International Journal of Electrical and Computer Engineering (IJECE)
Vol. 13, No. 6, December 2023, pp. 6525~6537
ISSN: 2088-8708, DOI: 10.11591/ijece.v13i6.pp6525-6537

An efficient reconfigurable workload balancing scheme for fog computing network using internet of things devices

Nandini Gowda Puttaswamy[1], Anitha Narasimha Murthy[2]
[1]Computer Science and Engineering, East Point College of Engineering and Technology, Visvesvaraya Technological University, Belagavi, India
[2]Computer Science and Engineering, BNM Institute of Technology, Visvesvaraya Technological University, Belagavi, India

Article Info

Article history:

Received Jun 5, 2023
Revised Jul 11, 2023
Accepted Jul 18, 2023

Keywords:

Base station
Fog node computing network
Fuzzy golden eagle load balancing
Internet of things
Reconfigurable workload balancing algorithm
Work load allocation

ABSTRACT

Nowadays a huge amount of data has been communicated using fog nodes spread throughout smarty cities. the communication process is performed using fog nodes which are co-located with cellular base stations (BSs) that can move the computing resources close to internet of things (IoT) devices. In smart cities, a different type of data flow has been communicated through IoT devices. The communication process performs efficiently using the remote cloud. The IoT devices very close to the BS can communicate data without using fog nodes. Due to these phenomena, workload unbalancing occurs in IoT devices communicating in fog computing networks. Hence, it generates communication and computing latency. The task distribution process between the IoT devices is unbalanced. Hence, congestion and loss of information occur in fog computing network. A proposed reconfigurable load balancing algorithm (RLBA) is efficiently balancing the workload by reconfigurable communication channels and deviates the task with respect to the BS locations, IoT devices density and load IoT devices in each fog nodes in a network to minimize the communication and computing latency. As per the performance analysis, the proposed algorithm shows better performance as compared to conventional methods' average latency ratio, communication latency ratio, computing load and traffic load.

Corresponding Author:

Nandini Gowda Puttaswamy
Computer Science and Engineering, East Point College of Engineering and Technology, Visvesvaraya Technological University
Belagavi-590018, India
Email: nandini.educator@gmail.com

1. INTRODUCTION

The development of smart gadgets that can perceive the environment physical information has received a lot of research attention in recent years [1]. The term "internet of things (IoT)" refers to a concept in which linked the smart devices of one another over the internet and equipped with data analytics. IoT is currently employed in many different applications, including smart cities, smart homes, smart transportation, and smart health [2]. As the number and types of devices connected to the IoT increases, managing these devices efficiently becomes a vital consideration. As the number and variety of devices increase, so does the need for cloud-based thinking [3], [4]. But since the data streams produced by IoT devices are sent to a distant cloud over the internet, it uses an amount of energy and core network bandwidth [5].

Many delay-sensitive IoT applications typically have a significant data streams processing with a delay in cloud hosted architectures where the main network is frequently located long distance from the IoT devices [6]. Fog nodes, which bring computer resources closer to devices of IoT and IoT users, are one option for easing the traffic burden on the core network and lowering IoT devices latency. Sometimes IoT application require a huge number of linked devices that produce enormous volumes of data. A paradigm called fog computing seeks to deal with the difficulties of processing this data fast and effectively. Fog nodes, often referred to as fog servers or fog devices, are essential to fog computing because they bring storage and processing resources closer to the network's edge [7], [8].

IoT devices sense data flows, which are transmitted to the appropriate base stations (BSs) in a fog network processed by fog nodes that are physically close to the BSs. Therefore, each data flow's latency is the sum of the communications times between an IoT device and a BS and a fog node. In the IoT, there are two different types of latency in communications: device latency and channel latency. The delay brought on by the wireless channel is referred to as channel latency. In a network, for example, when a wireless transmission involves multiple BSs, the wireless signal has to be transmitted from one BS to another sequentially. As a result, there will be significant communication delay due to the time wasted by round-trip communications. This kind of delays is known as channel latency. On the other hand, an IoT device will experience a delay if it sends its data through fog nodes, which are dispersed in areas with less traffic. This is because data transit through wired or wireless lines takes time. Device latency is the term for this delay. Therefore, it is required for the data flows total latency in a network that computational tasks are dynamically distributed among fog nodes. The fog node workload is closely correlated with the number of IoT devices connected to its corresponding BS because each fog node in this article is coupled to a single BS. That is, data flows from any IoT device that joins an IoT network are offloaded to a single co-located fog node [9].

Research has shown that nearby wireless service blocks BSs overlap and give coverage is as shown in Figure 1, which depicts the basic architecture computing of fog network using IoT devices. To balance loads among these BSs, fog nodes placed within these overlapping BSs might be connected to appropriate BSs. This connection has a significant influence on both the computation demands of traffic and fog nodes of BSs. Load balancing should take into account both the traffic loads of BSs and the computation loads of fog nodes in order to lower the latency of all data flows among devices and servers in the network. This is because latency is made up of two components: communication delay and calculation latency. In essence, when overloaded some BSs, they turn into bottlenecks and slow down communications for connected devices. To lessen their traffic demands, some IoT devices connected to these overcrowded BSs should be offloaded to nearby BSs.

Our aim is to link devices of IoT to various fog and BS fog network nodes to lower the latency of all network data flows in the IoT devices. The average-latency ratios at BSs/fog nodes are used to represent, respectively, the processing latency in fog network nodes and the communications delay in BSs. We introduce the models for the compute demands at fog network nodes and mean traffic loads at BSs/fog nodes. Additionally, we have examined the effect load balancing has on the usual latency of IoT device traffic [10], [11]. As a solution to the problem of load-balancing in a fog IoT device network, we create a distributed IoT device association algorithm to map IoT based network devices to pertinent BSs/fog nodes, thereby reducing the mean latency of all network IoT data flows. The LAB accomplishes this by employing a greedy heuristic to repeatedly estimate traffic and resource burdens per BS before broadcasting this information back to devices of IoT so they may select practical and appropriate destinations. Additionally, a simulation is offered to demonstrate the convergence.

