FogPrime: Dynamic Pricing-Based Strategic Resource Management in Fog Networks

Subhas Chandra Misra† and Ayan Mondal‡, *Member, IEEE*

***Abstract*—In this paper, the problem of strategic resource management in fog networks is discussed while considering a pay-per-use model, similar to that used in cloud. Fog networks are distributed in nature, because of which resource management in these networks is an NP-hard problem. In the existing literature, the researchers focused on resource management in fog networks, while considering the network delay constraint. However, none of these works considered the effect of pricing policy while deciding on resource allocation. Hence, there is a need for pricing-based resource management in fog networks. In this work, we proposed a dynamic pricing-based resource allocation scheme, named FogPrime, for analyzing the trade-off between the service delay and the associated price. In FogPrime, we use dynamic coalition-formation game to decide the resource allocation strategy locally within a cluster. On the other hand, we use utility game to choose the fog nodes, strategically, while considering the aforementioned trade-off. Through simulation, we observed that FogPrime outperforms the existing schemes in terms of satisfaction of the involved entities – the end-user and the fog nodes. Using FogPrime, the satisfaction of the end-users and the fog nodes increases by 24.49–47.82%, respectively. Additionally, we observe that FogPrime ensures an even distribution of profit among the fog nodes and enables the end-users to pay less at most by 15.88–47.27%.**

***Index Terms*—Fog Computing, Game theory, Pricing, Dynamic Coalition formation, Utility game, Offloading.**

I. Introduction

With the advent of the Internet of Things (IoT), fog networks have become popular as there is a trend to shift the computing resources from cloud to the edge networks [1]. In fog networks, the computing resources, including storage and network, are placed at the edge, thereby reducing the latency in serving the applications. In fog networks, the applications are deployed closest to the fog nodes to ensure reduced delay. Due to this, the load on the fog nodes depends on the user density. Additionally, we observe that a subset of nodes of the fog network is oversubscribed, which degrades the performance of the fog networks. On the other hand, fog network extends the cloud architecture to the edge [2]. Hence, we argue that the service provisioning in fog networks also abides by the properties of cloud architecture. In other words, the service provisioning in fog networks needs to follow the pay-per-use model. However, in the existing literature, the researchers did not focus on the business model of fog networks.

Fog networks provide the platform to support the application requirements of the end-users. In the existing literature, the researchers considered the presence of a single cluster in a fog network and proposed models to optimize resource utilization while focusing on minimizing the latency in provisioning services [3]. The existing literature focused on different aspects of fog networks such as reliability, the capacity of the fog nodes, and delay [4]. However, no work exists on the business perspective of fog networks. In the presence of multiple clusters in the decentralized fog networks, the competition among the clusters gives rise to an 'oligopolistic' market scenario. Additionally, we argue that fog networks also support the 'pay-per-use' model, which necessitates the designing of resource management schemes for these networks. The fog networks are distributed in nature. Hence, the resource management schemes designed for cloud infrastructure cannot be used for fog networks.

In this work, we propose FogPrime, a scheme based on coalition formation and utility game-theoretic approaches, to handle the problem of dynamic pricing-based resource management in fog networks. We consider that, within a cluster, the fog nodes are cooperative. However, the individual clusters are non-cooperative. On the other hand, the end-users aim to achieve a trade-off between the price paid and delay in services. In FogPrime, we consider that each cluster comprises of a single master node and multiple fog nodes. Initially, the end-users inform the clusters available about the application requirements. Thereafter, the master nodes evaluate the fog nodes that satisfy the requirements of the end-users. Accordingly, the master nodes decide the local equilibrium subset of fog nodes and inform the end-users. Thereafter, each end-user decides the global equilibrium subset of fog nodes while minimizing the price to be paid and the associated service delay. In summary, specific *contributions* are as follows:

1) We propose a dynamic pricing-based resource management scheme, named FogPrime, for fog networks, while considering user satisfaction. We evaluate user satisfaction as the trade-off between the price to be paid and the associated service delay.

2) In FogPrime, the problem of resource management in fog networks is divided into two parts — (a) local strategic resource management within a cluster, i.e., an equilibrium subset of fog nodes within a cluster, and (b) global strategic resource management.

a) To identify the equilibrium subset of fog-nodes, locally, we use a dynamic coalition-formation cooperative game with transferable utility. This is to be executed by each master node in the cluster to find the equilibrium subset of fog nodes, locally.

† Subhas Chandra Misra is with Industrial and Management Engineering, Indian Institute of Technology Kanpur, India (Email: subhasm@iitk.ac.in).
‡ Ayan Mondal is with Computer Science, Univ Rennes, Inria, CNRS, IRISA, France. (Email: ayan.mondal@irisa.fr).

b) We use utility game to find the equilibrium subset of fog nodes, globally. This is to be performed by the end-users to obtain the equilibrium subset of fog nodes while ensuring the trade-off between the price to be paid and the associated delay.

3) We present three different algorithms to ensure optimal resource management in fog networks while choosing an equilibrium subset of fog nodes.

II. Related Works

In the existing literature, resource management in fog networks mainly focused on latency minimization, viz. [12]–[14]. Some of the works are discussed here. Abedin *et al.* [12] and Zhao *et al.* [15] studied resource management in fog networks considering the QoS requirements in ultra-reliable low latency communications and enhanced mobile broadband services. The authors formulated a joint user association and resource allocation scheme using a two-sided matching game to ensure a stable association between the fog network infrastructure (i.e., fog devices) and IoT devices. Similarly, Zhang *et al.* [11] proposed a resource allocation scheme for fog networks, while formulating it as a joint optimization problem. Nguyen *et al.* [13] proposed a resource allocation scheme as a buyer-seller game where the services act as buyers and fog resources act as divisible goods. The authors evaluated the equilibrium for every service while designing the problem to a convex problem. Akram *et al.* [3] evaluated different load balancing schemes such as Round Robin, Throttled, Active Virtual Machine, Particle Swarm Optimization, Ant Colony Optimization, and odds algorithm for resource allocation in fog networks in the context of smart grid. The authors showed that Particle Swarm Optimization is most suitable for resource allocation among the aforementioned techniques. Name *et al.* [14] designed schemes for seamless handover of mobile IoT devices in the context of fog networks. In this work, the authors also considered the presence of cloud data centers in the presence of fog networks. Shaik and Baskiyar [16] considered the multi-layered fog architecture with different parameters such as physical location, resources, privacy, and security while allocating resources in the fog networks. Some of the works in the existing literature focused on the pricing-based resource allocation for heterogeneous services in fog networks. Farooq and Zhu [9] proposed a price-based virtual memory allocation scheme in which the authors considered homogeneous services. On the other hand, Bandyopadhyay *et al.* [10] designed a pricing-based resource allocation scheme for fog networks in the presence of a single service provider. Hence, these aforementioned approaches cannot handle price-based resource allocation for heterogeneous services in fog networks in the presence of multiple service providers.

