is balanced by adding dummy columns to $\mathbb{O}$.

Step 2:: Calculation of the penalty cost for each row and column by subtracting the lowest cell cost in the row or column from the next lowest cell cost in the same row or column.

Step 3: Determination of the row or column with the largest penalty. Arbitrarily break the tie if the largest penalty and the second largest penalty of the row and the columns are equal.

Step 4: Assignment of the maximum number of jobs from the overloaded FSN in the row to the underloaded FSN with the smallest offloading cost.

4(a) Adjustment of the supply and demand and remove the satisfied rows or columns and set the values to 0.

Step 5: Follow the stopping criteria *5(a)* If exactly one row or column with zero supply or demand remains uncrossed out, then stop.

5(b) Repetition of the steps **2**, **3**, and **4** until conditions (23), (24), and (25) are all fulfilled.

Step 6: Computation of the total transportation cost (i.e., the inter-fog job assignment cost) for the feasible assignments Θ.

VII. SIMULATION RESULTS

Figure 2 depicts a network scenario with $\mathcal{F} = 20$, where the NB-IoT device density per FSN is 10. Within the range of each FSN, the number of NB-IoT devices is uniformly distributed. The number of jobs at each FSN is a random number from the exponential distribution with mean 5. In addition, the simulation results are obtained based on a machine with a 3.00 GHz Intel i5-7400 CPU and 16 GB RAM. The main parameters for the simulation are provided in Table 2.

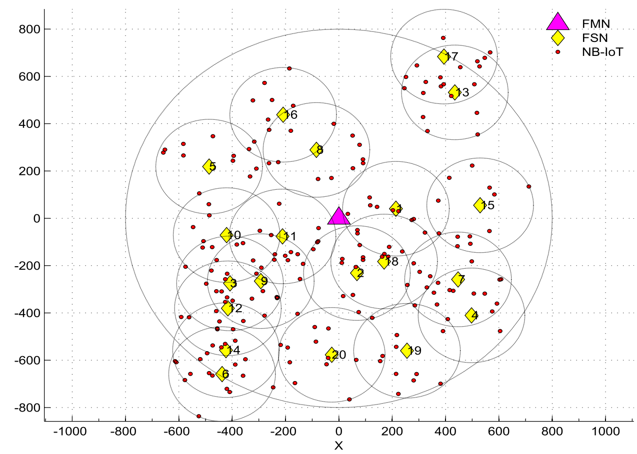

FIGURE 2. An example simulation setup.

A. SIMULATION SETTING

At each complete iteration during the simulation, the network is randomized based on the simulation parameters in Table 2 and the performance of the proposed approach is illustrated. Using the simulation, we compare our proposed approaches with two baseline methods, which are given below.

(1) **Isolated FSN:** In this approach, there is no cooperation between neighboring FSNs, and individual FSNs are considered to be isolated data-centers-in-a-box. In other words, the FSNs are treated as self-sufficient computational resource blocks (CRBs) as the job load increases at each FSN, where the jobs are served locally.

(2) **Nearest Neighbor Offloading:** In the nearest neighbor offloading approach, the neighboring FSNs cooperate among themselves and the over-utilized FSNs offload the excess amount of jobs to their nearest underutilized FSNs.

TABLE 2. Simulation settings.

Simulation Parameters	Values
No. of FSN	$[4, 20]$
No. of CRB/FSN	2
No. of NB-IoT devices	$[40, 200]$
No. of FMN	1
No. of jobs	$[10, 80]$
Radius of FSN	200 (m)
Radius of FMN	800 (m)
System Bandwidth	180 kHz
Carrier frequency	900 MHz
Number of tones	1, fixed
NB-IoT subcarrier spacing	180 kHz, 3.75 kHz subcarrier spacing
NB-IoT 1 RF Radio Frame	10 ms [55]
Channel Model	AWGN
Message Size $\mathcal{L}$	33 bytes [33]
NB-IoT Device Maximum Transmission power κ	10 mW [56]
Noise Spectral Density	10^{-13} (Urban case) [57]
Propagation speed in medium	2×10^8 (m/sec)

B. LOAD BALANCING COST ANALYSIS

In this subsection, we analyze the load balancing cost of different methods in terms of the load balancing cost and queuing delay with the increased number of FSNs and computational jobs in the fog network.

