The Journal of Supercomputing (2019) 75:7125–7146
https://doi.org/10.1007/s11227-019-02928-0

IoT-F2N: An energy-efficient architectural model for IoT using Femtolet-based fog network

Anwesha Mukherjee[1] · Priti Deb[2] · Debashis De[2,3] · Rajkumar Buyya[4]

Published online: 11 June 2019

Abstract

Energy- and latency-optimized Internet of Things (IoT) is an emerging research domain within fifth-generation (5G) wireless network paradigm. In traditional cloud-centric IoT the sensor data processing and storage occurs inside remote cloud servers, which increase delay and energy consumption. To reduce the delay and energy consumption, an IoT paradigm is proposed using 5G device Femtolet-based fog network. In this architecture, the data obtained from sensors are processed and maintained inside the edge and fog devices. The Femtolet works as an adaptable fog device and it expands and shrinks coverage according to user's presence. A mathematical model is developed for the proposed paradigm. The delay and power consumption in the proposed model are determined. Qualnet 7 is used for simulating the proposed model. The results of simulation illustrate that the proposed architectural model reduces the energy consumption and delay by approximately 25% and 43% respectively than the fog computing-based existing IoT paradigm. The comparative analysis with the existing IoT paradigm shows that IoT using Femtolet-based fog network is a green and efficient approach.

Keywords Internet of Things · Fog computing · Femtolet · Fog network · Power · Delay

✉ Debashis De
dr.debashis.de@gmail.com

1 Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, Kharagpur, West Bengal 721302, India

2 Department of Computer Science and Engineering, Centre of Mobile Cloud Computing, Maulana Abul Kalam Azad University of Technology, West Bengal, BF-142, Salt Lake, Sector-I, Kolkata, West Bengal 700064, India

3 Department of Physics, University of Western Australia, 35 Stirling Hwy, Crawley, WA 6009, Australia

4 Cloud Computing and Distributed Systems (CLOUDS) Lab, School of Computing and Information Systems, The University of Melbourne, Melbourne, Australia

1 Introduction

The connection of embedded devices within an existing Internet infrastructure, where the devices are uniquely identified and computing environment is created, is referred to IoT paradigm [1]. In traditional IoT, data captured by the sensor nodes are transferred to the servers through a mobile device. The cloud servers process and maintain the received data. The mobile device is linked with the cloud servers via a base station or an access point. Nowadays femtocell is used as a low power indoor base station [2]. The transmission power of a femtocell is less than a large cell base station and provides good signal strength to the users at indoor region. A femtocell can switch to active and idle modes based on user's presence inside its coverage, to reduce the power consumption [3–5]. The users can access the cloud servers through the femtocell at indoor region. However, data processing inside the remote cloud servers increases communication delay and power consumption [6]. For hard deadline applications, a delay is a vital parameter and battery life is a major concern for the mobile devices. For delay and energy optimization, fog computing is introduced [7]. Integration of IoT and fog computing is an emerging field. In a fog network, the data processing occurs inside the intermediate devices, e.g. gateway, router between the end node, e.g. sensor nodes and the remote cloud servers (see Fig. 1).

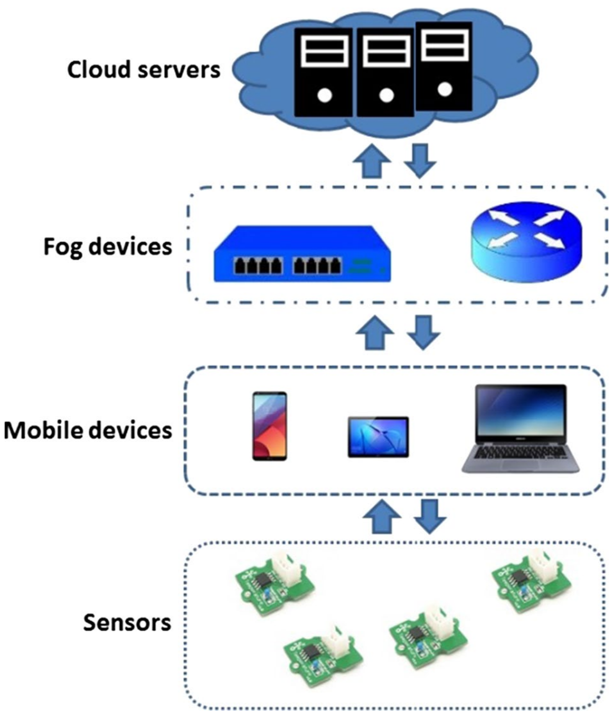


Fig. 1 Fog-based IoT framework with sensor nodes, mobile devices, fog devices and cloud servers