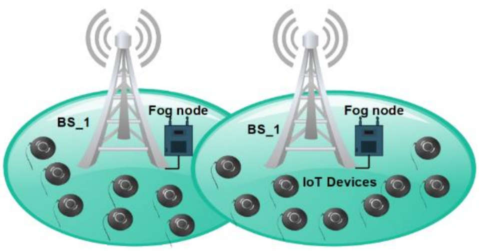

Figure 1. The fundamental structure of fog computing network using IoT device

2. RELATED WORKS

Since fog computing is close to users and devices off IoT, in some studies have concentrated on merging IoT with fog network computing. Fog computing has applications that were developed by Yang *et al*. [12]. In the environment of networking, Marbukh [13] outlined the advantages and disadvantages of fog computing and stated that it can cover the gaps left by IoT. An architectural strategy (EdgeIoT) was suggested by Donno and Dragoni [14] to control the IoT network device data streams at the fog network node. To further optimize traffic data flows and IoT network resources, Ju *et al.* [15] presented adaptive fog computing systems for IoT networking. It has been suggested that fog computing will enhance the functionality of mobile and IoT apps. In this work, we investigate the association strategy, the FogNode allocation technique, and the placement of virtual machines based on traffic load balancing and quality of service (QoS) in the medical cyber-physical system. Task scheduling and picture placement in a software-defined embedded system have been presented by Tadesse *et al*. [16] in fog computing as a way to reduce task request response times. A workload placement approach was developed by Kuzman *et al.* [17] in order to optimize the response times of hierarchical edge cloud network all tasks. The algorithm distributes tasks among various tiers of fog network nodes and allots computational resources from each fog network node for the tasks that each fog network node is given. In order to improve user task response times and lower cloudlet brown energy consumption, Elavarasan and Vincent [18] introduced a workload allocation technique, known as workload allocation (WALL), in a hierarchical cloudlet network. Tomar *et al.* [19] suggested transferring mobile users' virtual-machines (VM) among scattered cloudlets while also taking into account the energy requirements for VM migrations in order to decrease the overall consumption of energy dispersed cloudlets. The workload allocation method described in this study allocates cloudlet users to edge computing resources at various sites and then schedules tasks for the edge computing resources in accordance with the real-time needs of the associated users. Some studies also look at how wireline delay affects user task latency when edge computing resources are set up at various locations. Keep in mind that all of the aforementioned works only take wired communications latency into account, leaving out wireless delay. To reduce user response times for mobile edge computing networks, Hu *et al.* [20] present a model for load balancing cloudlets. In the present research, it is assumed that each user's wireless delay is constant. Some studies suggested adjusting BS transmission power to modify data speeds and shorten user wireless delays. Additionally, a number of research on cellular mobile networks' energy efficiency have been done.

3. SYSTEM MODEL

A fog IoT based network topology with fog nodes connected to BSs and overlapped coverage zones between close-by BSs is shown in Figure 2. It should be mentioned that all BSs use the next generation IoT (NBIoT) interface to deliver to IoT network devices communications services [21]. Because of the distribution of a workload across fog network nodes, the data flows require to pass over the mobile cellular, the IoT information flows are often to be processed at the local BS fog network node is preferred, which adds additional latency. On the other hand, to collect data a central controller is required on both the workload devices of IoT in the network topology and fog nodes in order to implement a centralized algorithm for the real-time allocation of workload among fog network nodes. For big size networks, such metropolitan area networks, this complexity makes it unusable. Therefore, we consider that the fog network node connected to the IoT network device's BS processes network data flows from IoT devices rather than other fog nodes. Other recent studies, like [22], similarly make the same assumption based on similar concerns. Because from network no need to collect data from every device, it should be noted that in this situation the compute loads can be distributed easily and with flexibility over several fog nodes.

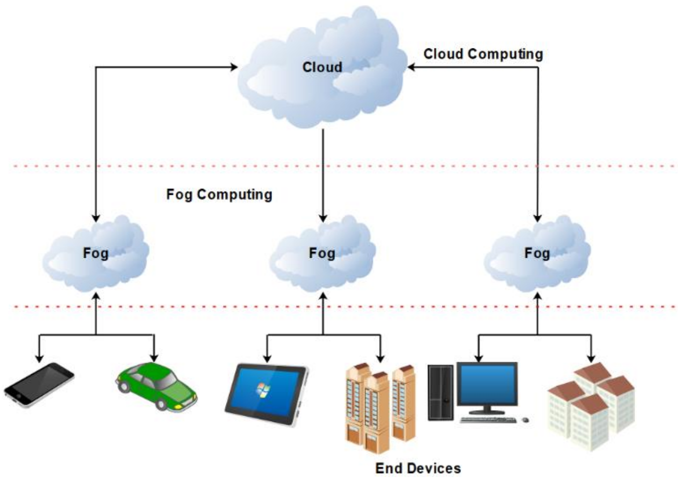

Figure 2. Fog network architecture

4. TRAFFIC LOAD MODEL

In this study, we considered that traffic loads arrive at a location x averaging a unit area rate, $\gamma(x)$ as by using a poisson-point process. Additionally, we suppose that each BS has a unique fog node, *F*. Indicate *x* as a place inside C and A as the total area covered by all BSs. We assume that IoT network data flows arrive at location x at a poisson-point process average rate of $\gamma(x)$ per unit area. Traffic loads vary spatially as a result of inhomogeneity. The main notations used in this work is shown in Table 1. The uplink channel gain from an IoT device's position to BS j, the noise power θ^2, and the IoT device transmission power at that location are all specified as $u_j(x)$. The IoT device-based network signal-to-noise ratio (SNR) may be computed using (1) and (2) illustrates in the theorem of Shannon Hartley, if we denote $b_j(x)$ as the IoT transmission power at location x and it is logarithmic function is denoted by $\lambda_j(x)$.

$$\beth_j(x) = \frac{T(x)u_j(x)}{\theta^2} \tag{1}$$

$$b_j(x) = \varphi_j \log(1 + \lambda_j(x)) \tag{2}$$

where $b_j(x)$ is represented IoT devices load traffic of average and φ_j is described the *j*th BS total bandwidth [23]. In various network environments, fog network nodes can be deployed by grouping smart devices together and creating a group of networks known as fog networks. The fog topology greatly improves the coverage of wireless network access points that are fixed to particular locations. In this arrangement, data flows received up by an IoT device are sent to its BS, where the fog network nodes is processed residing next to the BS. We will therefore concentrate on IoT device uplink connections and data processing of fog node in view to determine the flow of data latency. Equation (3) has been used to calculate the network load density average traffic for IoT devices.

$$\propto_j (x) = \frac{\gamma(x)m(x)\rho_j(x)}{b_j(x)} \tag{3}$$

where $\rho_j(x)$ is represented as binary variable with the location of 'x' is associated fog network nodes with the *j*th BS and $\aleph_j$ is described as avaearge traffic at BS, $m(x)$ is identified as exponential data flow distribution average value and the IoT device's typical traffic-load density at location *x* in BS *j*. The fog node average traffic has been calculated using (4).

$$\aleph_j = \sum_{x \epsilon C} \propto_J (x) \tag{4}$$

A several factors, including user fairness and network capacity are considered to manage the IoT network devices access radio resources in mobile communications. In this study, IoT network devices are arranged in round-robin fashion for uplink transmissions such that uplink channel access by several devices concurrently for the sake of analytical tractability. While the network data rates at each place are provided, the IoT based network traffic arrival information rate at location 'x' also complies with the poisson-process, the information traffic volumes of data flows follow the exponential distribution, and the service duration of IoT network data flows at location 'x' also exponential distribution satisfied [24] and the mean service time (MST) of network data flows at point *x* is determined by (5).