In Fig. 3, we compare the performance of the proposed VAM-based load balancing with the isolated FSN and the nearest neighbor offloading methods in terms of the average cost. At a small number of fog computing nodes in the network (i.e., $|\mathcal{F}| = 4$), the average cost is relatively high for all the methods due to the smaller number of FSNs in the network that provide the computational services for the jobs. However, starting from $|\mathcal{F}| = 4$, placing two additional FSNs in the network decreases the average cost linearly by 40%. Moreover, when the fog computing network size reaches up to $|\mathcal{F}| = 20$ from $|\mathcal{F}| = 4$, the average costs of the isolated FSN and the nearest neighbor offloading are significantly increased by 84.4857% and 81.1062%, respectively compared to that of the proposed VAM-based load balancing solution. As a result, in Fig. 3, we can infer three crucial observations. First, after a certain number of FSN placements (i.e., $|\mathcal{F}| = 14$) in the network, the placement of additional FSNs follows the law of diminishing returns [58] in terms of job load assignment cost minimization. Second, in the case of overloaded FSNs, offloading the excess amount of jobs to the neighboring underloaded FSNs is not the best option for ensuring a well-balanced network. In fact, this is the reason why the proposed VAM approach significantly outperforms the nearest neighbor offloading approach. Unlike the nearest neighbor approach, the offloading decision of the proposed approach considers not only the propagation and transmission costs for load-balancing, but also the system utilization of the underloaded FSNs. Finally, the cooperation between the FSNs in the network can significantly reduce congestion during the computational service by the FSNs. Therefore, the nearest neighbor offloading approach incurs

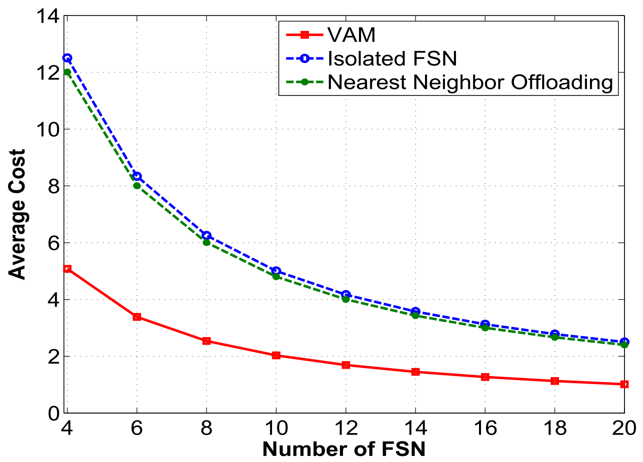

FIGURE 3. Comparison of average cost between the proposed method and the baseline methods, $|\mathcal{N}| = 10$ per FSN.

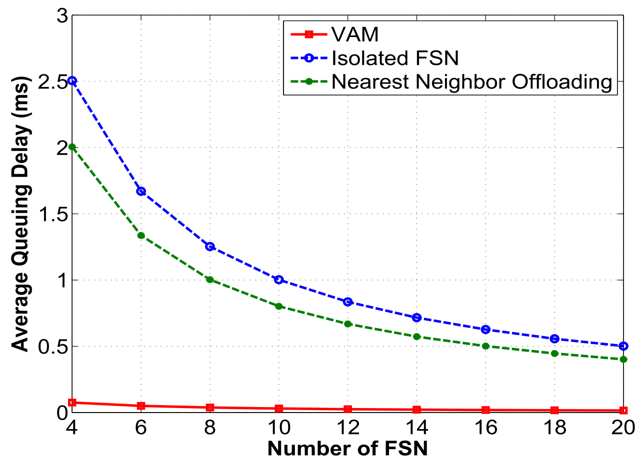

FIGURE 5. Comparison of average delay between different methods, $|\mathcal{N}| = 10$ per FSN, $\mu = 0.125$.

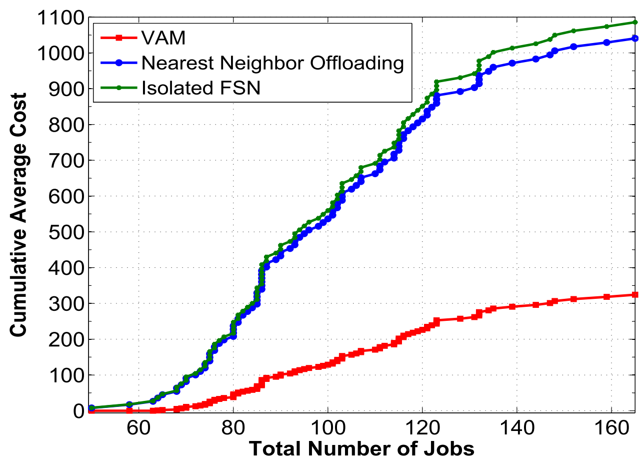

FIGURE 4. Comparison of cumulative average cost between the proposed method and the baseline methods, $|\mathcal{F}| = 20$.

a 4% smaller cost than that of the isolated FSN approach. Overall, the proposed approach is shown to be more efficient than that of the two baseline methods. On the other hand, Fig. 4 depicts the number of FSNs as fixed at $|\mathcal{F}| = 20$ while the number of computational jobs varies between $|\mathcal{J}| = [55, 165]$, the cumulative average cost of the nearest neighbor offloading and the isolated FSN increase drastically by 71.215% and 72.367%, respectively, compared to that of the proposed VAM approach for job assignment.