A mobile device is connected to the cloud servers through the fog devices, and the mobile device is connected with the fog device through an access point or base station. However, an indoor base station with computation ability (e.g. Femtolet [8], small cell cloud-enhanced e-nodeB [9]) can also work as a fog device and in that case the mobile device is connected with the cloud servers through that device. Small cell cloud-enhanced e-nodeB (SCceNB) is an indoor base station with limited computation and storage ability [9]. On the other hand, Femtolet is a device possessing the attributes of femtocell and cloudlet [8]. Femtolet is an indoor base station that contains storage as well as has the computation ability. The users under a Femtolet can store data as well as offload applications inside it. Use of Femtolet as a fog device for code offloading has been discussed in [10].

In fog-based IoT framework, the sensor nodes are used to collect environmental object status, and the collected data are sent to the cloud servers, where the intermediate fog devices also perform data processing. The fog-based IoT framework using sensors, mobile devices, fog devices (e.g. switch, routers), and cloud servers, is presented in Fig. 1. In the present work an IoT framework is proposed where the Femtolet is used as a fog device. The proposed architecture is referred as IoT using Femtolet-based Fog Network (IoT-F2N).

There are several application areas of IoT such as smart home, health care. However, all the real-time applications require energy-efficiency and delay optimization in service provisioning. In existing cloud-centric real-time IoT applications such as smart home [11], health care [12–14], the use of remote cloud increases the energy consumption of the user device as well as the delay in providing service. If IoT-F2N is applied in such real-time applications the delay as well as energy consumption of the user device will be reduced. IoT-F2N can be applied in smart home monitoring for faster decision making in adverse situation. If the Femtolet detects any abnormality in the collected data from the IoT devices in a smart home environment, for example, very high room temperature, gas leakage, it sends an alert message to the local cloud server admin to take care of the incident. Health monitoring is another application where latency is a crucial parameter. If IoT-F2N is applied in health monitoring, at low latency indoor health monitoring will be possible due to the use of indoor base station as a fog device. Explosive detection [15] is another application of IoT where the use of Femtolet can help to alert people if an explosive is detected. In this way IoT-F2N can be applied in several real-time applications to provide energy as well as delay optimization.

The rest of the paper is organized as follows: Sect. 2 explains related works on fog computing and IoT. Section 3 proposes the IoT-F2N architecture and its working model. Section 4 evaluates the performance of IoT-F2N in terms of theoretical and simulation analysis. Finally Sect. 5 concludes along with future research scope of IoT-F2N.

2 Related work

IoT has become an emerging domain in the field of smart computing. Sensor network is a major component of IoT. A Wireless Sensor Network (WSN) is formed with huge number of sensor nodes, which sense and report to a central node [16, 17]. These embedded devices connect with each other through Internet and form an infrastructure which is known as IoT. IoT has the potentiality and capability to find out novel solutions in different areas like healthcare, industrial automation, transportation business, environmental monitoring, etc. [16–18].

Cloud Computing (CC) [19, 20] offers new opportunities in the area of IoT. Sensor cloud platform has been created to assemble data from sensors as per application and direct it to the cloud [21]. However, for real-time applications only cloud-based framework suffers from the increase in delay and energy. To give service to the end users at low latency and low energy, fog computing has come. Energy management in IoT has been discussed in [22, 23]. For large scale IoT applications fog computing has been used in [24]. A resource allocation process for the three-tier IoT-fog network has been discussed in [25]. In the three-tier architecture, the intermediate fog devices between the end nodes and cloud servers participate in data processing.