In another work, Javaid *et al.* [17] used fog networks for effective resource management in smart buildings while ensuring low latency and high reliability. Vasconcelos *et al.* [1] proposed to use a learning approach to decide the resource allocation to support mobile devices. The authors showed that fog networks help in the reduction of latency for provisioning services to IoT devices. Similarly, Xiang *et al.* [18] proposed a radio access network-based slicing scheme for fog using a deep reinforcement learning while optimizing the caching and evaluated for two scenarios – hotspot and vehicle-to-infrastructure. Another radio access network-based content sharing scheme is proposed by Yan *et al.* [19] for non-orthogonal multiple access-enabled fog while using different optimization approaches such as game theory and machine learning. On the other hand, Zamil *et al.* [20] studied the false alarm detection using the hidden Markov model in the context of broadcasting false content in the fog networks. Zhou *et al.* [21] focused on minimizing the network delay using a matching game while offloading the computation task to the fog nodes from the vehicular. The authors studied vehicular ad-hoc networks in the context of fog networks. Du *et al.* [22], [23] proposed a computation offloading scheme for fog networks using the backbone of cloud infrastructure while ensuring the fairness of the users and threshold delay of requested service. The authors used mixed-integer non-linear programming to design the scheme.

On the other hand, the researchers in the existing literature, also designed schemes for price-based resource allocation in cloud. Some of the works are discussed here. Ben Halima *et al.* [5] proposed a linear programming-based optimization model for resource allocation in cloud. The authors considered that optimization is performed centrally in cloud. Chakraborty *et al.* [6] proposed a pricing model-based resource allocation scheme in sensor-cloud while considering the trustworthiness of the service entity. Another game-theoretic resource allocation scheme is designed by Misra *et al.* [7] for sensor-cloud. The authors considered that the entities allocate cloud resources to the end-users based on the price decided by the centralized entity. Misra and Chakraborty designed another resource allocation scheme in Ref. [24] for sensor-cloud. Aazam *et al.* [8] studied an on-demand resource allocation scheme based on historical records. The aforementioned scheme also follows a centralized approach. However, as these schemes follow a centralized approach, these cannot be applied to distributed fog networks.

Synthesis: In the existing literature, the researchers studied different aspects of fog networks. In some of the works, the authors focused on the mobility issue and its implications in the context of fog networks. However, none of these works considered the pay-per-use model in the fog networks. Similar to cloud infrastructure, we argue that the pricing policies will play an integral role in resource allocation in fog networks. Moreover, the resource allocation schemes designed for cloud environment follow a centralized approach and cannot be applied for distributed fog networks, as mentioned in Table I. Hence, there is a need for a pricing-based resource allocation scheme in fog networks while considering that the service provisioning in fog networks follows the pay-per-use model.

III. System Model

In this work, we consider a fog network-based system comprising of multiple fog nodes and multiple end-users. The fog nodes are deployed over a geographical region and they form clusters. In each cluster, a single fog node acts as the coordinator, named as '*master*' node, as shown in Figure 1.

TABLE I: Comparison of Pricing-Based Resource Allocation Schemes

Price-Based Resource Allocation	IoT-Enabled Fog Network Features				
	Decentralized	**Heterogeneous**	**Dynamic Configuration**	**Fog**	**Cloud**
Ben Halima *et al.* [5]	✗	✗	✓	✗	✓
Chakraborty *et al.* [6], Misra *et al.* [7]	✗	✓	✓	✗	✓
Aazam *et al.* [8]	✗	✗	✗	✗	✓
Farooq and Zhu [9], Bandyopadhyay *et al.* [10]	✗	✗	✓	✓	✗
Zhang *et al.* [11]	✓	✗	✗	✓	✗

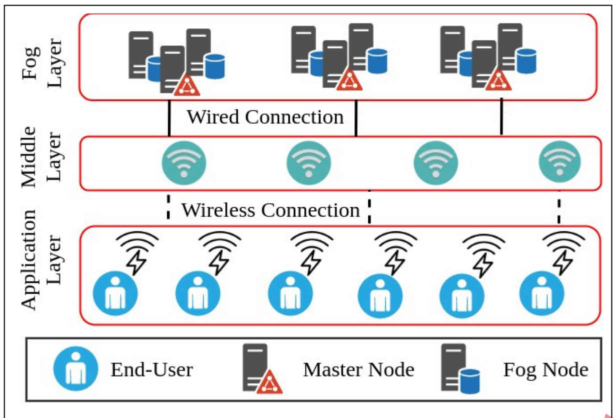


Fig. 1: Schematic Diagram of FogPrime Architecture

Master Node Selection and Cluster formation: We consider that the master nodes are self-elected fog nodes while satisfying the following constraints:

$$C_f \geq C_{th} \text{ and } E_f \geq E_{th} \quad (1)$$

where C_f and E_f represent the computational capacity and residual energy of fog node f, respectively. On the other hand, C_{th} and E_{th} denote the threshold values of the computational capacity and residual energy for getting selected as master node. To optimize the number of master nodes, i.e., clusters, the master nodes select $|\mathcal{C}|$ number of master nodes or clusters, using *p-dispersion* method [24]. We assumed that there are, at most, $|\mathcal{C}|$ service providers in the network. We considered that the dispersion index I_{m_i,m_j} of two master nodes m_i and m_j is defined as their Euclidean distance in a two-dimensional space having the difference in their computational capacities along the X-axis and that in their residual energies along the Y-axis. Mathematically,

$$I_{m_i,m_j} = \sqrt{\left(\frac{C_{m_i} - C_{m_j}}{C_{th}}\right)^2 + \left(\frac{E_{m_i} - E_{m_j}}{E_{th}}\right)^2} \quad (2)$$

The master nodes aim to maximize $fn(\mathcal{C})$ [24], where $fn(\mathcal{C}) = \min\{I_{m_i,m_j} : 0 < i < j < |\mathcal{F}|\}$. After selecting the set of master nodes, each fog node $f \in \mathcal{F}$ joins the cluster of the nearest master node based on the geographical distance among them.

Additionally, there exist multiple service providers in the fog network, where each service provider owns a cluster of fog nodes. On the other hand, the end-users do not own the fog nodes. However, they enjoy the service from the fog nodes based on the *pay-per-usage* model which is similar to the cloud-based services. The fog network is visualized to be a distributed system, whereas the cloud-based networks are centralized. Additionally, we argue that the fog nodes have high computational capacity than normal IoT devices. Hence, in the presence of multiple IoT nodes, the fog network ensures services with low latency.

We consider that at a particular time instant t, each end-user $n \in \mathcal{N}$, where $\mathcal{N}$ is the set of end-users, requests to serve a set of applications $\mathcal{A}_n(t)$ to the available clusters, i.e., the master nodes of the clusters. On receiving this request, the master node of each cluster selects an optimal fog node and informs the end-user. Thereafter, the end-user selects an optimal cluster and informs the corresponding master node to deploy the application. The master node of the selected cluster deploys the application by initializing a *container* along with a *pod* on the optimal fog node. On the other hand, we consider that each cluster $c \in \mathcal{C}$, where $\mathcal{C}$ denotes the set of clusters available in the fog networks, comprises of $\mathcal{F}_c$ set of fog nodes. Therefore, we argue that the set of fog nodes $\mathcal{F}$ can be represented by $\bigcup_c \mathcal{F}_c$. Here, we assume that the fog nodes are heterogeneous in terms of memory and CPU capacities. Each fog node $f \in \mathcal{F}$ has computational and memory capacities of C_f and M_f, respectively. In addition, we argue that in traditional fog networks, there is a trend to allocate the applications to the nearest fog nodes, which may affect the overall performance of the fog networks. Hence, unlike the existing literature, we aim to consider the price paid by the end-users while scheduling the applications or jobs to a subset of fog nodes. We assume that each end-user n is willing to pay an amount $p^{a_n}_{max}$ for serving an application $a_n \in \mathcal{A}_n(t)$. Hence, the following constraint needs to be ensured:

$$p^n_a(t) \leq p^{a_n}_{max} \quad (3)$$

where $p^n_a(t)$ denotes the price paid by the end-user n for the service of application a_n.