In Fig. 5, a comparison of the average queuing delay is depicted, where the number of FSNs is given in increasing order. The average delay of the proposed VAM approach is significantly reduced by 97.006% and 96.2595% compared to the isolated FSN and nearest neighbor offloading approaches, respectively. Meanwhile, the queuing delay between the nearest neighbor and the isolated FSN approaches is reduced by 19.957%; this is because in the case of the nearest neighbor, the nearby FSNs cooperate with each other to perform the job inside the FSN network. As a result, the nearest neighbor offloading approach ensures better performance gain than that of the isolated FSN approach. On the other hand, the proposed VAM-based job load balancing approach has a significant advantage over the nearest neighbor approach. Unlike the nearest neighbor offloading approach, the offloading policy of the proposed approach depends on the expected queuing delay of the jobs at the overloaded FSNs, in which can easily offload to the corresponding underloaded FSNs. In addition, the underloaded FSNs only admit jobs that do not exceed their own system capacity. Under these circumstances, the proposed VAM approach significantly outperforms the nearest neighbor approach. In the case of the isolated FSN approach, due to the increased network size and lack of interactions between nearby FSNs, some of the FSNs become highly congested as the number of jobs increases. On the other hand, the underutilized FSNs remain inactive and do not contribute to the network, so that average delay of different jobs can be reduced. In fact, the job load discrepancy between the overloaded FSNs and underloaded FSNs increases significantly compared to that of the proposed VAM approach. This proves the efficiency of the proposed approach, particularly that the computational capacity of the network is fully utilized and the job load is well-balanced. Overall, from Fig. 5 we can also verify that the proposed approach is effective in enhancing the quality of service (QoS) for different jobs compared to the other two baseline approaches.

In Fig. 6, we also analyze the effects of increased job load when the number of FSNs in the network is fixed at $|\mathcal{F}| = 20$. From Fig. 6, it is noticeable that the cumulative average delay increases sharply in the case of baseline methods compared to the proposed approach. In fact, with an increasing number of jobs, the cumulative average delay results of the isolated FSNs and the nearest neighbor offloading are respectively 96.7097%, and 95.8902% higher than that of the proposed VAM approach. This clearly proves the fact that, the proposed VAM approach is more suitable in real environments where

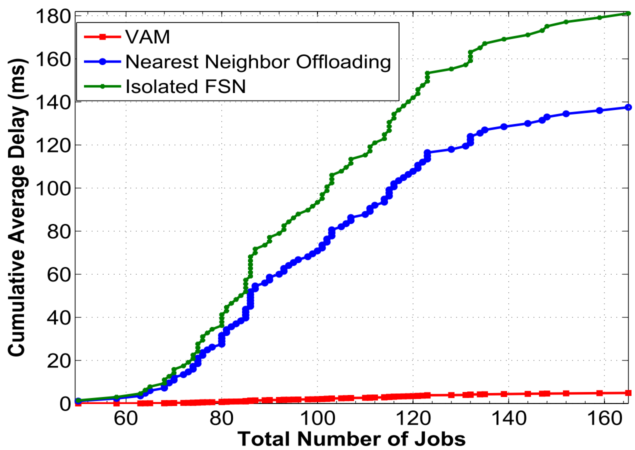

FIGURE 6. Comparison of cumulative average delay between the proposed method and the baseline methods, $|\mathcal{F}| = 20$.

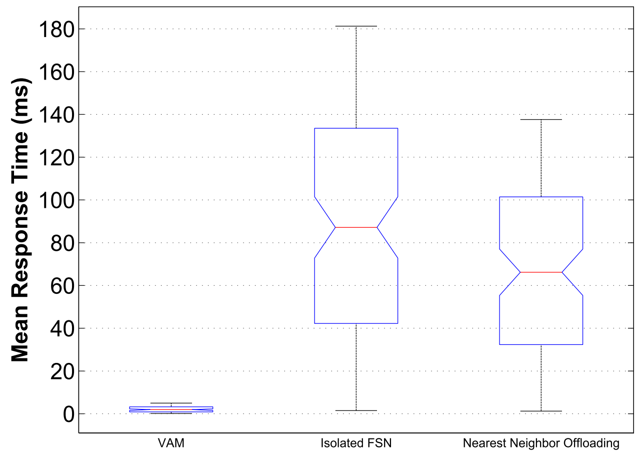

FIGURE 8. Comparison of mean response time between the VAM, Isolated FSN, and Nearest neighbor offloading approaches, $|\mathcal{F}| = 20$, $|\mathcal{J}| = [50, 165]$.

the number of computational jobs may increase or decrease drastically.

C. LOAD BALANCING BENEFIT ANALYSIS

In this subsection, we analyze the achieved gain or benefit of computational job assignment between the different methods in terms of average response time with increasing number of FSNs and computational jobs in the fog network.

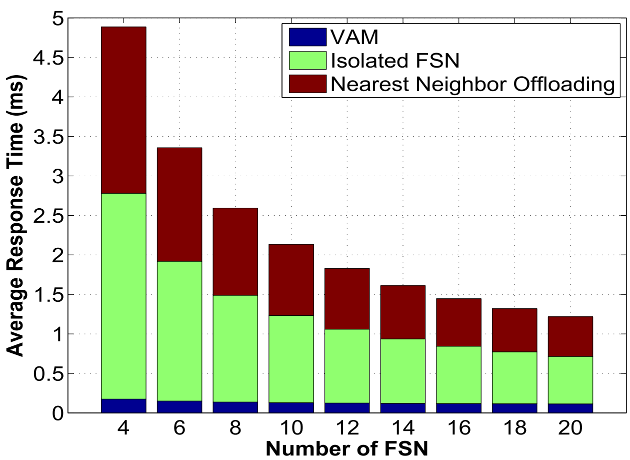

FIGURE 7. Comparison of average response time between different methods, $|\mathcal{N}| = 10$, $\mu = 0.0625$.