$$\emptyset_j(x) = \frac{m(x)}{b_j(x)} \tag{5}$$

As a simulation result, an M/M/1-processor serves sharing queue as the foundation for a BSs uplink communications (BS) [25]–[29]. As per the suggested approach, different IoT network devices will equitably information share the average ratio energy and IoT network devices information resources of a BS even when they have different IoT data rates depending on their channel conditions. Furthermore, we must constantly ensure that $\aleph_j$ is less than one in order to maintain the queue's stability. Processor sharing given the M/M/1-network data queue of a BS, mean value of delivery time at location x is given by (6).

$$\beta_j(x) = \frac{m(x)}{b_j(x)(1-\eta_j)} \tag{6}$$

The mean waiting time for each data flow at point x is given by (7).

$$\varphi_j = \beta_j(x) - \emptyset_j(x) = \frac{\eta_j m(x)}{b_j(x)(1-\eta_j)} \tag{7}$$

The symbol $\vartheta_j(x)$ represents the ratio of latency time of waiting to service in BS *j* for IoT network data flows at the location *x*. Then, since all of the devices in IoT connected to BS *j* have the same delay ratio and $\vartheta_j(x)$ only depends on the network traffic load of BS *j*, we define BS *j*'s latency ratio communications as we can deduct from (8) that raising the network traffic load *ρj* of BS *j* will increase *j*. IoT devices connected to BS *μj* must wait longer to reach the channel of transmission when *μj*>0, which is consistent with their delay of average delivery as indicated by (8).

$$\vartheta_j = \frac{\eta_j}{1-\eta_j} \tag{8}$$

5. PROBLEM FORMULATION

In this study, we balance workloads between fog and BSs nodes in an effort to reduce the data flows latency. We write $\tau(\alpha) = \sum_{j\in F} \vartheta_j + \widehat{\vartheta_j}$ as the fog network latency ratio. The goal of the suggested technique is to determine the ideal load distribution among BSs in view to reduce the fog network's latency ratio. Consequently, the issue might be stated as (9) to (13).

$$T_1 : \min_{\alpha} \complement(\alpha) \tag{9}$$

$$\sum_{j\in F}^{\alpha} \vartheta_j(x) = 1, \forall_x \in \tau \tag{10}$$

$$0 \le \eta_j \le \eta_{max}, \forall_j \in F \tag{11}$$

$$0 \le \widehat{\eta_j} \le \hat{\eta} max, \forall_j \in F \tag{12}$$

$$\eta_j(x) \epsilon \{0,1\}, \forall_x \in \tau \tag{13}$$

There can only be one BS associated with each location, according to constraints (10) and (13). The computation amount at fog node *i* is constrained by constraint (12). In load balancing, two factors influence the computation latency of data flows: traffic loads and computing loads. In an IoT network, when bandwidth-constrained devices are deployed in some areas, data congestion on overloaded BSs may result in a high latency for data flows [30]–[33]. On the other side, in a fog network, some fog nodes may become overloaded due to the heavy computational loads it has to host.

6. LAB: A DISTRIBUTED IoT DEVICE ASSOCIATION SCHEME

In this work, we suggest a load-shaping mechanism for wireless IoT networks that are delay-tolerant. The controller in our suggested system continuously calculates the compute and fog nodes (BSs) traffic loads, then broadcasts the results to devices of IoT [34]. The conclusion is that IoT devices can choose BSs iteratively based on information of real load and their data rates of uplink to various BSs. We demonstrate that this distributed system can manage problems like service disruptions while achieving the best possible trade-off between delays and power consumption.

6.1. Algorithm for IoT device side

Initially the kth iteration has been considered for all traffic loads η_j and manuplated loads $\hat{\eta}_j$ to devices of IoT. Based on fog node network $M(\alpha)$ as shown in (14).

$$\frac{\partial M(\alpha)}{\partial \alpha_j(x)} = \gamma(x) \frac{e_j(x)m(x)(1-\widehat{\eta}_j(k))^2 + b_j(x)v(x)(1-\eta_j(k))^2}{e_j b_j(x)(1-\widehat{\eta}_j(k))^2(1-\eta_j(k))^2} \tag{14}$$

IoT devices have been selected based on broadcast messages. Determine the BS is by using (15),

$$t^k(x) = arg \max_{j\in F} e_j(x) b_j(x) \psi_j(k) \tag{15}$$

where $e_j(x)$ is represented capacity computing (in CPU cycle/sec) for fog node 'J' and $\psi_j(k)$ is described the in 16 and $t^k(x)$ is identified as BS index and it controlled by condition is given in (17).

$$\psi_j(k) = \frac{(1-\hat{\eta}_j(k))^2(1-\eta_j(k))^2}{e_j(x)m(x)(1-\hat{\eta}_j(k))^2+b_j(x)v(x)(1-\eta_j(k))^2} \tag{16}$$

$$\alpha_j^k(x) = \begin{Bmatrix} 1, & if\ j = t^k(x), \forall_x \in \tau \\ 0, & if\ j \neq t^k(x), \forall_x \in \tau \end{Bmatrix} \tag{17}$$

6.2. Algorithm for BS side

A BS must be aware of both its own traffic load and the compute load of the relevant fog node in order to estimate the intermediary IoT association. For each IoT device in the iteration, it must therefore estimate an intermediate IoT association. Then, using their side algorithm, IoT devices choose their BSs or fog nodes depending on the predicted load information among BSs, and the current IoT device association in the k^{th} iteration changes to $\eta_j(k)$ as a result. As a result, BS 'j' can assessment IoT association an intermediate for location 'x' in the following iteration using the intermediate $\alpha_j^{\sim k}(x)$ (estimated by a BS) and the current IoT device association $\alpha_j^k(x)$ (determined by IoT devices) equations:

$$\alpha_J^{\sim K+1}(x) = (1-£)\alpha_j^k(x) + £\alpha_j^{\sim k}(x) \tag{18}$$

where £ is represented system parameter with a range $0 \leq £ \leq 1$, if devices of IoT are associate with (k+1), then the traffic load has been calculated by the (19).

$$\eta_j(k+1) = \int_{x\in\tau} \frac{\gamma(x)m(x)\alpha_j^{\sim k+1}(x)}{b_j(x)}\, dx \tag{19}$$

Similarly, the next computing traffic load at fog node 'j' is obtained from (20).

$$\hat{\eta}_j(k+1) = \int_{x\in\tau} \frac{\gamma(x)v(x)\alpha_j^{\sim k+1}(x)}{e_j(x)}\, dx \tag{20}$$

The BS selected algorithm has been done by using algorithm 1.