In fog networks, the master node of each cluster needs to ensure that the fog nodes are not oversubscribed. In order to do this, while deploying a pod, each master node needs to ensure the following constraints:

$$\sum_{a \in \mathcal{A}_n(t), \forall n} x_{a,f} m_a \leq M_f \text{ and } \sum_{a \in \mathcal{A}_n(t), \forall n} x_{a,f} c_a \leq C_f \quad (4)$$

where m_a and c_a represent the specified memory and CPU resources for each application a, respectively, and $x_{a,f}$ is a binary variable and denotes the association of an application a with an fog node f. We define $x_{a,f}$ as follows:

$$x_{a,f} = \begin{cases} 1, & \text{if application } a \text{ is allocated to fog node } f \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

On the other hand, we consider that each fog node follows a first-in-first-out (FIFO) model for executing applications, similar to Kubernetes [1]. However, for efficient management of the fog network, each master node follows a redistribution of the applications, that are in the process, after a fixed interval which is assumed to be predefined[2].

IV. FogPrime: The Proposed Price-based Strategic Resource Management Scheme

To model the interaction between the master node and the fog nodes in a cluster, we use a *dynamic coalition-formation cooperative game with transferable utility* [25] for strategic allocation of resource to the applications and *utility theory* for strategic choice of each end-user. Hence, we argue that the proposed price-based strategic resource management scheme, named *FogPrime*, follows a multi-stage game-theoretic model.

Justification for Using Dynamic Coalition and Utility Games: Dynamic coalition game is an important game-theoretic approach to study the social welfare of forming a group of players through their internal interaction. It is also adaptive to the change in the surroundings. For the fog nodes, each application is served for a finite time. Hence, we argue that there will be a change in the surroundings, aperiodically, in terms of active applications and their resource requirements. Therefore, dynamic coalition game is one of the suitable choices. Additionally, we consider an extension of the dynamic coalition game by introducing the transferable utility. It signifies that the centralized coordinator always ensures the overall benefit of the system. Here, in FogPrime, the master nodes are the centralized coordinators that ensure the improved network performance of the cluster, i.e., fog networks, holistically, in terms of satisfaction of the fog nodes, network delay, and utilization of resources. The aforementioned problem can be mapped to the *bin-packing problem* [24], which is NP-hard. Therefore, we modeled the local strategic resource allocation problem in fog networks using dynamic coalition game.

On the other hand, utility theory defines the satisfaction of a consumer against any product in economics. Similarly, in FogPrime, each end-user follows the utility theory to ensure his/her satisfaction with an application to get served in the fog networks. We argue that the satisfaction of each end-user depends proportionally on his/her expectations in terms of tolerable delay and paid price. Therefore, using the utility theory, each end-user tries to ensure a trade-off between the paid price and the tolerable delay, as a reduction in service delay results in an increase in the price to be paid.

[1] https://kubernetes.io/

[2] It is to be noted that in this work, we do not consider the presence of cloud infrastructure.

A. Game Formulation

In FogPrime, we consider that the incoming application follows an M/M/1 queuing model for the master node and fog nodes in each cluster. Therefore, the following properties are true: (1) each application is processed individually, i.e., memoryless, and application arrivals follow a poison distribution, and (2) application inter-arrival time follows an exponential distribution. Hence, while allocating resources to an application, we follow a first-come-first-serve model, as mentioned earlier. In FogPrime, we consider that each fog node signifies a coalition[3]. After receiving an application from an end-user, the master node tries to find the equilibrium coalition, i.e., an optimal fog node, to which the application will be associated. In order to allocate an application $a \in \mathcal{A}_n$ to fog node $f \in \mathcal{F}_c$, the master node of cluster c takes into consideration the different parameters such as associated transmission delay $d_{n,f}$, price paid by end-user $p^{a^n}_{max}$, capacity[4] and memory requirement of the application, i.e., c_a and m_a, respectively, total and available capacity of fog node f, i.e., C_f and C_f^{eff}, respectively, and total and available memory associated with fog node f, i.e., M_f and M_f^{eff}, respectively. Hence, depending on the availability of the fog nodes, end-user n receives a set of prices, i.e., $\{\cdots, p^c_a(t), \cdots\}$ from the available clusters, where $p^c_a(t)$ denotes the price charged by cluster c. Thereafter, using utility theory, each end-user selects an equilibrium cluster, while deciding a trade-off between the price paid and the associated delay. The following are the three components of FogPrime:

1) The set of applications $\mathcal{A}_n$ of each end-user n gets served by a single cluster of fog nodes based on the availability. However, the end-user always has an option to choose a single cluster among the available clusters.

2) Each cluster c has a single master node, which uses a dynamic coalition-formation cooperative game with a transferable utility game. The master node decides which incoming applications are to be deployed in which fog node within the cluster and the price to be charged while maximizing the payoff of its utility function $\mathcal{U}_{a,c}(\cdot)$.

3) Each end-user n uses utility theory to decide the equilibrium cluster in which the application a is to be deployed, while maximizing the payoff of its utility function $\mathcal{K}_{a,c}(\cdot)$.

1) Resource Allocation in Clusters: Initially, for each application a, each end-user n requests the master nodes, i.e., the available clusters, while mentioning the tolerable delay d^a_{th}, $p^{a^n}_{max}$, c_a, and m_a. We consider that the set of clusters which receives a request for serving the application a, is denoted as $\mathcal{C}_a(t) \subseteq \mathcal{C}$. Thereafter, the strategy of the master node of each cluster $c \in \mathcal{C}_a(t)$ is to select an optimal fog node f for serving the application a, while maximizing the utility function $\mathcal{U}_{a,c}(\cdot)$ of the cluster c, as discussed in the following sections.

[3] Here, we would like to mention that in this work, we use two terminologies — cluster and coalition, which are not similar. Each coalition is associated with a fog node, where it signifies the group of applications deployed on that fog node. On the other hand, the cluster signifies the set of fog nodes. Each cluster has a master node, which is also a fog node.

[4] We consider that the computation capacity is measured by the number of instructions executed per cycle.

a) Utility Function of the Fog Nodes: The utility function $\mathcal{B}_{a,f}(\cdot)$ of each fog node f signifies the revenue earned by providing services to the end-users. Each fog node f tries to maximize its resource utilization while maximizing the earned profit. The utility function $\mathcal{B}_{a,f}(\cdot)$ of each fog node f has the dependencies mentioned as follows. (1) The payoff of the utility function $\mathcal{B}_{a,f}(\cdot)$ decreases linearly with the increase in resource requirement. For example, let us consider that we have two applications a_1 and a_2 and the corresponding resource requirements are c_{a_1} and c_{a_2}, respectively. We argue that FogPrime satisfies the constraint, $\mathcal{B}_{a_1,f}(\cdot, c_{a_1}) < \mathcal{B}_{a_2,f}(\cdot, c_{a_2})$, while considering that $c_{a_1} > c_{a_2}$ and the other parameters are constant. (2) Similarly, with the increase in the memory requirement m_a of each application a, the payoff of the utility function $\mathcal{B}_{a,f}(\cdot)$ decreases. (3) The payoff of the utility function $\mathcal{B}_{a,f}(\cdot)$ varies linearly with the increase in the price p_a^f, where p_a^f signifies the price to be charged by the fog node f for provisioning services to application a. (4) We argue that the price p_a^f varies exponentially with the change in the transmission delay $d_{n,f}$, as discussed in Section IV-A1b. With the increase in the transmission delay $d_{n,f}$, the price p_a^f to be charged decreases.