In Fig. 7, the average response time between the different approaches are evaluated, where the service rate of the FSNs is set to $\mu = 0.125$. From the simulation, we observe that as the network size increases, the proposed VAM approach is proven to be more effective than that of the other approaches in terms of the average response time. When the network size is $|\mathcal{F}| = 20$, we observe that the average response time of the proposed approach is reduced by 80% compared to the network size $|\mathcal{F}| = 4$. The reason behind this is that as the network size increases, the number of off-loadable FSNs also increases, which leads to more options for the overloaded FSNs to offload jobs to the underloaded FSNs. In the case of the nearest neighbor approach, when the network size is $|\mathcal{F}| = 4$, the average response time is at a maximum as compared to when the network size is $\mathcal{F} = 20$. If the network size is $|\mathcal{F}| > 12$, the average response time gradually decreases in nearest neighbor approach. However, the average response time of the nearest neighbor approach is still 96.2595% higher than that of the proposed VAM approach when the number of FSNs in the network is $|\mathcal{F}| = 20$. The average response time is significantly reduced for the proposed VAM approach compared to that of the other two baseline methods which clearly correspond to the fairness of the job load balancing process among the FSNs. In fact, the efficiency of job load balancing is defined for the individual FSNs where the proposed VAM approach can significantly optimize the job assignment decision to ensure a reduced average response time under different network sizes than that of the baseline methods. On the other hand, in Fig. 8, we observe that the cumulative average response time of job offloading through the proposed VAM approach is significantly reduced by 97.9618% and 97.4878%, respectively compared to that of the isolated FSN and nearest neighbor offloading approaches. This clearly demonstrates that the proposed VAM approach is less sensitive to the increased job load after placing a certain number of FSNs in the network than that of the other two baseline methods. Moreover, the performance gap between the nearest neighbor offloading and the isolated FSN approach is small even with a different amount of computational load in the network. The reason for this is that both the nearest neighbor offloading and the isolated FSN approaches are very sensitive to drastic increases in computational jobs in the network, and thus the performance varies noticeably compared to the proposed VAM approach.

D. COMPUTATIONAL TIME ANALYSIS

In this subsection, we perform a computational time analysis between the different methods with an increased number of FSNs and computational jobs in the fog network.

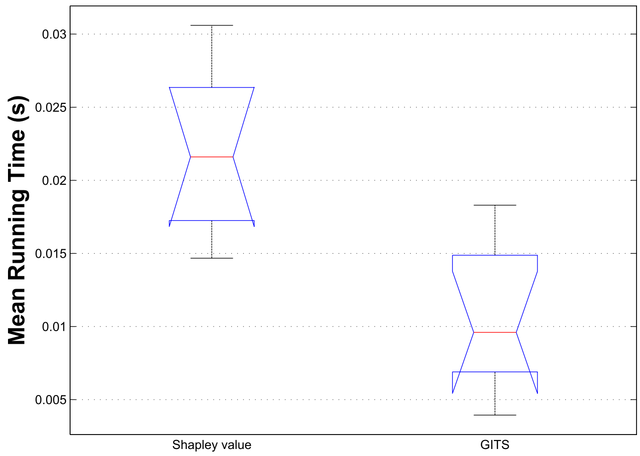

FIGURE 9. Comparison of computation times between the Shapley value and GITS algorithms, $|\mathcal{F}| = [4, 20]$, $|\mathcal{N}| = [40, 220]$.

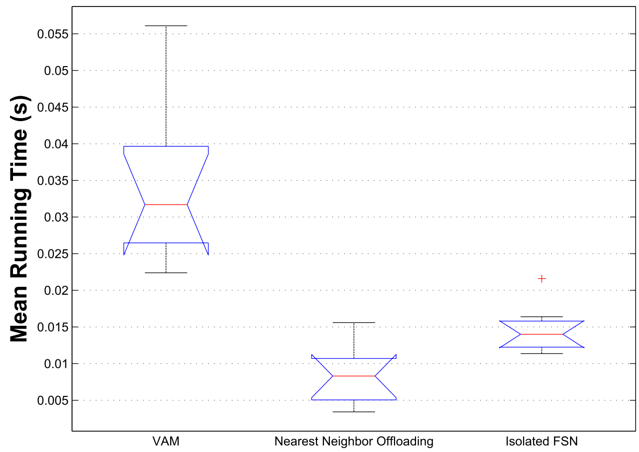

FIGURE 10. Comparison of running times between the VAM, Nearest neighbor offloading, and Isolated FSN approaches, $|\mathcal{F}| = [4, 20]$.