Algorithm 1. The BS side algorithm
Input: BS choice for IoT devices:$t^k(x)$, $\forall_x \in \tau$. The kth iteration's intermediate IoT device association vector, α_j^k.
Output: The estimated traffic loads of BSs $\eta\ (k+1)$ and the computing loads is estimated of fog nodes $\hat{\eta}(k+1)$ in the (k+1) th.
The estimated loads computing of fog nodes $\hat{\eta}(k+1)$ in the $(k+1)$ th and the estimated traffic loads of BSs in the $\eta\ (k+1)$ th.

a. Update the IoT device association intermediate for different locations based on: $\alpha_J^{\sim K+1}(x) = (1-£)\alpha_j^k(x) + £\alpha_j^{\sim k}(x), x \in \tau, j \in F$.
b. Calculate $\eta_j(k+1)$ and $\hat{\eta}_j(k+1)$ based on (19) and (20);
c. Return $\eta_j(k)$ and $\hat{\eta}_j(k+1)$.

Treated in algorithm 1. Equation (21) contains the feasible set for problem P1, as is well known.

$$\widehat{M} = \left\{\alpha | \eta_j = \int_{x\in\tau} \frac{\gamma(x)m(x)\alpha_j(x)}{b_j(x)}\, dx, \right\} \tag{21}$$

where $\alpha_j(x) \in \{0,1\}, 0 \leq \eta_j \leq \eta_{max}$ and $\sum_{j\in F} \alpha_j(x) = 1, \forall_j \in F, \forall_x \in \tau$
Treated in algorithm 1. As we are aware, the equation (22) can be used to define the feasible set of problem P1.

$$\begin{gathered} M = \left\{\alpha | \eta_j = \int_{x\in\tau} \frac{\gamma(x)m(x)\alpha_j(x)}{b_j}\, dx\right\} \\ \alpha_j(x) \in \{0,1\}, 0 \leq \eta_j \leq \eta_{max} \\ \sum_{j\in F} \alpha_j(x) = 1, \forall_j \in F, \forall_x \in \tau \end{gathered} \tag{22}$$

We must employ IoT association an intermediate value to lower the average latency ratio $L()$ in each iteration because the traffic load (or computing load) vector for problem P1 is not convex, as indicated by the $\alpha_j(x) \in \{0,1\}$. We demonstrate that the load vectors computing and traffic load will converge to be in the feasible set by first relaxing to make $0 \leq \alpha_k \leq 1$. Given in (23), the feasible set relaxed for problem P1 can be expressed.

$$\widehat{M} = \left\{ \alpha | \eta_j = \int_{x \in \tau} \frac{\gamma(x) m(x) \alpha_j(x)}{b_j(x)} dx, \right\}$$
$$\alpha_j(x) \in \{0,1\}, 0 \leq \eta_j \leq \eta_{max}$$
$$\sum_{j \in F} \alpha_j(x) = 1, \forall_j \in F, \forall_x \in \tau \tag{23}$$

Theorem 1: The feasible relaxed set $\widehat{M}$ is a convex set.
Proof: As the set $\widehat{M}$ includes any convex combination of α, it is defined in $\widehat{M}$.
Theorem 2: The convex function of α, when is defined by objective function (α), when 'α' is defined in $\widehat{M}$.
Proof: $\nabla^2 \tau(\alpha) > 0$ during (α) is defined in $\widehat{M}$ proved by showing this equation.

6.3. Analysis of the algorithm

In this part, we will examine the LAB scheme's convergence and optimality with regard to problem P1.
Theorem 3: Gives a direction for $\tau(\hat{\alpha})$ at $\hat{\alpha}^k$, when $\hat{\alpha}^{k+1} \neq \hat{\alpha}^k$, $\hat{\alpha}^{k+1}$.
Proof: As $0 \leq \hat{\alpha}_j^k(x) \leq 1$, $\tau(\hat{\alpha})$ is defined in $\widehat{M}$, As shown in Theorem 2 $\tau(\hat{\alpha})$ is a convex function of $\hat{\alpha}$, and thus, we need to prove $\langle \nabla C(\tilde{\alpha}^k), \alpha^{\sim k+1} - \tilde{\alpha}^k \rangle < 0$ thus, we have (24).

$$\langle \nabla C(\tilde{\alpha}^k), \alpha^{\sim k+1} - \tilde{\alpha}^k \rangle$$
$$= \int_{x \in \tau} \sum_{j \in F} \gamma(x) v(x) \frac{\alpha_J^{\sim K+1}(x) - \alpha_J^{\sim K}(x)}{e_j b_j(x) \psi_j(k)}$$
$$= \int_{x \in \tau} \gamma(x) v(x) \frac{\alpha_J^{\sim K+1}(x) - \alpha_J^{\sim K}(x)}{e_j b_j(x) \psi_j(k)} \tag{24}$$

Based on (18), we can compute using (25).

$$\alpha_J^{\sim K+1}(x) - \alpha_J^{\sim K}(x) = (1 - \pounds)(\alpha_j^k(x) + \pounds \alpha_j^{\sim k}(x)) \tag{25}$$

As we know that, in (26).

$$\alpha_J^{\sim K}(x) = \begin{Bmatrix} 1, & if\ j = t^k(x) \\ 0, & if\ j = t^k(x) \end{Bmatrix} \tag{26}$$

The k^{th} iteration is adopted for BS selection rule at the destine side, then it can be achieved by the equations $t^k(x) = arg \max_{j \in F} e_j(x) b_j(x) \psi_j(k)$, from we can derive the (27) and (28):

$$\sum_{j \in F} (1 - \pounds) \frac{\alpha_j^k(x) - \alpha_J^{\sim K}(x)}{e_j b_j(x) \psi_j(k)} \leq 0 \tag{27}$$

$$since\ \alpha_J^{\sim K+1}(x) \neq \alpha_J^{\sim K}(x)$$

$$\sum_{j \in F} (1 - \pounds) \frac{\alpha_j^k(x) - \alpha_J^{\sim K}(x)}{e_j b_j(x) \psi_j(k)} < 0 \tag{28}$$

Thus, we have proved $\langle \nabla C(\tilde{\alpha}^k), \alpha^{\sim k+1} - \tilde{\alpha}^k \rangle < 0$.