Therefore, we formulate the utility function $\mathcal{B}_{a,f}(\cdot)$ of each fog node f as follows, while satisfying the aforementioned properties:

$$\mathcal{B}_{a,f}(\cdot) = \frac{p_a^f}{\phi_a \frac{x_{a,f}}{C_f} + (1-\phi_a)\frac{x_{a,f} m_a}{M_f}} - \frac{L_f}{|\mathcal{A}_f(t)|+1} \tag{6}$$

where ϕ_a is a constant for application a_n and denotes the type of the application, i.e., computation or memory-intensive, where $0 \leq \phi_a \leq 1$; L_f denotes the cost incurred by the fog node f for maintaining the resources with high efficiency; and $\mathcal{A}_f(t)$ denotes the set of applications that have already been allocated to fog node f. Here, we consider that L_f is constant for each fog node f. Hence, each fog node f evaluates its won payoff while satisfying the constraints mentioned in Equation (4).

b) Price Function of Each Fog Node: Motivated by Friis equation [26], we consider that the price received for provisioning service decreases with the increase in the overall delay $d_{n,f}$[5]. We consider that the price charged p_a^f is defined as follows:

$$p_a^f = \alpha + \beta \frac{p_{max}^{a_n}}{(d_{n,f})^2 + \epsilon} \tag{7}$$

where α, β and ϵ are constants; and $\alpha > 0$, $0 < \epsilon << 1$, and $0 < \beta < 1$. We consider an ideal wireless network scenario, and hence, do not consider the effects of noise and fading in FogPrime. Hence, for mathematical modeling, we consider that the transmission delay between two nodes n and f varies linearly with the Euclidean distance $\delta_{n,f}$ between those nodes, i.e., $\theta\delta_{n,f}$, where θ is a constant.

c) Utility Function of the Clusters: The utility function $\mathcal{U}_{a,c}(\cdot)$ of each cluster c signifies that the overall revenue of the cluster. The master node of each cluster tries to maximize the overall payoff of the cluster by choosing the strategic fog node, i.e., the equilibrium coalition to deploy the pod of the requested application. Therefore, according to the dynamic coalition-formation cooperative game, in FogPrime, each master node derives a preferential relation among the coalitions, i.e., the fog nodes. Here, we consider that the payoff of the utility function $\mathcal{U}_{a,c}(\cdot)$ is calculated by using the following equation, while satisfying the constraints in Equations (3) and (4).

$$\mathcal{U}_{a,c}(\cdot) = \max\{\mathcal{U}_{a,c}^f(\cdot)\} \tag{8}$$

where $\mathcal{U}_{a,c}^f(\cdot)$ signifies the overall payoff of the cluster while allocating the incoming application a to fog node f. Based on the payoff of $\mathcal{U}_{a,c}^f(\cdot)$, the master node defines the preferences. We define the utility function $\mathcal{U}_{a,c}^f(\cdot)$ as follows:

$$\mathcal{U}_{a,c}^f(\cdot) = \mathcal{B}_{a,f}(\cdot) \prod_{k\in\mathcal{F}_c, k\neq f} \{\mathcal{B}_{a',k}(\cdot) | a' \in [\mathcal{A}_k \setminus \{a\}]\} \tag{9}$$

To illustrate this, we consider the following example. We consider that two fog nodes f_1 and f_2 satisfy the constraints mentioned in Equation (4). However, we observe that $\mathcal{U}_{a,c}^{f_1}(\cdot) \geq \mathcal{U}_{a,c}^{f_2}(\cdot)$. Therefore, the application would be allocated to fog node f_1, i.e., $f_1 \triangleright f_2$, where $\triangleright$ represents the relation – *preference-over*. Additionally, we argue that FogPrime follows a transferable utility model. In other words, in FogPrime, the profit of provisioning service to an application is evenly distributed among the fog nodes within the cluster. In other words, the objective of each cluster is to maximize $\mathcal{U}_{a,c}(\cdot)$, as defined below:

$$\arg_f \max \mathcal{U}_{a,c}^f(\cdot) \tag{10}$$

while satisfying the constraints in Equations (3) and (4). In FogPrime, based on the preference relation among the fog nodes, the price to be charged for each cluster is decided. We consider that each cluster c defines the price $p_a^c(t)$ to be charged and the associated delay $d_{n,c}(t)$ for serving application a is calculated as:

$$p_a^c(t) = \{p_a^f | f \in \mathcal{F}_c \text{ and } f \triangleright \forall k \in \mathcal{F}_c/\{f\}\} \tag{11}$$

$$d_a^c(t) = \{d_{n,f} | f \in \mathcal{F}_c \text{ and } f \triangleright \forall k \in \mathcal{F}_c/\{f\}\} \tag{12}$$

2) Selection of Cluster for Acquiring Service: Based on the information provided by each cluster – price to be charged $p_a^c(t)$ and associated transmission delay $d_a^c(t)$, where $c \in \mathcal{C}_n$, each end-user n selects a subset of clusters using utility theory based on his/her requirements. In FogPrime, the strategy of each end-user is to maximize its utility function to ensure a trade-off between the aforementioned parameters. The detailed information on the utility function is discussed below.

a) Utility Function of the End-Users: The utility function $\mathcal{K}_{a,c}(\cdot)$ of each end-user n signifies his/her satisfaction by availing service from the cluster c in fog network. The payoff value of $\mathcal{K}_{a,c}(\cdot)$ may vary based on the preference of the end-users. Therefore, while designing the utility function $\mathcal{K}_{a,c}(\cdot)$, we ensure the following properties. (1) With the increase in

[5] We consider the transmission and processing delay.

the price, the utility function decreases. We consider that the utility function follows a linear relation with the price charged. (2) Each end-user wants to avail services with minimum delay. Hence, we consider that with the increase in the transmission delay, the payoff value decreases.

Therefore, we formulate the utility function $\mathcal{K}_{a,c}(\cdot)$, as follows:

$$\mathcal{K}_{a,c}(\cdot) = \frac{1}{\gamma_n + \zeta_n}\left[\gamma_n \frac{p^{a_n}_{max}}{p^c_a(t)} + \zeta_n \frac{d^a_{th}}{d^c_a(t)}\right] \tag{13}$$

where γ_n and ζ_n are constants for each end-user n, and signifies the preferences among the paid price $p^c_a(t)$ and associated delay $d^c_a(t)$. The objective of each end-user n is to maximize his/her utility function, which is defined as follows:

$$\arg_c \max \mathcal{K}_{a,c}(\cdot) \tag{14}$$

while satisfying the constraint $d^a_{th} \geq d^c_a(t)$. Moreover, the constraint mentioned in Equation (3) is to be satisfied.