Figure 9 depicts a computational time comparison between computing the Shapley value and utilizing the greedy iterative time scheduling (GITS) approach, where the number of FSNs is $|\mathcal{F}| = [4, 20]$ and NB-IoT devices are distributed randomly throughout the network. We repeated the simulation 100 times to simulate the user mobility in the network. We also recorded the maximum and minimum computational times for both algorithms to show the range of values for the algorithmic running time for both proposed algorithms. From Fig. 9, we observe, the running times of both approaches increase sharply as the number of NB-IoT devices increases. Furthermore, the mean running time of the Shapley value and GITS approaches are 0.0197 seconds and 0.0159 seconds, respectively. In fact, the computational complexity of the Shapley value is 19.289% larger than that of the GITS approach. As a result, we can infer that the Shapley value-based time scheduling approach is more sensitive compared to the GITS approach in terms of user mobility. However, the running time of the Shapley value is still tolerable and practical due to the admission control policy at different FSNs, which is based on the channel conditions and energy resource availability at the corresponding NB-IoT devices. In fact, the bankruptcy game is enforced distributedly at different FSNs where the number of the served NB-IoT devices is limited due to the admission control-based NB-IoT device association policy. Nevertheless, the running time can also be significantly reduced if the Shapley value is pre-computed before transmission time allocation. In real network scenarios, this is usually accomplished by the FSNs or the FMN at a fixed time interval to meet the shifting dynamics of the NB-IoT devices. Thus, both proposed approaches are proven to be effective at ensuring adaptive transmission time scheduling depending on the number of NB-IoT device arrival rates. In Fig. 10, the average running time of the VAM approach is given by the increasing job arrivals at the FSNs. The mean running time of the proposed VAM approach is 0.034 seconds whereas the running times of the isolated FSN

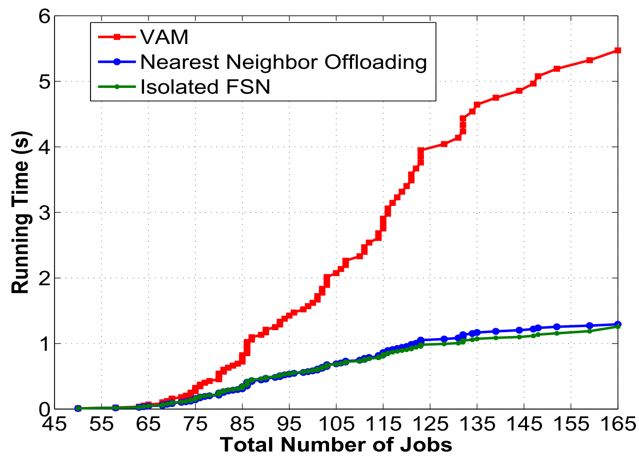

FIGURE 11. Comparison of mean computation time between the VAM, Isolated FSN, and Nearest neighbor offloading approaches, $|\mathcal{F}| = 20$.

and nearest neighbor offloading approaches are 0.0146 seconds and 0.0087 seconds, respectively. The running time of the proposed VAM approach is 57.0588% and 74.4118% higher, respectively, than that of the isolated FSN and nearest neighbor offloading approaches. Moreover, as the job load increases, the number of iterations to optimize the job assignment cost increases. This is expected since the proposed VAM provides the optimal job assignment decision with increasing number of job demands, achieving significant performance gain compared to the other baseline methods. Since the proposed approach is applied in the FMN, which has a higher computational capacity than the FSNs, the FMN can meet the computational time requirements to ensure job load-balancing for the FSNs. In addition, from Fig. 11, we observe that with increasing job load and a fixed network size of $|\mathcal{F}| = 20$, the mean running time of the proposed VAM approach is 0.0543 seconds, whereas the running times of the isolated FSN and nearest neighbor offloading approaches are 0.0128 seconds and 0.0108 seconds, respectively.

The running time of the proposed VAM approach is 76.427% and 80.110% higher, respectively, than that of the isolated FSN and nearest neighbor offloading approaches.

VIII. CONCLUSION

In this paper, we focus on the inter-fog load balancing problem, where the objective is to minimize the computational costs with respect to communication and computation constraints in the fog network empowered with the NB-IoT wireless technology. Therefore, we proposed a Bankruptcy game framework for distributive and strategic time scheduling that is well-suited for data acquisition from NB-IoT devices in a fog computing network. Unlike conventional approaches, the proposed Shapley value-based approach can fairly schedule the limited time resources provided by the NB-IoT carrier to the NB-IoT devices. We also proposed a relatively lightweight greedy transmission time scheduling algorithm that complements the Shapley value-based time scheduling approach. Further, we formulated the inter-fog load offloading problem as a Hitchcock-Koopmans problem and solved the problem using Vogel's approximation method for load balancing among the cooperative fog computing nodes so that the inter-fog offloading cost is optimized efficiently. Finally, we validated the performance of the proposed approaches through extensive numerical analyses in terms of different performance metrics.