By proving the subsequent theorem as part of the LAB scheme, we will also examine if the optimal selection rule for BS each iteration at the IoT device side is the best choice.
Theorem 1: The ideal IoT device association rule at the fog node side is the following given the advertised traffic loads of BSs and computing loads of fog nodes $t^k(x) = arg \max_{j \in F} e_j(x) b_j(x) \psi_j(k)$, where x is a feature vector associated with an IoT device.
Proof: The proposed kth iteration side algorithm for IoT side algorithm achieves association IoT device given by $t^k(x) = arg \max_{j \in F} e_j(x) b_j(x) \psi_j(k)$ Meanwhile, let 'α' denote any other possible IoT devices, with a condition and (29).

$$
\begin{aligned}
&< \nabla C(\alpha^k), \alpha' - \alpha^k \; > 0 \\
&< \nabla C(\alpha^k), \alpha' - \alpha^k \; > \\
&= \int_{x\in\tau} \sum_{j\in F} \gamma(x) v(x) \left(\alpha' - \alpha_j^k(x)\right) \frac{1}{e_j b_j(x) \psi_j(k)} dx \\
&= \int_{x\in\tau} \gamma(x) v(x) \sum_{j\in F} \left(\alpha' - \alpha_j^k(x)\right) \frac{1}{e_j b_j(x) \psi_j(k)} \, dx
\end{aligned} \tag{29}
$$

since (30)

$$
\begin{aligned}
&t^k(x) = arg \min_{j\in F} e_j(x) b_j(x) \psi_j(k) \\
&\alpha_j^k(x) = \begin{Bmatrix} 1, & if\ j = t^k(x)\, , \forall_x \in \tau \\ 0, & if\ j \neq t^k(x)\, , \forall_x \in \tau \end{Bmatrix}
\end{aligned} \tag{30}
$$

then, we have

$$
\sum_{j\in F} \alpha_j'(x) \frac{1}{e_j b_j(x) \psi_j(k)} \;\; \geq \;\; \sum_{j\in F} \alpha_j^k(x) \frac{1}{e_j b_j(x) \psi_j(k)}
$$

hence, $\langle \nabla C(\alpha^k), \alpha' - \alpha^k \rangle > 0$. Therefore, α^k is an optimal IoT device association in the k^{th} iteration.

7. NUMERICAL RESULTS

In this section, we present simulations to gauge how well our proposed approach works. In this part, we compare it to the Best SNR algorithm and distributed algorithm. The fundamental goal of the distributed algorithm is to divide workloads of traffic among BSs as efficiently as possible in view to reduce communications delay ratio (i.e., $\sum_{j\in F} \vartheta_j$) without taking load distribution at fog nodes into account. The Best SNR algorithm, on the other hand, aims to link IoT devices with BSs that offer the conditions best channel. The Table 1 displays the simulation parameters for a comparison of proposed and standard methods' performance.

Table 1. Simulation parameter

SI.NO	Parameter	Range
1.	Number of BSs	6
2.	Total network area	3,000×2,000 m^2
3.	Number of areas divided	15,000 Location
4.	Location area	20×20 m
5.	Mean arrival rate per unit area is set	0.50 flows/second
6.	Mean traffic area	0.05 Mbits

Figure 3 shows the network topology BSs distribution with respect to the parameter consider in Table 1. The Figure 4 shown the performance analysis between proposed method and conventional methods average latency ratio $L(\alpha)$ with respect to the $\gamma = 0.6, e_j = 7.6 * 10^6$. The average delay ratio $L()$ for the various algorithms is shown in Figure 5. Figure 6 displays the communication delay (average) ratio in relation to the number of iterations, $\gamma = 0.6, e_j = 7.6 * 10^6$, whereas Figures 7 and 8 display the compute loads placed on various fog nodes within a network structure.

Figure 9 shows the average delay ratio in terms of each fog's capacity ($\gamma = 0.5$), and Figure 10 compares the performance of the suggested method to that of the more traditional approach in terms of the flow arrival rate (x), where $\gamma\,(x)\,, e_j = 7.6 * 10^6$. As a result, as compared to fuzzy golden eagle load balancing (FGELB), the compute load balancing of reconfigurable load balancing algorithm (RLBA) is unable to reduce average latency. However, as the average traffic arrival rate rises, both the traffic loads and compute load in a network become heavy, which causes the average latency ratio of RLBA to develop slowly while the performance of the FGELB declines quickly. In the present scenario, average latency is impacted by both traffic loads among BSs and computing loads among fog nodes. However, this method does not concentrate on balancing traffic loads between BSs but simply balances compute loads across fog nodes. As RLB takes into account both load balancing algorithms, it can still maintain low average latency in spite of large computing loads. This indicates that some fog nodes are crowded, particularly when the network is experiencing heavy compute load.

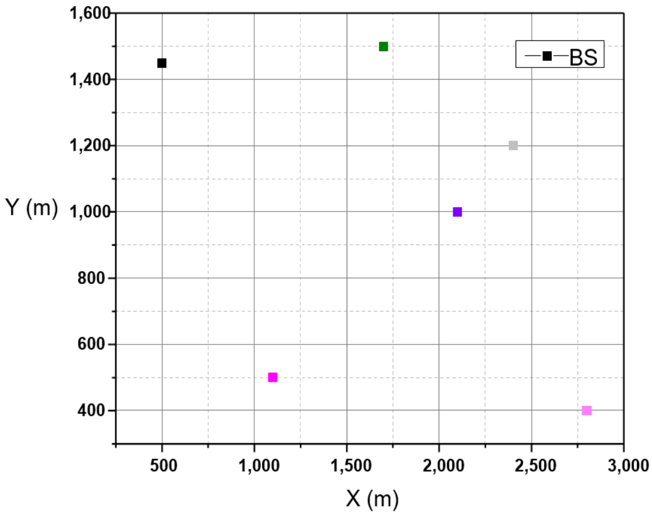

Figure 3. Fundamental base station distribution in network topology

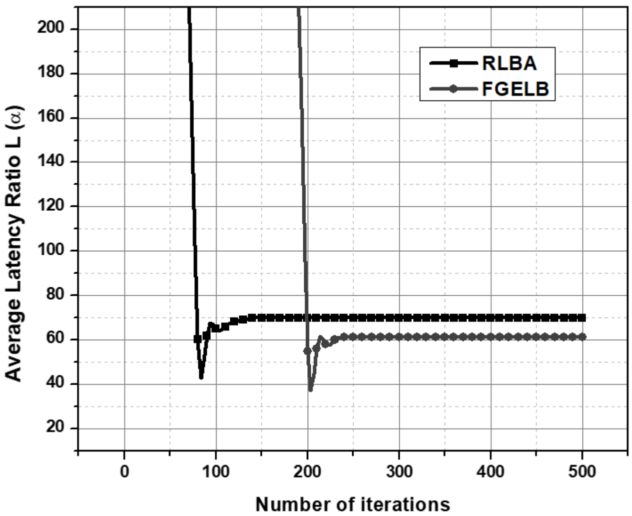

Figure 4. The average latency ratio $L(\alpha)$ performance analysis between proposed and conventional methods