Algorithm 1 RAC: Resource Allocation in Each Cluster

INPUTS:

1: $p^{a_n}_{max}$, $\{\delta_{n,f}, C_f, M_f, L_f, \mathcal{A}_f\}\forall_f$, $\theta, \alpha, \beta, \epsilon$, c_a, m_a, d^a_{th}

OUTPUT:

1: $p^c_a(t)$, $d^c_a(t)$

PROCEDURE:

1: $\mathcal{B} \leftarrow \{\emptyset\}$; $p \leftarrow \{\emptyset\}$; $d \leftarrow \{\emptyset\}$
2: **for** each $f \in \mathcal{F}_c$ **do**
3: $\quad < v_1, v_2, v_3 > \leftarrow$ CALL RAF $\qquad \triangleright$ Algorithm 2
4: $\quad p \leftarrow p \cup \{v_1\}$; $d \leftarrow v \cup \{v_2\}$; $\mathcal{B} \leftarrow \mathcal{B} \cup \{v_3\}$
5: **end for**
6: **for** each $f \in \mathcal{F}_c$ **do**
7: $\quad$ Calculate $\mathcal{U}^f_{a,c}(\cdot)$ using Eq. (9),
8: **end for**
9: $\mathcal{U}_{a,c}(\cdot) \leftarrow -\infty$; $p^c_a(t) \leftarrow \infty$; $d^c_a(t) \leftarrow \infty$
10: **for** each $f \in \mathcal{F}_c$ **do**
11: $\quad$ **if** $\mathcal{U}_{a,c}(\cdot) < \mathcal{U}^f_{a,c}(\cdot)$ **then**
12: $\qquad p^c_a(t) \leftarrow p^f_a$ and $d^c_a(t) \leftarrow d_{n,f}$
13: $\quad$ **end if**
14: **end for**
15: **Return** $< p^c_a(t), d^c_a(t) >$;

B. Equilibrium in FogPrime Scheme

To identify the strategic equilibrium [25] of FogPrime, the preference relation among the super set of the possible partitions $\mathcal{U}^f_{a,c}(\cdot)$ of set $\mathcal{F}_c$ of the fog nodes in each cluster c needs to be evaluated, as defined in Definition 1.

Definition 1. *Given a set $\mathcal{F}_c$ of fog nodes in a cluster c and the requirements of an incoming application a, the preference of two probable partitions P_f and $P_{f'}$ follows $P_f \preceq P_{f'}$, if $\mathcal{U}^f_{a,c}(\cdot) \leq \mathcal{U}^{f'}_{a,c}(\cdot)$, where $f' \neq f$ and $\{f', f\} \in \mathcal{F}_c$, is satisfied. Therefore, we get that, in cluster c, the master node can achieve local equilibrium by allocating the application a to fog node $f' \in \mathcal{F}_c$.*

On the other hand, based on Definition 1, each end-user n can achieve global equilibrium by satisfying the following inequality while allocating each application $a \in \mathcal{A}_n$.

$$\mathcal{K}_{a,c}(\cdot) \leq \mathcal{K}_{a,c'}(\cdot) \tag{15}$$

From Equation (15), we obtain that FogPrime ensures the presence of global equilibrium and cluster c' is the global equilibrium solution for allocating the application a.

Algorithm 2 RAF: Resource Allocation in Each Fog Node

INPUTS:

$p^{a_n}_{max}$, $\delta_{n,f}$, $\theta, \alpha, \beta, \epsilon$, c_a, m_a, d^a_{th}, C_f, M_f, L_f, $\mathcal{A}_f$

OUTPUT:

p^f_a, $d_{n,f}$, $\mathcal{B}_{a,f}(\cdot)$

PROCEDURE:

$d_{n,f} \leftarrow \theta\delta_{n,f}$
if $d_{n,f} \leq d^a_{th}$, $c_a \leq C_f$, and $m_a \leq M_f$ **then**
$\quad$ Calculate the optimal price p^f_a using Equation (7)
$\quad$ Calculate $\mathcal{B}_{a,f}(\cdot)$ using Equation (6)
$\quad$ **Return** $< p^f_a, d_{n,f}, \mathcal{B}_{a,f}(\cdot) >$
end if
Return $< \infty, \infty, -\infty >$

C. Algorithms

In FogPrime, the end-users, and the fog nodes interact with one another in real-time to make an agreement for service requirements and the price to be paid. Initially, each end-user connects with the subset of clusters within his/her access, distributively, and informs the requirements of the application – memory, computation resource, and the threshold delay – to be served. After receiving this information, the master node in each cluster performs Algorithm 1 to decide the suitable fog node. In the process, each master node calculates the coalition payoff for each fog node using Algorithm 2, if the application gets allocated to the particular fog node. Each master node finds out the strategic subset of fog nodes that are eligible to serve the requested application and informs the end-user accordingly. By receiving the reply messages from each cluster, using Algorithm 3, each end-user decides the strategic subset of fog nodes/clusters to be selected based on the trade-off between the service delay and the associated price. The flowchart of FogPrime is presented in Figure 2.

Complexity Analysis: We observe that, in Algorithm 1, the time complexity for line 1 is $O(1)$. Algorithm 2 is invoked in line 3 of Algorithm 1. We observe that the time complexity of Algorithm 2 is $O(1)$. Hence, for lines 2-5 of Algorithm 1, the time complexity is $O(|\mathcal{F}_c|)$. Similarly, we observe that lines 6-8 and 10-14 of Algorithm 1 have time complexity $O(|\mathcal{F}_c|)$. For line 9 of Algorithm 1, the time complexity is $O(1)$. Therefore, the overall time complexity for Algorithm 1 is $O(|\mathcal{F}_c|)$, whereas the complexity for Algorithm 2 is $O(1)$. On the other hand, the time complexity of lines 1-2 of Algorithm 3 is $O(1)$. Therefore, the overall time complexity of Algorithm 3 is $O(|\mathcal{C}_n|)$. Hence, we conclude that the time complexity of FogPrime is $O(|\mathcal{F}_c| + |\mathcal{C}_n|)$.

Algorithm 3 SCAS: Selection of Cluster for Acquiring Service

INPUTS:

$p_a^c(t), \forall c \in \mathcal{C}_n$, $d_a^c(t), \forall c \in \mathcal{C}_n$, $p_{max}^{a_n}$, d_{th}^a, γ_n, ζ_n

OUTPUT:

c^* ▷ Selected cluster

PROCEDURE:

$c^* \leftarrow$ NIL and $K_n \leftarrow -\infty$
for each $c \in \mathcal{C}_n$ **do**
  Calculate $\mathcal{K}_{a,c}(\cdot)$ using Equation (13)
  if $K_n < \mathcal{K}_{a,c}(\cdot)$ **then**
    $c^* \leftarrow c$
  end if
end for
Return c^*;

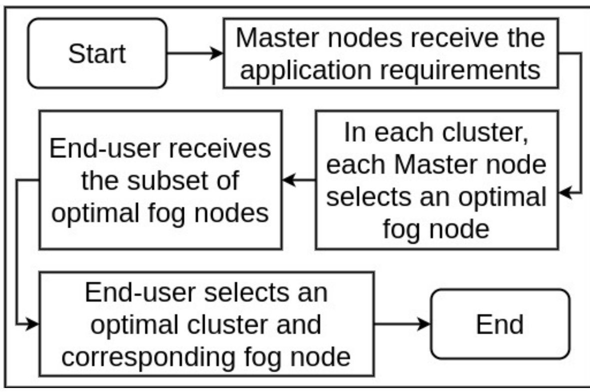


Fig. 2: Flowchart of FogPrime

Additionally, FogPrime consumes memory of the master node of each cluster and the end-users. In Algorithm 1, we observe that the memory consumption of each master node is $O(|\mathcal{F}_c|)$, whereas the memory consumption of each ordinary fog node is $O(1)$, as shown in Algorithm 2. On the other hand, the memory consumption of Algorithm 3 is $O(1)$. Therefore, we argue that the memory consumption of FogPrime is $O(|\mathcal{F}_c|)$.

V. Performance Evaluation

To evaluate the performance of FogPrime, we simulated it on a python-based platform. The detailed simulation platform and the observations are discussed in the following sections.