REFERENCES

[1] I. Farris, T. Taleb, H. Flinck, and A. Iera, "Providing ultra-short latency to user-centric 5G applications at the mobile network edge," *Trans. Emerg. Telecommun. Technol.*, vol. 29, no. 4, p. e3169, 2018.

[2] A. K. Bairagi, S. F. Abedin, N. H. Tran, D. Niyato, and C. S. Hong, "Qoe-enabled unlicensed spectrum sharing in 5G: A game-theoretic approach," *IEEE Access*, vol. 6, pp. 50538–50554, 2018.

[3] S. Li, L. Da Xu, and S. Zhao, "5G Internet of Things: A survey," *J. Ind. Inf. Integr.*, vol. 10, pp. 1–9, Jun. 2018.

[4] J. Lee and J. Lee, "Prediction-based energy saving mechanism in 3GPP nb-IoT networks," *Sensors*, vol. 17, no. 9, p. 2008, 2017.

[5] L. Da Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.

[6] 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.*, 2012, pp. 13–16.

[7] J. Gozalvez, "New 3GPP standard for IoT [mobile radio]," *IEEE Veh. Technol. Mag.*, vol. 11, no. 1, pp. 14–20, Mar. 2016.

[8] 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–Pacific Netw. Oper. Manage. Symp. (APNOMS)*, Aug. 2015, pp. 309–314.

[9] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[10] Y.-P. E. Wang *et al.*, "A primer on 3GPP narrowband Internet of Things," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 117–123, Mar. 2017.

[11] Y.-C. P. Chang, S. Chen, T.-J. Wang, and Y. Lee, "Fog computing node system software architecture and potential applications for NB-IoT industry," in *Proc. Int. Comput. Symp. (ICS)*, 2016, pp. 727–730.

[12] S. F. Abedin, M. G. R. Alam, S. M. A. Kazmi, N. H. Tran, D. Niyato, and C. S. Hong, "Resource allocation for ultra-reliable and enhanced mobile broadband IoT applications in fog network," *IEEE Trans. Commun.*, vol. 67, no. 1, p. 1, Sep. 2018.

[13] G. Lee, W. Saad, and M. Bennis, "An online secretary framework for fog network formation with minimal latency," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[14] H. Zhang, Y. Xiao, S. Bu, D. Niyato, F. R. Yu, and Z. Han, "Computing resource allocation in three-tier IoT fog networks: A joint optimization approach combining Stackelberg game and matching," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1204–1215, Oct. 2017.

[15] Y. Gu, Z. Chang, M. Pan, L. Song, and Z. Han, "Joint radio and computational resource allocation in IoT fog computing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 7475–7484, Aug. 2018.

[16] M. S. Munir, S. F. Abedin, M. G. R. Alam, N. H. Tran, and C. S. Hong, "Intelligent service fulfillment for software defined networks in smart city," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2018, pp. 516–521.

[17] M. Altamimi, A. Abdrabou, K. Naik, and A. Nayak, "Energy cost models of smartphones for task offloading to the cloud," *IEEE Trans. Emerg. Topics Comput.*, vol. 3, no. 3, pp. 384–398, Sep. 2015.

[18] X. Zhang, A. Kunjithapatham, S. Jeong, and S. Gibbs, "Towards an elastic application model for augmenting the computing capabilities of mobile devices with cloud computing," *Mobile Netw. Appl.*, vol. 16, no. 3, pp. 270–284, 2011.

[19] C. Shi, M. H. Ammar, E. W. Zegura, and M. Naik, "Computing in cirrus clouds: The challenge of intermittent connectivity," in *Proc. 1st Ed. MCC Workshop Mobile Cloud Comput.*, 2012, pp. 23–28.

[20] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," *Wireless Commun. Mobile Comput.*, vol. 13, no. 18, pp. 1587–1611, Dec. 2013.

[21] D. H. Tran, N. H. Tran, C. Pham, S. A. Kazmi, E.-N. Huh, and C. S. Hong, "OaaS: Offload as a service in fog networks," *Computing*, vol. 99, no. 11, pp. 1081–1104, 2017.

[22] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing—A key technology towards 5G," ETSI, Sophia Antipolis, France, White Paper 11, 2015, pp. 1–16.

[23] X. Lin, Y. Wang, Q. Xie, and M. Pedram, "Energy and performance-aware task scheduling in a mobile cloud computing environment," in *Proc. IEEE 7th Int. Conf. Cloud Comput. (CLOUD)*, Jun. 2014, pp. 192–199, doi: 10.1109/CLOUD.2014.35.

[24] M. G. R. Alam, Y. K. Tun, and C. S. Hong, "Multi-agent and reinforcement learning based code offloading in mobile fog," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, 2016, pp. 285–290.

[25] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.

[26] M. Jia, J. Cao, and W. Liang, "Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks," *IEEE Trans. Cloud Comput.*, vol. 5, no. 4, pp. 725–737, Oct./Dec. 2017.

[27] J. Oueis, E. C. Strinati, and S. Barbarossa, "The fog balancing: Load distribution for small cell cloud computing," in *Proc. IEEE 81st Veh. Technol. Conf. (VTC Spring)*, May 2015, pp. 1–6.