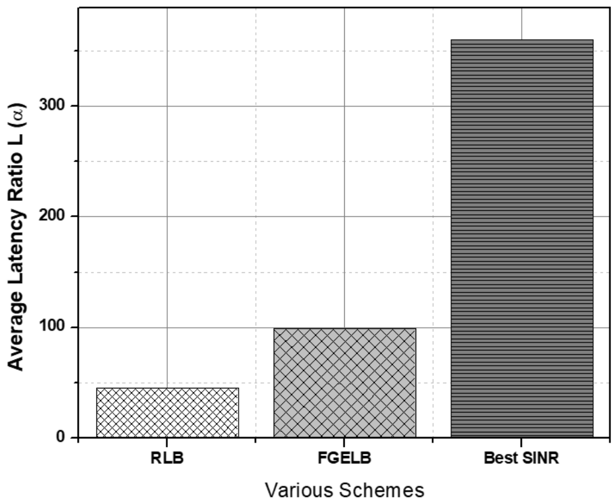

Figure 5. The average latency ratio $L(\alpha)$ for the different algorithms with respect to the $\gamma = 0.6, e_j = 7.6 * 10^6$

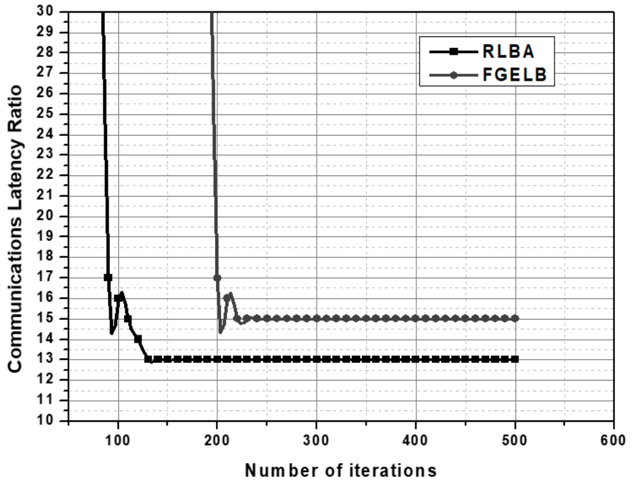

Figure 6. The average latency ratio $L(\alpha)$ performance analysis between proposed and conventional methods

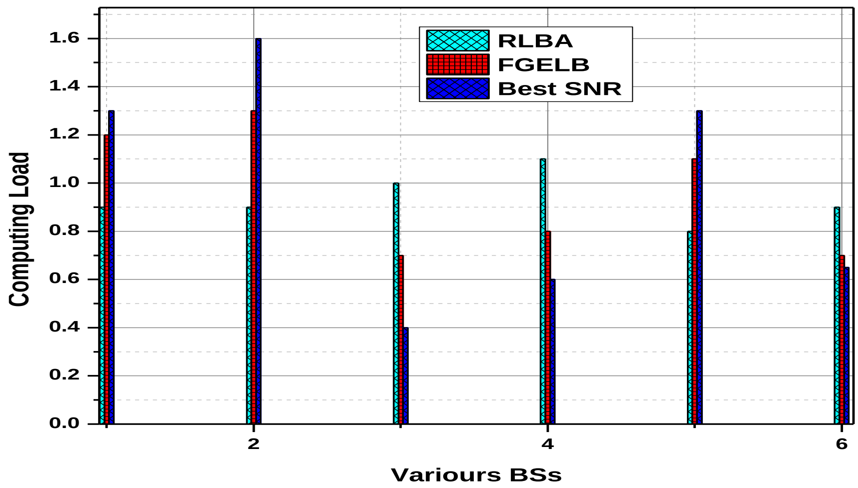

Figure 7. Computing loads of different fog nodes

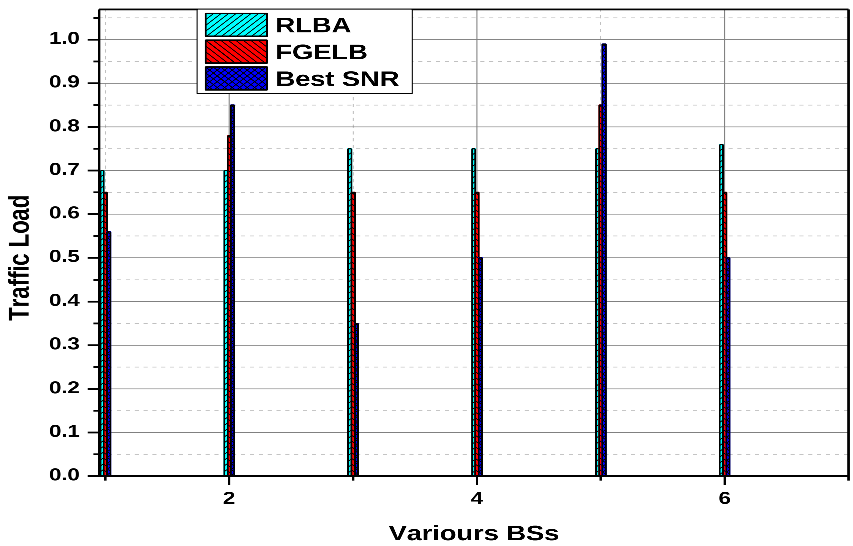

Figure 8. Various BSs traffic loads

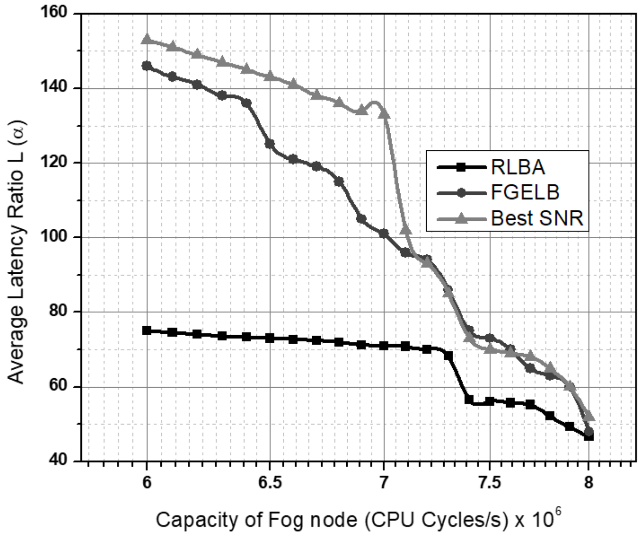

Figure 9. The performance analysis between proposed and conventional methods with respect to $\gamma = 0.6\,, e_j = 7.6 * 10^6$