A. *Simulation Parameters*

We simulated the fog network with multiple end-users and multiple clusters with a finite number of fog nodes on a python-based platform. We considered that the end-users and the fog nodes are deployed randomly over an area of 100 $km\times$ 100 km. We varied the number of applications in the fog network and evaluated the performance of FogPrime. We consider that each end-user requests to serve a single application. Additionally, we varied the number of fog nodes and the number of clusters, as mentioned in Table II. We selected the application requirements – CPU and memory – randomly. Based on these requirements, the master nodes select the strategic subset of fog nodes.

TABLE II: Simulation Parameters

Parameter	Value
Number of applications	$500, 1,000, 2,000$
Number of fog nodes	100, 200, 300
Average number of incoming app.	50-100/unit-time
Average number of serving app.	40-80/unit-time
Maximum price	20-40 unit/app
CPU requirement	0.2-1.0 mCPU/app
Memory requirement	50-100 mb/app
β, γ_n, ζ_n, ϕ_a	0.5
α	1
d_{th}^a	30-50 ms
θ	$\frac{1}{299792458}$ s/m

B. *Benchmarks*

To evaluate the performance of FogPrime, we compare with two schemes in the existing literature – (a) Computing Resource Allocation scheme (IoTFog) [11], and (b) Trust Enforcing Pricing scheme (DETER) [6][6]. In IoTFog, Zhang *et al.* [11] proposed a resource allocation scheme for fog networks. In this work, the authors considered fog as a multi-tier network. Using Stackelberg and matching games, the authors formulated the resource allocation problem as a joint optimization problem. However, the authors did not consider the effects of the pricing model on fog networks. On the other hand, in DETER, Chakroborty *et al.* [6] proposed a pricing model based resource allocation scheme in sensor-cloud. In this work, the authors considered a hierarchical service-oriented architecture in the presence if end-users, sensor-owners, and sensor cloud service provider and decided the allocation of resources while focusing on the trustworthiness of the service entity, and accordingly evaluated pricing. However, this scheme is not suitable for fog networks. Hence, for the sake of uniformity, we remodeled the DETER pricing scheme for fog networks and compared it with FogPrime.

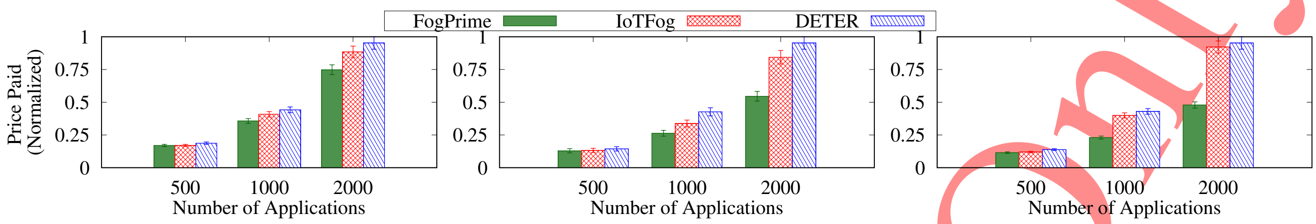
C. *Performance Metrics*

We compare the performance of FogPrime while considering the following performance metrics.

Price Paid by the End-Users: We consider that the price paid by the end-users is evaluated based on Equation (7) for the three schemes. However, we clarify that for the decision of resource allocation, price is one of the important factors in FogPrime. We note that the end-users try to pay less while ensuring that their service requirements are satisfied.

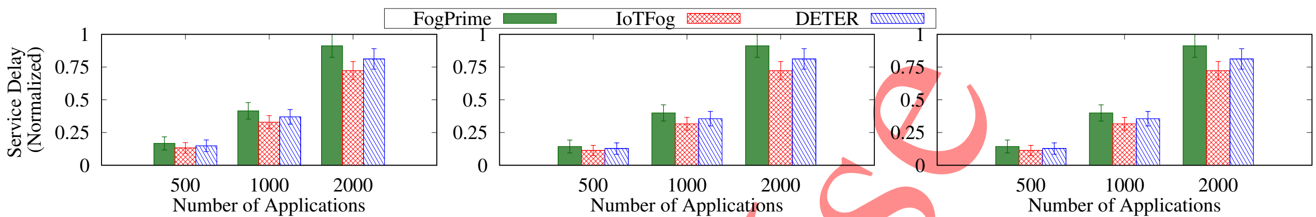
Delay in Service: We consider that the delay in service comprises of two factors – processing and transmission delays. The end-users aim to reduce the delay in service while paying less. Hence, the end-users need to make a trade-off between the service delay and the paid price.

[6]We considered two benchmark schemes – IoTFog and DETER – as representatives of IoT-enabled fog networks and IoT-enabled cloud networks, respectively. Hence, we compared the performance of FogPrime with these two existing benchmark schemes which are most suitable for the application scenarios considered in this work.



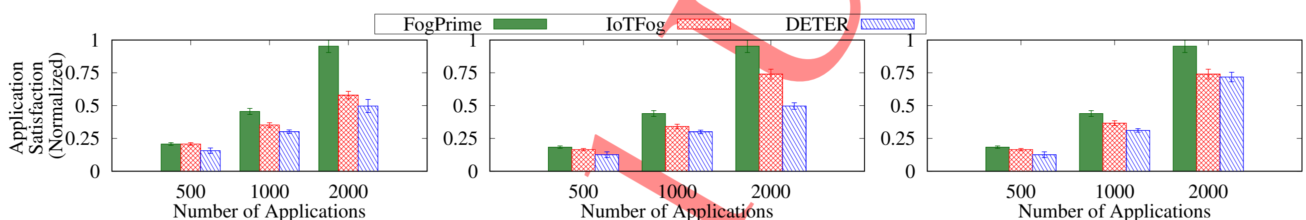
(c) Number of FogNodes = 100 (b) Number of FogNodes = 200 (c) Number of FogNodes = 300

Fig. 3: Average Price Paid by the End-Users



(c) Number of FogNodes = 100 (b) Number of FogNodes = 200 (c) Number of FogNodes = 300

Fig. 4: Average Service Delay per Application



(c) Number of FogNodes = 100 (b) Number of FogNodes = 200 (c) Number of FogNodes = 300

Fig. 5: Application Satisfaction

Satisfaction of the End-Users: We evaluate the satisfaction $sf_e(t)$ of the end-users based on the following equation:

$$sf_e(t) = \begin{cases} 1 - \frac{p_a^{c*}(t)}{p^a_{max}}, & \text{if } d_{n,f} \le d^a_{th},\ c_a \le C_f,\ \&\ m_a \le M_f \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

We argue that the satisfaction of the end-users becomes zero if their delay or resource requirements are not fulfilled.
Satisfaction of the Fog Networks: We evaluate the satisfaction $sf_{Fog}(t)$ of the fog networks, as the ratio of cumulative price earned for utilizing unit resource and the maximum price which can be earned per unit resource consumption. Mathematically,

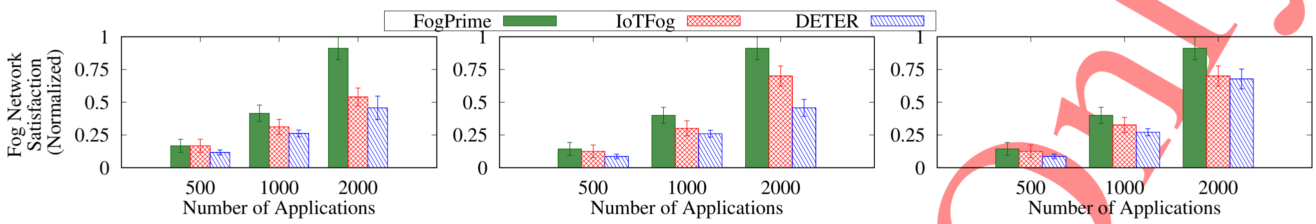
$$sf_{Fog}(t) = \sum_{f \in \mathcal{F}} \sum_{n \in \mathcal{N}} \sum_{a \in \mathcal{A}_n(t)} x_{a,f} \frac{p_a^f}{p^a_{max}} \tag{17}$$

We argue that if the resource utilization of a fog node increases and the price charged by the fog nodes reduces significantly, the change in the contribution of that fog node to the satisfaction of the fog networks tends to zero.