[28] T. Verbelen, P. Simoens, F. De Turck, and B. Dhoedt, "Leveraging cloudlets for immersive collaborative applications," *IEEE Pervasive Comput.*, vol. 12, no. 4, pp. 30–38, Oct. 2013.

[29] S. F. Abedin, M. G. R. Alam, R. Haw, and C. S. Hong, "A system model for energy efficient green-IoT network," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, 2015, pp. 177–182.

[30] C. T. Do, N. H. Tran, C. Pham, M. G. R. Alam, J. H. Son, and C. S. Hong, "A proximal algorithm for joint resource allocation and minimizing carbon footprint in geo-distributed fog computing," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, 2015, pp. 324–329.

[31] C. Sarkar, S. N. Nambi, R. V. Prasad, and A. Rahim, "A scalable distributed architecture towards unifying IoT applications," in *Proc. IEEE World Forum Internet Things (WF-IoT)(WF-IOT)*, Mar. 2014, pp. 508–513, doi: 10.1109/WF-IoT.2014.6803220.

[32] C. Yu, L. Yu, Y. Wu, Y. He, and Q. Lu, "Uplink scheduling and link adaptation for narrowband Internet of Things systems," *IEEE Access*, vol. 5, pp. 1724–1734, 2017.

[33] S.-M. Oh and J. Shin, "An efficient small data transmission scheme in the 3GPP NB-IoT system," *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 660–663, Mar. 2017.

[34] W. Yang *et al.*, "Enhanced system acquisition for NB-IoT," *IEEE Access*, vol. 5, pp. 13179–13191, 2017.

[35] M. R. Palattella *et al.*, "Internet of Things in the 5G era: Enablers, architecture, and business models," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 3, pp. 510–527, Mar. 2016.

[36] *Openfog Reference Architecture for Fog Computing*, document 2OPFRA001.02081, OpenFog Consortium Architecture Working Group, Fremont, CA, USA, 2017, p. 162.

[37] S. Moon, T. Z. Oo, B. J. Park, and C. S. Hong, "SDN-based self-organizing energy efficient downlink/uplink scheduling in heterogeneous cellular networks," *IEICE Trans. Inf. Syst.*, vol. 100, no. 5, pp. 939–947, 2017.
[38] V. Dedeoglu, S. Perreau, and A. Grant, "Cross-layer energy minimization in correlated data gathering wireless sensor networks," in *Proc. IEEE 13th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jun. 2012, pp. 304–308.
[39] W. Yang *et al.*, "Narrowband wireless access for low-power massive Internet of Things: A bandwidth perspective," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 138–145, Jun. 2017.
[40] N. Mangalvedhe, R. Ratasuk, and A. Ghosh, "NB-IoT deployment study for low power wide area cellular IoT," in *Proc. IEEE 27th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2016, pp. 1–6.
[41] S. Cui, A. J. Goldsmith, and A. Bahai, "Energy-constrained modulation optimization," *IEEE Trans. Wireless Commun.*, vol. 4, no. 5, pp. 2349–2360, Sep. 2005.
[42] C. Chembe, I. Ahmedy, R. M. Noor, and D. Kunda, "Impact of multipath fading on spectrum sensing in vehicular communication environment," in *Proc. 6th Int. Conf. Commun. Broadband Netw.*, 2018, pp. 48–52.
[43] D. Gamarnik and D. A. Goldberg, "On the rate of convergence to stationarity of the M/M/N queue in the Halfin–Whitt regime," *Ann. Appl. Probab.*, vol. 23, no. 5, pp. 1879–1912, 2013.
[44] M. Jia, W. Liang, Z. Xu, and M. Huang, "Cloudlet load balancing in wireless metropolitan area networks," in *Proc. 35th Annu. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2016, pp. 1–9.
[45] M. Jia, W. Liang, Z. Xu, M. Huang, and Y. Ma, "Qos-aware cloudlet load balancing in wireless metropolitan area networks," *IEEE Trans. Cloud Comput.*, to be published.
[46] K. Nag and M. Helal, "Evaluating erlang c and erlang a models for staff optimization: A case study in an airline call center," in *Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage. (IEEM)*, Dec. 2017, pp. 1–5.
[47] M. S. Elbamby, M. Bennis, and W. Saad, "Proactive edge computing in latency-constrained fog networks," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, 2017, pp. 1–6.
[48] S. C. Littlechild and G. Owen, "A simple expression for the shapley value in a special case," *Manage. Sci.*, vol. 20, no. 3, pp. 370–372, 1973.
[49] D. Niyato and E. Hossain, "A cooperative game framework for bandwidth allocation in 4G heterogeneous wireless networks," in *Proc. IEEE Int. Conf. Commun.*, vol. 9, Jun. 2006, pp. 4357–4362.
[50] D. G. Cattrysse and L. N. Van Wassenhove, "A survey of algorithms for the generalized assignment problem," *Eur. J. Oper. Res.*, vol. 60, no. 3, pp. 260–272, Aug. 1992.
[51] S. O. Krumke and C. Thielen, "The generalized assignment problem with minimum quantities," *Eur. J. Oper. Res.*, vol. 228, no. 1, pp. 46–55, 2013.
[52] L. Ford, Jr., and D. R. Fulkerson, "Solving the transportation problem," *Manage. Sci.*, vol. 3, no. 1, pp. 24–32, 1956.
[53] H. H. Shore, "The transportation problem and the Vogel approximation method," *Decis. Sci.*, vol. 1, nos. 3–4, pp. 441–457, 1970.
[54] H. Bulut, "Multiloop transportation simplex algorithm," *Optim. Methods Softw.*, vol. 32, no. 6, pp. 1206–1217, 2017.
[55] M. Chen, Y. Miao, Y. Hao, and K. Hwang, "Narrow band internet of things," *IEEE Access*, vol. 5, pp. 20557–20577, 2017.
[56] Y. Miao, W. Li, D. Tian, M. S. Hossain, and M. F. Alhamid, "Narrowband Internet of Things: Simulation and modeling," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2304–2314, Aug. 2017.
[57] S. Persia and L. Rea, "Next generation M2M cellular networks: LTE-MTC and NB-IoT capacity analysis for smart grids applications," in *Proc. Int. Annu. Conf. (AEIT)*, 2016, pp. 1–6.
[58] R. W. Shephard and R. Färe, "The law of diminishing returns," *Zeitschrift Nationalökonomie*, vol. 34, nos. 1–2, pp. 69–90, 1974.