To minimize the energy and latency than the cloud only framework, cloudlet has also been used [26, 27]. Cloudlet provides cloud services at low power and latency by working as an agent [26, 27]. In mobile cloud computing middleware is used for task delegation [28]. Cloudlet is a middleware which plays a vital role for power and latency aware offloading. Mobile devices offload their data and applications inside the cloud let instead of using the cloud servers [6]. For reducing energy consumption and enhancing the indoor signal strength, femtocell base station is used in cellular network [3–5]. An enhanced device of femtocell, SCceNB, has been proposed in [9], which is nothing but a small cell base station with limited computation ability. To integrate the properties of femtocell and cloudlet Femtolet has been proposed in [8]. Use of Femtolet as a fog device for code offloading has been discussed in [10]. Fog computing enables computation inside the network devices at different classified levels with several degrees of computational and storage capabilities [7].

The combination of IoT and fog network increases the development of wireless network as well as reduces the energy consumption because data are processed nearer than cloud servers [24, 25]. Quality of Service (QoS) in fog-based IoT has been discussed in [29]. The QoS in IoT has been discussed in [30]. In [31–33] security and privacy issues related to fog computing have been discussed. The use of blockchain for security in IoT has been highlighted in [34, 35]. Use of blockchain has been highlighted in these articles. An IoT security middleware for cloud and fog has been discussed in [36]. Offloading in fog-based IoT has been discussed in [37]. A service management framework for cloud-based IoT has been discussed in [38]. The relationships between different aspects of IoT, e.g. Internet of Everything (IoE), Internet of Mobile Things (IoMT), Internet of Mission-Critical Things (IoMCT), Internet of Nano Things (IoNT), etc. has been studied in [39]. To find smart solutions in daily life, IoT is a principle component. For monitoring and control home appliances in a smart home environment, natural language processing has been used with text and audio commands in [40]. For indoor environment monitoring, machine learning has been used in [41]. From the real life perspective such as health care, traffic manufacturing, IoT network has been discussed in [42]. The role of 5G enabled devices in IoT has been highlighted in [43]. For smart health care, fog-based IoT has been discussed in [44]. For health care system, fog computing has been used in [14] also. Energy-efficient offloading for fog-cloud network-based IoT applications has been discussed in [45].

The fog-based IoT has solved the difficulties in energy and delay management with respect to the only-cloud-based IoT system. However, for accessing a fog device, an end user has to be connected with the base station or access point. Our motivation is to propose an IoT framework, which will use a base station as a fog device to reduce the delay and energy even than the fog-based existing IoT framework. To store and process data inside a nearby fog device, Femtolet is used in this paper, which is an indoor base station with storage and computation ability. Our aim is to use Femtolet-based fog computing to create an energy-efficient and low-latency IoT framework for indoor zone. In our paper adaptable Femtolet is used. In [46] adaptable femtocell has been proposed based on octopus algorithm [46]. The octopus algorithm has been proposed based on the arm expansion and shrinking features of the Pacific creature octopus during its feeding [46]. An octopus expands its arm to capture and hold food when it is available. The octopus shrinks its arm to put the food into the mouth. Following this feature adaptable femtocell has been discussed in [46]. An adaptable femtocell shrinks and expands coverage according to user's location inside its coverage [46]. In our present article, adaptable Femtolet is used, that can shrink and expand coverage according to user's location inside its coverage. Table 1 illustrates the contributions and novelty of the proposed architectural model with respect to the existing fog computing-based IoT systems. From the table, it is observed that in this work an adaptable indoor base station that can expand and shrink coverage based on user's location, is used as fog device for faster service provisioning, and this is the uniqueness of the proposed framework with respect to the existing models [22, 23, 37].

2.1 Motivation and contributions

From the discussions on the existing works, it has been observed that energy and delay are two vital parameters in the field of QoS-aware IoT. Our motivation is to introduce an energy and latency aware IoT paradigm. The contributions of this article are:

1. A new architectural model for IoT is proposed for indoor users based on 5G device Femtolet. In the proposed IoT-F2N architecture, Femtolet works as a fog device for data processing and storage. The mathematical model of IoT-F2N is developed.
2. Based on octopus algorithm, the Femtolet expands and shrinks coverage according to the user's presence. By using adaptable Femtolet the energy consumption is reduced.
3. The delay and power consumption in proposed IoT-F2N model are calculated and compared with the existing IoT models. The simulation results explain that IoT-F2N is energy-efficient as well as reduces delay than the existing IoT architectures.