D. Results and Discussions

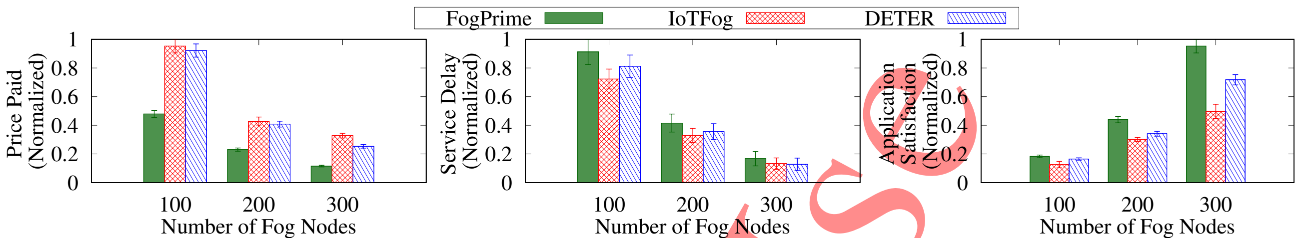
For simulation, we observe that for the considered deployment, the number of formed clusters is 5 and simulated for 50 unit time. We observe that using FogPrime, the applications having less delay threshold end up paying high. Using FogPrime, the end-users can choose from a local equilibrium subset of fog nodes, while ensuring a trade-off between the price and delay. However, using IoTFog, the closest fog node which satisfies the delay and resource constraint of the end-users get selected. Therefore, IoTFog does not explore the eligible set of fog nodes for serving the end-users' applications. Therefore, the end-users end up paying high while having less delay. On the other hand, using DETER, the end-users eventually generate an affinity towards a subset of fog nodes. We observe that these affinity fog nodes are close to the end-users; hence, the price paid by the end-users are highly similar to the IoTFog. Therefore, we observe that the price paid by the end-users is almost similar, i.e., the minimum price, using FogPrime and IoTFog in the presence of less number of applications. However, with the increase in the number of applications, it decreases significantly using FogPrime than using IoTFog. Moreover, Figure 3 depicts that, using FogPrime, the amount paid by the end-users for provisioned applications reduces at most by 15.88–47.27% than using IoTFog and DETER while ensuring that the service requirements such as delay and resources, i.e., computational capacity and memory, are satisfied.

On the other hand, from Figure 4, we observe that the incurred service delay increases using FogPrime than using IoTFog and DETER. This is since, using IoTFog, the closet



(c) Number of FogNodes = 100 (b) Number of FogNodes = 200 (c) Number of FogNodes = 300

Fig. 6: Fog Network Satisfaction



(a) Price Paid (b) Service Delay (c) Application Satisfaction

Fig. 7: Comparison of Applications' Parameters (Number of Applications = 1000)

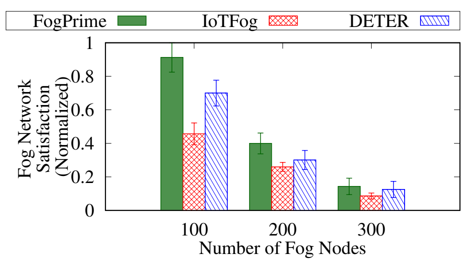


Fig. 8: Fog Network Satisfaction (Number of Apps. = 1000)

fog nodes which satisfy the resource requirements, i.e., computation resource and memory, are selected. Similarly, using DETER, the fog nodes closer to the end-users have high trust values. Therefore, the service delay using IoTFog and DETER is lesser than that using FogPrime. However, FogPrime ensures a trade-off between the service delay and the price charged. Hence, we yield that FogPrime ensures price-based strategic resource allocation in fog networks.

Figure 5 depicts the application satisfaction of the end-users. We observe that using FogPrime, there is an increase of 24.49-47.82% in the application satisfaction of the end-users than using IoTFog and DETER. FogPrime allocates resources, strategically, while ensuring the service requirements are satisfied and the end-users pay less. On the other hand, the other existing approaches focus on allocating resources in a greedy manner, which results in decreased user satisfaction. We observe that, using FogPrime, the delay in service increases in comparison with the existing schemes IoTFog and DETER, as mentioned earlier. From Figure 5, we get that, though the average delay in service increases using FogPrime, the end-users pay a lesser price than using IoTFog and DETER, which confirms the result obtained from Figure 3. Additionally, Figure 6 depicts that the overall satisfaction of the fog networks increases by 30.35-49.92% using FogPrime than using IoTFog and DETER. The reason behind this observation is that using IoTFog and DETER, the closest fog nodes of the end-users are oversubscribed. Consequently, the processing delay increases significantly, whereas most of the fog nodes in the fog networks remain idle. However, using FogPrime, we observe that each fog node gets an equal opportunity to get selected for serving an application. Additionally, we observe that the satisfaction of each fog node is random, as the allocation of a fog node to serve an application is decided by the local eligible clusters in the fog networks. Therefore, we argue that using FogPrime, the average fog node utilization increases, which, in turn, increases the satisfaction of the fog networks.

On the other hand, from Figures 7(a) and 7(b), we observe that, with the increase in the number of fog nodes in the network, the price paid by the end-users and service delay per application decreases. This is since, with the increase in the number of fog nodes, the availability of fog nodes increases. However, we observe that using FogPrime, the price paid per application is less than using IoTFog and DETER, as the existing schemes use a greedy approach and select the nearest possible node, as mentioned earlier. On the other hand, due to reduced price per application, the application satisfaction of the end-users increases with the increase in the number of fog nodes, as shown in Figure 7(c). Moreover, from Figure 8, we observe that, with the increase in the number of fog nodes, the satisfaction of the fog network decreases, which supports the facts observed earlier. Although there is a reduction in network satisfaction with the reduction in price per application, we observe that using FogPrime, network satisfaction is higher than using IoTFog and DETER. This is because of using FogPrime, a high number of applications are served than using IoTFog and DETER.

Hence, we conclude that FogPrime outperforms the existing schemes – IoTFog and DETER – while minimizing the charged price and the associated delay.

VI. Conclusion

In this work, we proposed a dynamic pricing-based resource allocation scheme, named FogPrime, for fog networks. We considered that the fog networks comprised of multiple clusters in the presence of multiple end-users. In each cluster, we have a single master node and multiple fog nodes. In FogPrime, initially, the end-users send the service request to the clusters. Using dynamic coalition-formation game with transferable utility, the master nodes decide a local equilibrium subset of fog nodes eligible for service the requested application. Thereafter, each end-user using utility game decides the global equilibrium subset of fog nodes/clusters, while minimizing the service delay and associated price. Through simulation, we observed that FogPrime outperforms the existing schemes in terms of resource utilization and the price paid by the end-users.