SARDER FAKHRUL ABEDIN (S'18) received the B.S. degree in computer science from Kristianstad University, Kristianstad, Sweden, in 2013. He is currently pursuing the Ph.D. degree in computer science and engineering with Kyung Hee University, South Korea. His research interests include the Internet of Things network management, cloud computing, fog computing, and wireless sensor networks. He is a member of KIISE. He was a recipient of a scholarship for his graduate study at Kyung Hee University, in 2014.

ANUPAM KUMAR BAIRAGI (S'17–M'18) received the B.Sc. and M.Sc. degrees in computer science and engineering from Khulna University (KU), Bangladesh. He is currently pursuing the Ph.D. degree with Kyung Hee University, South Korea. He is also a Faculty Member in computer science and engineering with KU. His research interests include resource management in LTE-U, cooperative communication, and game theory. He received a scholarship for his Ph.D. degree, in 2014.

MD. SHIRAJUM MUNIR received the B.S. degree in computer science and engineering from Khulna University, Bangladesh, in 2010. He is currently pursuing the Ph.D. degree in computer science and engineering with Kyung Hee University, South Korea. He served as a Lead Engineer with Solution Laboratory, Samsung R&D Institute, Bangladesh, from 2010 to 2016. His research interests include the IoT network management, fog computing, mobile edge computing, software-defined networking, smart grid, and machine learning.

NGUYEN H. TRAN (S'10–M'11–SM'18) received the B.S. degree in electrical and computer engineering from the Hochiminh City University of Technology, in 2005, and the Ph.D. degree in electrical and computer engineering from Kyung Hee University, South Korea, in 2011. He was an Assistant Professor with the Department of Computer Science and Engineering, Kyung Hee University, from 2012 to 2017. Since 2018, he has been with the School of Computer Science, The University of Sydney, where he is currently a Senior Lecturer. His research interest is to apply analytic techniques of optimization, game theory, and stochastic modeling to cutting-edge applications, such as cloud and mobile edge computing, data centers, heterogeneous wireless networks, and big data for networks. He received the Best KHU Thesis Award in Engineering, in 2011, and the Best Paper Award at the IEEE ICC 2016. He has been the Editor of the IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, since 2016. He served as the Editor of the 2017 Newsletter of the Technical Committee on Cognitive Networks on Internet of Things.

CHOONG SEON HONG (S'95–M'97–SM'11) received the B.S. and M.S. degrees in electronic engineering from Kyung Hee University, Seoul, South Korea, in 1983 and 1985, respectively, and the Ph.D. degree from Keio University, Japan, in 1997. Since 1993, he has been with Keio University. In 1988, he joined KT, where he was involved in broadband networks, as a Member of Technical Staff. He was with the Telecommunications Network Laboratory, KT, as a Senior Member of Technical Staff and as the Director of the Networking Research Team, until 1999. Since 1999, he has been a Professor with the Department of Computer Engineering, Kyung Hee University. His research interests include future Internet, ad hoc networks, network management, and network security. He is a member of the ACM, IEICE, IPSJ, KIISE, KICS, KIPS, and OSIA. He has served as the General Chair, the TPC Chair/Member, or an Organizing Committee Member for international conferences, such as NOMS, IM, APNOMS, E2EMON, CCNC, ADSN, ICPP, DIM, WISA, BcN, TINA, SAINT, and ICOIN. He is currently an Associate Editor of the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, the *International Journal of Network Management*, and the IEEE JOURNAL OF COMMUNICATIONS AND NETWORKS, and an Associate Technical Editor of the *IEEE Communications Magazine*.