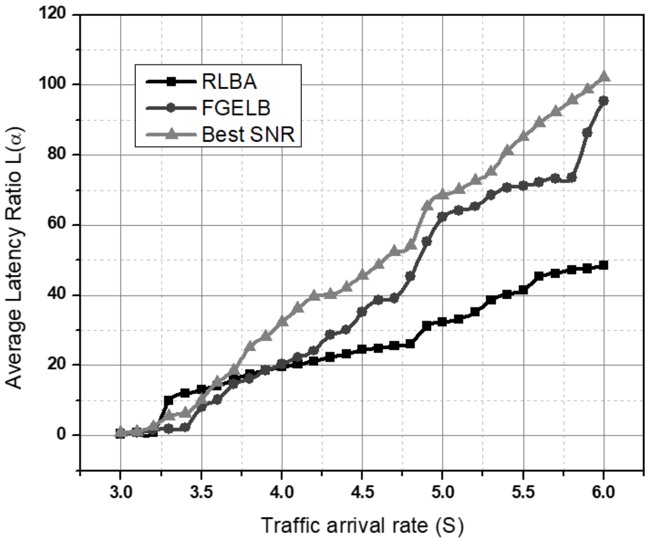

Figure 10. The performance analysis between proposed and conventional methods with respect to $\gamma = 0.6\,, e_j = 7.6 * 10^6$

8. CONCLUSION

In this paper, the main concentration is on the proper workload distribution in smart cities using a RLBA. When a huge number of data flows across the IoT device in a fog computing network, there may be a loss of information, more attenuation, and unbalancing task in fog nodes. to overcome the above-mentioned drawback, a proposed RLBA has been used to optimize the workload by considering a few standard parameters in the fog computing network. With respect to the average latency ratio, $L(\alpha)$ performance analysis has been done by considering ($\gamma = 0.6, e_j = 7.6 * 10^6$.) the proposed RLBA. The result shows better performance as compared to the conventional method of about 1.6% with respect to the communication latency ratio. As per the analysis of performance that has been done by considering ($\gamma = 0.6, e_j = 7.6 * 10^6$), the proposed RLBA shows better performance as compared to the conventional method of about 0.96%. With respect to the computing load, the proposed method shows better performance of about 0.26% and 1.21% as compared to FGELB and best SNR respectively. With respect to the traffic load, the proposed method RLBA optimized to 0.98% and 1.25% as compared to FGELB and best SNR respectively.

FUTURE SCOPE

The emerging technology is six generation (6G), our proposed method is capable to operate the 6G communication in a fog computing network using IoT devices without lose of information and attention.

ACKNOWLEDGEMENTS

The author would like to thank East Point College of Engineering and Technology, Sapthagiri College of Engineering, BNM Institute of Technology, Visvesvaraya Technological University (VTU), Belagavi for all the support and encouragement provided by them to take up this research work and publish this paper.

REFERENCES

[1] Z. Liu, Y. Yang, Y. Chen, K. Li, Z. Li, and X. Luo, "A multi-tier cost model for effective user scheduling in fog computing networks," in *IEEE INFOCOM 2019-EEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 1–6, doi: 10.1109/INFCOMW.2019.8845252.

[2] S. Thazeen, S. Mallikarjunaswamy, and M. N. Saqhib, "Septennial adaptive beamforming algorithm," in *2022 International Conference on Smart Information Systems and Technologies (SIST)*, Apr. 2022, pp. 1–4, doi: 10.1109/SIST54437.2022.9945753.

[3] B. Jia, H. Hu, Y. Zeng, T. Xu, and Y. Yang, "Double-matching resource allocation strategy in fog computing networks based on cost efficiency," *Journal of Communications and Networks*, vol. 20, no. 3, pp. 237–246, Jun. 2018, doi: 10.1109/JCN.2018.000036.

[4] S. Mallikarjunaswamy, N. M. Basavaraju, N. Sharmila, M. H. N, S. Pooja, and B. L. Deepak, "An efficient big data gathering in wireless sensor network using reconfigurable node distribution algorithm," in *2022 Fourth International Conference on Cognitive Computing and Information Processing (CCIP)*, Dec. 2022, pp. 1–6, doi: 10.1109/CCIP57447.2022.10058620.

[5] C. Lin, G. Han, X. Qi, M. Guizani, and L. Shu, "A distributed mobile fog computing scheme for mobile delay-sensitive applications in SDN-enabled vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5481–5493, May 2020, doi: 10.1109/TVT.2020.2980934.

[6] Q. Jin, R. Lin, H. Zou, and F. Yang, "A distributed fog computing architecture supporting multiple migrating mode," in *2018 5th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2018 4th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, Jun. 2018, pp. 218–223, doi: 10.1109/CSCloud/EdgeCom.2018.00046.

[7] F. M. Siddiqui, "Fog computing application for effective fronthaul management in fifth generation networks," in *2018 International Conference on Computing, Electronics and Communications Engineering (iCCECE)*, Aug. 2018, pp. 107–112, doi: 10.1109/iCCECOME.2018.8658911.

[8] S. Jin, Z. Zhu, Y. Yang, M.-T. Zhou, and X. Luo, "Alternate distributed allocation of time reuse patterns in Fog-enabled cooperative D2D networks," in *2017 IEEE Fog World Congress (FWC)*, Oct. 2017, pp. 1–6, doi: 10.1109/FWC.2017.8368532.

[9] S. Pooja, S. Mallikarjunaswamy, and N. Sharmila, "Image region driven prior selection for image deblurring," *Multimedia Tools and Applications*, vol. 82, no. 16, pp. 24181–24202, Jul. 2023, doi: 10.1007/s11042-023-14335-y.

[10] S. Kitanov and T. Janevski, "Energy efficiency of fog computing and networking services in 5G networks," in *IEEE EUROCON 2017-17th International Conference on Smart Technologies*, Jul. 2017, pp. 491–494, doi: 10.1109/EUROCON.2017.8011159.

[11] L. Velasco and M. Ruiz, "Flexible fog computing and telecom architecture for 5G networks," in *2018 20th International Conference on Transparent Optical Networks (ICTON)*, Jul. 2018, pp. 1–4, doi: 10.1109/ICTON.2018.8473934.

[12] L. Yang, H. Zhu, H. Wang, H. Qian, and Y. Yang, "Incentive propagation mechanism of computation offloading in fog-enabled D2D networks," in *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*, Nov. 2018, pp. 1–4, doi: 10.1109/ICDSP.2018.8631842.

[13] V. Marbukh, "Towards fog network utility maximization (FoNUM) for managing fog computing resources," in *2019 IEEE International Conference on Fog Computing (ICFC)*, Jun. 2019, pp. 195–200, doi: 10.1109/ICFC.2019.00032.

[14] M. De Donno and N. Dragoni, "Combining AntibIoTic with fog computing: AntibIoTic 2.0," in *2019 IEEE 3rd International Conference on Fog and Edge Computing (ICFEC)*, May 2019, pp. 1–6, doi: 10.1109/CFEC.2019.8733144.