Table 1 Comparison between IoT-F2N and existing fog computing-based IoT paradigms

Properties	IoT using (femto-cell + cloudlet)-based fog network	IoT using fog network [22]	Energy-management-as-a-service [23]	Offloading in fog-based IoT [37]	Proposed IoT-F2N
Working principle	Sensor nodes capture object status and send the raw data to the cloudlet via femtocell base station. Here cloudlet participates in data processing	Sensor nodes capture object status and send the raw data to the fog devices via an access point. Here gateway and router serve as fog devices. The fog devices process the raw data	Sensor nodes capture object status and send the raw data to the fog devices via an access point. Here different controlling and computing devices are used as fog devices along with gateway and router. The fog devices process the raw data	Task offloading in fog computing-based IoT has been discussed along with the research challenges. The fog devices are used to execute the tasks	Sensor nodes capture object status and send the raw data to the Femtolet. The Femtolet is an indoor base station which works as a fog device due to its ability of computation and storage
Intermediate device used in processing	Cloudlet	Gateway and Router	Controlling devices, computing devices, gateway and router	Intermediate devices, e.g. micro data centre, nanodata centre, femto cloud	Femtolet
Access point/base station has computation ability	×	×	×	Partially. Femto access points through collaboration can participate in partial task execution	✓
Adaptable access point/base station that expands/shrinks coverage based on mobile device's location	×	×	×	×	✓

3 IoT-F2N architectural model

This section proposes the architecture of IoT-F2N with its working model. IoT-F2N contains the following principle components:

1. Sensors,
2. Mobile device as edge device,
3. Adaptable Femtolet as fog device,
4. Security-gateway,
5. Home node base station-gateway,
6. Local cloud servers,
7. Remote cloud servers.

The pictorial representation of the architecture of IoT-F2N is shown in Fig. 2.

3.1 Three-layer architecture of IoT-F2N

The IoT-F2N architecture is divided into three layers, which are described as follows.

3.1.1 Layer 1: sensors as IoT devices

The sensor devices are attached with different objects. These sensors collect object status and send it to the sensor base station. The sensor base station forwards the collected data to the edge device with which it is connected. These sensor nodes are referred as IoT devices in our approach.

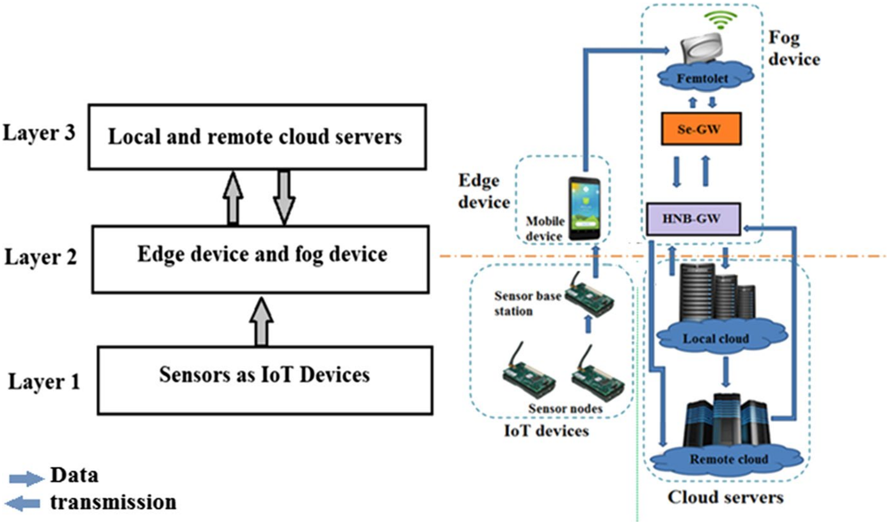


Fig. 2 Architecture and working model of IoT-F2N

3.1.2 Layer 2: edge device and fog device

The mobile device works as an edge device in IoT-F2N. The edge device processes the incoming sensor data (raw data) and then sends the processed data to the Femtolet under which the mobile device is registered. If the edge device cannot process the raw data due to limited processing ability, then the raw data are forwarded to the Femtolet for processing. The mobile device accesses the data inside the Femtolet, or the data maintained inside the remote cloud servers via Femtolet. If the mobile device wants to get back the processed data, it attaches its ID.