In the future, this work can be extended while considering the presence of cloud architecture in the presence of fog networks. Through simulations, we observed that a subset of applications is not served due to the limitation of resources. Hence, this work can be extended while exploring the migration of services from fog networks to the cloud to ensure that all the services are getting served. This work also can be extended while consolidating the service requests from the end-users, and provisioning a single fog node for the subset of applications.

Acknowledgement

This work was supported by the Indo-French Center for the Promotion of Advanced Research (IFCPAR) and Centre Franco-Indien Pour La Promotion De La Recherche Avancée (CEFIPRA) (Grant No: IFC/A/DST-INRIA-5^{th} call/2017/18/02/476).

References

[1] D. R. D. Vasconcelos, R. M. D. C. Andrade, and J. N. D. Souza, "Smart Shadow – An Autonomous Availability Computation Resource Allocation Platform for Internet of Things in the Fog Computing Environment," in *Proc. of Int. Conf. on Dist. Comp. in Sen. Syst.*, Jun 2015, pp. 216–217.

[2] M. Iorga, L. Feldman, R. Barton, M. J. Martin, N. Goren, and C. Mahmoudi, "Fog Computing Conceptual Model," National Institute of Standards and Technology, Tech. Rep., Mar 2018.

[3] J. Akram, Z. Najam, and A. Rafi, "Efficient Resource Utilization in Cloud-Fog Environment Integrated with Smart Grids," in *Proc. of Int. Conf. on Frontiers of Inf. Tech.*, Dec 2018, pp. 188–193.

[4] S. Yi, C. Li, and Q. Li, "A Survey of Fog Computing," in *Proc. of Wrkshp on Mobile Big Data*, 2015, pp. 1–6.

[5] R. Ben Halima, S. Kallel, W. Gaaloul, Z. Maamar, and M. Jmaiel, "Toward a correct and optimal time-aware cloudresource allocation to business processes," *Future Generation Computer Systems*, vol. 112, pp. 751 – 766, 2020.

[6] A. Chakraborty, A. Mondal, A. Roy, and S. Misra, "Dynamic Trust Enforcing Pricing Scheme for Sensors-as-a-Service in Sensor-Cloud Infrastructure," *IEEE Trans. on Serv. Comp.*, pp. 1–12, 2018.

[7] S. Misra, R. Schober, and A. Chakraborty, "RACE: QoI-Aware Strategic Resource Allocation for Provisioning Se-aaS," *IEEE Transactions on Services Computing (Early Access)*, pp. 1–1, 2020, DOI: 10.1109/TSC.2020.3001078.

[8] M. Aazam, E. Huh, M. St-Hilaire, C. Lung, and I. Lambadaris, "Cloud Customer's Historical Record Based Resource Pricing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 7, pp. 1929–1940, 2016.

[9] M. J. Farooq and Q. Zhu, "QoE Based Revenue Maximizing Dynamic Resource Allocation and Pricing for Fog-Enabled Mission-Critical IoT Applications," *IEEE Transactions on Mobile Computing*, 2020, dOI: 10.1109/TMC.2020.2999895.

[10] A. Bandyopadhyay, T. S. Roy, V. Sarkar, and S. Mallik, "Combinatorial Auction-Based Fog Service Allocation Mechanism for IoT Applications," in *Proc. of International Conference on Cloud Computing, Data Science Engineering*, 2020, pp. 518–524.

[11] 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 Int. of Things J.*, vol. 4, no. 5, pp. 1204–1215, Oct 2017.

[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. on Comm.*, vol. 67, no. 1, pp. 489–502, Jan 2019.

[13] D. T. Nguyen, L. B. Le, and V. K. Bhargava, "A Market-Based Framework for Multi-Resource Allocation in Fog Computing," *IEEE/ACM Trans. on Net.*, vol. 27, no. 3, pp. 1151–1164, Jun 2019.

[14] H. A. M. Name, F. O. Oladipo, and E. Ariwa, "User Mobility and Resource Scheduling and Management in Fog Computing to Support IoT Devices," in *Proc. of the 7^{th} Int. Conf. on Innov. Comp. Tech.*, Aug 2017, pp. 191–196.

[15] Z. Zhao, S. Bu, T. Zhao, Z. Yin, M. Peng, Z. Ding, and T. Q. S. Quek, "On the Design of Computation Offloading in Fog Radio Access Networks," *IEEE Trans. on Veh. Tech.*, vol. 68, no. 7, pp. 7136–7149, Jul 2019.

[16] S. Shaik and S. Baskiyar, "Fog Infrastructure and Service Management," in *Proc. of the 9^{th} Int. Green and Sust. Comp. Conf.*, Oct 2018, pp. 1–2.

[17] S. Javaid, N. Javaid, S. K. Tayyaba, N. A. Sattar, B. Ruqia, and M. Zahid, "Resource Allocation using Fog-2-Cloud based Environment for Smart Buildings," in *Proc. of the 14^{th} Int. Wireless Comm. Mobile Comp. Conf.*, Jun 2018, pp. 1173–1177.

[18] H. Xiang, S. Yan, and M. Peng, "A Realization of Fog-RAN Slicing via Deep Reinforcement Learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2515–2527, 2020.

[19] S. Yan, L. Qi, Y. Zhou, M. Peng, and G. M. S. Rahman, "Joint User Access Mode Selection and Content Popularity Prediction in Non-Orthogonal Multiple Access-Based F-RANs," *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 654–666, 2020.

[20] M. G. Al Zamil, S. Samarah, M. Rawashdeh, M. S. Hossain, M. F. Alhamid, M. Guizani, and A. Alnusair, "False-Alarm Detection in the Fog-Based Internet of Connected Vehicles," *IEEE Trans. on Veh. Tech.*, vol. 68, no. 7, pp. 7035–7044, Jul 2019.

[21] Z. Zhou, P. Liu, J. Feng, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Computation Resource Allocation and Task Assignment Optimization in Vehicular Fog Computing: A Contract-Matching Approach," *IEEE Trans. on Veh. Tech.*, vol. 68, no. 4, pp. 3113–3125, Apr 2019.

[22] J. Du, L. Zhao, J. Feng, and X. Chu, "Computation Offloading and Resource Allocation in Mixed Fog/Cloud Computing Systems With Min-Max Fairness Guarantee," *IEEE Trans. on Comm.*, vol. 66, no. 4, pp. 1594–1608, Apr 2018.

[23] J. Du, L. Zhao, X. Chu, F. R. Yu, J. Feng, and C. I, "Enabling Low-Latency Applications in LTE-A Based Mixed Fog/Cloud Computing Systems," *IEEE Trans. on Veh. Tech.*, vol. 68, no. 2, pp. 1757–1771, Feb 2019.

[24] S. Misra and A. Chakraborty, "QoS-Aware Dispersed Dynamic Mapping of Virtual Sensors in Sensor-Cloud," *IEEE Trans. on Serv. Comp.*, pp. 1–12, 2019, DOI: 10.1109/TSC.2019.2917447.

[25] A. Chakraborty, A. Mondal, and S. Misra, "Cache-Enabled Sensor-Cloud: The Economic Facet," in *Proc. of IEEE WCNC*, Apr 2018, pp. 1–6.

[26] S. Misra, G. Mali, and A. Mondal, "Distributed Topology Management for Wireless Multimedia Sensor Networks: Exploiting Connectivity and Cooperation," *Int. J. of Comm. Syst.*, vol. 28, no. 7, pp. 1367–1386, 2015.