[15] S. Ju *et al.*, "Optimal county-level crop yield prediction using MODIS-based variables and weather data: A comparative study on machine learning models," *Agricultural and Forest Meteorology*, vol. 307, Sep. 2021, doi: 10.1016/j.agrformet.2021.108530.

[16] K. Tadesse *et al.*, "Effects of preceding crops and nitrogen fertilizer on the productivity and quality of malting barley in tropical environment," *Heliyon*, vol. 7, no. 5, May 2021, doi: 10.1016/j.heliyon.2021.e07093.

[17] B. Kuzman *et al.*, "Estimation of optimal fertilizers for optimal crop yield by adaptive neuro fuzzy logic," *Rhizosphere*, vol. 18, Jun. 2021, doi: 10.1016/j.rhisph.2021.100358.

[18] D. Elavarasan and P. M. D. Vincent, "Crop yield prediction using deep reinforcement learning model for sustainable Agrarian applications," *IEEE Access*, vol. 8, pp. 86886–86901, 2020, doi: 10.1109/ACCESS.2020.2992480.

[19] V. Tomar *et al.*, "Rice equivalent crop yield assessment using MODIS sensors' based MOD13A1-NDVI data," *IEEE Sensors Journal*, vol. 14, no. 10, pp. 3599–3605, Oct. 2014, doi: 10.1109/JSEN.2014.2329185.

[20] W.-J. Hu, J. Fan, Y.-X. Du, B.-S. Li, N. Xiong, and E. Bekkering, "MDFC-ResNet: An agricultural IoT system to accurately recognize crop diseases," *IEEE Access*, vol. 8, pp. 115287–115298, 2020, doi: 10.1109/ACCESS.2020.3001237.

[21] L. Momesso, C. A. C. Crusciol, H. Cantarella, K. S. Tanaka, G. A. Kowalchuk, and E. E. Kuramae, "Optimizing cover crop and fertilizer timing for high maize yield and nitrogen cycle control," *Geoderma*, vol. 405, Jan. 2022, doi: 10.1016/j.geoderma.2021.115423.

[22] A. Shan *et al.*, "Effects of straw return with N fertilizer reduction on crop yield, plant diseases and pests and potential heavy metal risk in a Chinese rice paddy: A field study of 2 consecutive wheat-rice cycles," *Environmental Pollution*, vol. 288, Nov. 2021, doi: 10.1016/j.envpol.2021.117741.

[23] S. Yang *et al.*, "Integration of crop growth model and random forest for winter wheat yield estimation from UAV hyperspectral imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 6253–6269, 2021, doi: 10.1109/JSTARS.2021.3089203.

[24] P. Dayananda, M. Srikantaswamy, S. Nagaraju, R. Velluri, and D. M. Kumar, "Efficient detection of faults and false data injection attacks in smart grid using a reconfigurable Kalman filter," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 13, no. 4, pp. 2086–2097, Dec. 2022, doi: 10.11591/ijpeds.v13.i4.pp2086-2097.
[25] S. Mallikarjunaswamy, "Implementation of an effective hybrid model for islanded microgrid energy management," *Indian Journal of Science and Technology*, vol. 13, no. 27, pp. 2733–2746, Jul. 2020, doi: 10.17485/IJST/v13i27.982.
[26] P. Satish, M. Srikantaswamy, and N. K. Ramaswamy, "A comprehensive review of blind deconvolution techniques for image deblurring," *Traitement du Signal*, vol. 37, no. 3, 2020.
[27] S. Thazeen, S. Mallikarjunaswamy, M. N. Saqhib, and S. N, "DOA method with reduced bias and side lobe suppression," in *2022 International Conference on Communication, Computing and Internet of Things (IC3IoT)*, Mar. 2022, pp. 1–6, doi: 10.1109/IC3IOT53935.2022.9767996.
[28] S. Pooja, S. Mallikarjunaswamy, and N. Sharmila, "Adaptive sparsity through hybrid regularization for effective image deblurring," *Indian Journal of Science and Technology*, vol. 14, no. 24, pp. 2051–2068, Jun. 2021, doi: 10.17485/IJST/v14i24.604.
[29] S. Rathod, N. K. Ramaswamy, M. Srikantaswamy, and R. K. Ramaswamy, "An efficient reconfigurable peak cancellation model for peak to average power ratio reduction in orthogonal frequency division multiplexing communication system," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 6, pp. 6239–6247, Dec. 2022, doi: 10.11591/ijece.v12i6.pp6239-6247.
[30] S. Thazeen and S. Mallikarjunaswamy, "The effectiveness of 6T beamformer algorithm in smart antenna systems for convergence analysis," *IIUM Engineering Journal*, vol. 24, no. 2, pp. 100–116, Jul. 2023, doi: 10.31436/iiumej.v24i2.2730.
[31] H. N. Mahendra and S. Mallikarjunaswamy, "An efficient classification of hyperspectral remotely sensed data using support vector machine," *International Journal of Electronics and Telecommunications*, vol. 68, 2022.
[32] H. N. Mahendra *et al.*, "Deep learning models for inventory of agriculture crops and yield production using satellite images," in *2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon)*, Oct. 2022, pp. 1–7, doi: 10.1109/MysuruCon55714.2022.9972523.
[33] H. N. Mahendra, S. Mallikarjunaswamy, C. B. Nooli, M. Hrishikesh, N. Kruthik, and H. M. Vakkalanka, "Cloud based centralized smart cart and contactless billing system," in *2022 7th International Conference on Communication and Electronics Systems (ICCES)*, Jun. 2022, pp. 820–826, doi: 10.1109/ICCES54183.2022.9835856.
[34] S. Mallikarjunaswamy, N. Sharmila, G. K. Siddesh, K. R. Nataraj, and M. Komala, "A novel architecture for cluster based false data injection attack detection and location identification in smart grid," in *Lecture Notes in Mechanical Engineering*, Springer Singapore, 2022, pp. 599–611.

BIOGRAPHIES OF AUTHORS

Nandini Gowda Puttaswamy currently working as Asst. Professor in Information Science and Engineering, Sapthagiri College of Engineering, Bangalore. She completed her B E in CSE from VTU, M. Tech. in Software Engineering from VTU and pursuing Ph.D. from Visvesvaraya Technological University. She has published around 4 papers on National Conference and her area of research interest are cloud computing, fog computing, edge computing and IoT, artificial intelligence (AI), machine learning (ML), big data analytics. She can be contact at email: nandini.educator@gmail.com.

Anitha Narasimha Murthy currently working as Professor in Computer Science and Engineering, BNM Institute of Technology, Bangalore. She completed her B E in CSE from Bangalore University, M. Tech. in Information Technology from Bangalore University and Ph.D. from Visvesvaraya Technological University. She has published around 30 research papers and her area of research interest are AI, ML, big data analytics, data mining. She can be contacted at email: anitha.mhp@gmail.com.