The Femtolet is a home base station possessing storage and computation power. Femtolet incorporates the attributes of femtocell and cloudlet [8]. Femtolet contains radio frequency receiver (RFR), field programmable gate array (FPGA) and radio frequency transmitter (RFT) like femtocell, to act as a home base station [8]. Femtolet comprises high storage as well as can offload applications, like cloudlet [8]. The common components of femtocell and cloudlet which are contained in Femtolet are power amplifier, microprocessor and random access memory (RAM) [8]. The Femtolet if receives processed data from the edge device, it stores the data. Otherwise the Femtolet first processes the incoming raw data and then stores the processed data. If the ID of the mobile device is attached, the processed data are sent back to the device. If the data are to be accessed by other agencies, the data are forwarded to the remote cloud servers through the security-gateway (Se-GW) and home node base station-gateway (HNB-GW). Se-GW securely transmits the data through a security tunnel as shown in Fig. 1. If the Femtolet cannot process the data, it sends the data to the local cloud servers.

The Femtolet follows octopus algorithm [46] to shrink and expand its coverage according to the user's presence. The octopus algorithm is based on the feeding behaviour of the Pacific creature octopus [46]. An octopus expands its arm to capture food and then holds it. After that the octopus shrinks the arm and put the food into mouth. Based on this feeding behaviour, adaptable femtocell has been discussed in [46]. As Femtolet is a femtocell with storage and computation ability, the octopus feeding logic can be used in Femtolet also to make it adaptable. In our IoT-F2N such an adaptable Femtolet is used, which can expand and shrink coverage based on user's presence. The Femtolet can cover maximum 20 m area. If the mobile device is located at a distance less than the Femtolet's coverage, the Femtolet shrinks its coverage to the location of the mobile device. If the IoT and edge devices are in idle mode, the Femtolet also goes to idle mode.

3.1.3 Layer 3: local and remote cloud servers

If the local cloud receives raw data, it processes the data and returns the processed data to the Femtolet. If other agency will access the processed data, the local cloud sends the processed data to the remote cloud servers along with the Femtolet ID. If the local cloud is unable to process the raw data, it forwards the data to the remote cloud servers. In this case, the Femtolet ID is sent with the data, so that the remote cloud servers can directly communicate with the Femtolet without involving the local cloud servers. The remote cloud sends the processed data to the Femtolet. If other agency will access the processed data, the remote cloud keeps a copy of the processed data.

3.2 Working model of IoT-F2N

In our architecture, the computation on sensor data is performed inside the intermediate devices between the sensors and the remote cloud servers. The steps of the working model of IoT-F2N are:

1. Sensor nodes collect object status and send the raw data to the mobile device via sensor base station.
2. If the mobile device has the ability to process the data, the mobile device works as an edge device, processes the data and sends the result to the Femtolet. Otherwise the mobile device forwards the raw data to the Femtolet.
3. If the Femtolet receives raw data and it has the ability to process the data, it works as fog device and processes and then stores the data. If the data is to be accessed by an agency, the Femtolet forwards a copy of the processed data to the remote cloud servers. The intended agency accesses the data in the remote cloud servers.
4. If the Femtolet is unable to process the data, the Femtolet forwards the data to the local cloud servers. If the local cloud is able to process the data, the local cloud processes the data and returns it to the Femtolet. If the processed data is to be accessed by an agency, the local cloud forwards the data to the remote cloud servers along with the Femtolet ID.
5. If the Femtolet and local cloud both are unable to process the data, the local cloud servers forward the raw data to the remote cloud servers along with the Femtolet ID. The remote cloud then processes the data and sends back the result to the Femtolet. If the processed data is to be accessed by an agency, the remote cloud keeps a copy of the processed data.

As the Femtolet has a large internal storage, the processed data are stored inside it, from where the users can access their data while belonging to the Femtolet's coverage. If the data are to be accessed by an agency, then only the data are stored inside the remote cloud servers. The distance between the Femtolet and the mobile device is very small as compared to the distance between the mobile device and the remote cloud servers. The processing occurs inside the edge and fog device in most of the cases. Therefore the communication and propagation delays are reduced. Consequently the power consumption of the mobile device is also reduced.

The proposed system is based on Femtolet that contains a cryptographic processor [8, 47]. The Femtolet is connected with the network through the Se-GW and the data is transmitted through a security tunnel. For security purpose, Internet Protocol Security (IPSec) connection is used. Authentication is performed using a hash function and key, referred as Keyed-Hash Message Authentication Code (HMAC)-Secure Hash Algorithm (SHA) [47]. Tripple Data Encryption Standard (Tripple DES)-based Cipher Block Chaining (CBC) and Advanced Data Encryption Standard (AES)-based Cipher Block Chaining (CBC) can also be used [47]. For secure data storage, Elliptic-Curve

Cryptography (ECC) and Hierarchical Identity-Based Cryptography (HIBC) can be used [47]. For security purpose in IoT blockchain can also be used [34, 35].

3.3 Mathematical model of IoT-F2N

The entities of IoT-F2N and their operational functionalities are mathematically defined in this section. It is assumed that the total number of IoT devices straddling transversely and all fog devices are constant eventually. Furthermore, the fog device gives complete coverage for all the IoT devices. The physical and virtual components of IoT-F2N are described as follows.

3.3.1 Mathematical definition of layer-1 components

In IoT-F2N, layer-1 contains the IoT devices which are used to collect object status.

Definition 1 *IoT device* (*IoTD*) An IoTD is denoted by I and defined as [22],

$$I = (I_{\mathrm{id}}, I_{\mathrm{st}}, I_{\mathrm{s}}, A_{\mathrm{IoT}})$$

where I_{id} is the ID of the IoTD that uniquely identifies the device, I_{st} denotes the status of the device, I_{s} denotes the event type sensed by the device and A_{IoT} denotes the application.

Definition 2 (*Status of IoTD*) The status of an *IoTD* is denoted by I_{st}. It can hold either 0 or 1. If $I_{\mathrm{st}}=0$, the device is in active state and if $I_{\mathrm{st}}=0$, the device is in inactive state.

Definition 3 (*Type of IoTD*) The type of an $IoTD(I_{\mathrm{s}})$ refers to the type of an event that the *IoTD* senses. If a set $i = (i_1, i_2, \dots, i_{\mathrm{p}})$ is considered, then an element of the set denotes the type, here I denotes the set of events monitored by *IoTD*, and p is the number of distinct event types.

Definition 4 (*Application of IoTD*) An application A_{IoT} is defined as [22],

$$A_{\mathrm{IoT}} = \left(AI_{\mathrm{id}}, AI_{\mathrm{t}}, AI_{\mathrm{sp}}\right)$$

where AI_{id} is the ID of the application and AI_{t} refers to the use of the application, AI_{sp} states the least amount of system configuration specifications such as primary and secondary memory, processor configuration, version of the operating system, required for executing the application.

3.3.2 Mathematical definition of layer-2 components

In IoT-F2N layer-2 describes the virtual components. Each IoT device from layer-1 is mapped to the edge and fog devices for their computations. The edge and fog computing components defined which reside in the middle of the IoT devices and cloud servers, are defined.

Definition 5 (*Edge device computing instance* (*EDI*)) *EDI* is defined as [22],

$$EDI = \left(E_{id}, FD, d[s]\right)$$

where E_{id} is the distinctive identity of the edge device, *FD* is the fog device through which the *EDI* is associated with the local or remote cloud servers, and $d[s]$ is the third tuple, refers to an array of size s that contains the device IDs of all the edge devices of an *EDI*.

The mapping from the IoT devices of layer-1 to the edge device of layer-2 is many-to-one and it is denoted as,

$$M'(.) : I' \rightarrow EDI'$$

Definition 6 (*Fog computing device* (*FD*)) *FD* is defined as [22, 47],

$$FD = \left(FD_{\text{id}}, FD_{\text{type}}, FD_{\text{sp}}\right)$$

where FD_{id} and FD_{type} are the ID and the type of the fog computing device. FD_{sp} tuple defines the hardware related specification.

The mapping from edge devices to the fog device of layer-2 is many-to-one and it is denoted as,

$$M'(.) : EDI' \rightarrow FD'$$

3.3.3 Mathematical definition of layer-3 components

If the data sent from the IoT devices are not processed in layer-2, then the data are sent to the cloud servers. The local and remote cloud servers are in layer-3.

Definition 7 (*Cloud computing instance* (*CI*)) *CI* is defined as [22, 47],

$$CI = \left(C_{\text{id}}, Cd[s]\right)$$

where C_{id} is the distinctive identity of the cloud component, $Cd[s]$ is the tuple defining an array of size s which contains the processing unit IDs of all the essential cloud servers of a *CI*.

The mapping from the layer-2 to layer-3 components is many-to-one and it is denoted as:

$$M'(.) : FD' \rightarrow CI'$$

Definition 8 (*Cloud Server* (*CS*)) *CS* is defined as [22, 47],

$$CS = \left(CS_{\mathrm{id}}, CS_{\mathrm{type}}, CS_{\mathrm{sp}}\right)$$

where CS_{id} and CS_{type} are ID and type of the cloud server. Type can be local or remote. CS_{sp} tuple defines the hardware related specification.

4 Performance evaluation

In this section both the theoretical and simulation-based evaluations of IoT-F2N are carried out.

4.1 Theoretical analysis

The amount of sensor data transmission is assumed 100–400 MB. The speed of the fog device Femtolet is assumed 5 Gbps. The speed of the cloud server is assumed 8 Gbps. The delay and power consumption in the IoT-F2N is determined. The parameters used for calculating power and delay are given in Table 2.

4.1.1 Delay

If data processing occurs inside the edge device (mobile device), the delay is calculated as,

$$Del_1 = \sum_N Dl_{\mathrm{s}} + \frac{D_{\mathrm{raw}}}{R_{\mathrm{sm}}}(1 + U_{\mathrm{f1}}) + \frac{D_{\mathrm{raw}}}{S_{\mathrm{m}}} + \frac{D_{\mathrm{proc}}}{R_{\mathrm{mf}}}(1 + U_{\mathrm{f2}}) + p_{\mathrm{ab}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{fr}}}(1 + U_{\mathrm{f4}}) \tag{1}$$

where $p_{\mathrm{ab}} \leq 1$.

If data processing occurs inside the fog device (Femtolet), the delay is given as,

$$Del_2 = \sum_N Dl_{\mathrm{s}} + \frac{D_{\mathrm{raw}}}{R_{\mathrm{sm}}}(1 + U_{\mathrm{f1}}) + \frac{D_{\mathrm{raw}}}{R_{\mathrm{mf}}}(1 + U_{\mathrm{f2}}) + \frac{D_{\mathrm{raw}}}{S_{\mathrm{f}}} + p_{\mathrm{ab}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{fr}}}(1 + U_{\mathrm{f4}}) \tag{2}$$

If data processing occurs inside the local cloud, the delay is determined as,

$$\begin{aligned} Del_3 = \sum_N Dl_{\mathrm{s}} &+ \frac{D_{\mathrm{raw}}}{R_{\mathrm{sm}}}(1 + U_{\mathrm{f1}}) + \frac{D_{\mathrm{raw}}}{R_{\mathrm{mf}}}(1 + U_{\mathrm{f2}}) + \frac{D_{\mathrm{raw}}}{R_{\mathrm{fl}}}(1 + U_{\mathrm{f3}}) \\ &+ \frac{D_{\mathrm{raw}}}{S_{\mathrm{l}}} + \frac{D_{\mathrm{proc}}}{R_{\mathrm{lf}}}(1 + D_{\mathrm{f3}}) + p_{\mathrm{ab}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{lr}}}(1 + U_{\mathrm{f5}}) \end{aligned} \tag{3}$$

If data processing occurs inside the remote cloud, the delay is calculated as,

Table 2 Parameters used in power and delay calculation

Parameter	Definition
D_{raw}	Collected sensor data amount
D_{proc}	Processed data amount
P_{sen}	Power in sending data from sensor node to sensor base station
Dl_{s}	Delay in sending data from sensor node to sensor base station
N	Number of sensor nodes
R_{sm}	Rate of transmitting data from sensor base station to mobile device
R_{mf}	Rate of transmitting data from mobile device to Femtolet or femtocell
R_{fl}	Rate of transmitting data from Femtolet or cloudlet to local cloud
R_{fr}	Rate of transmitting data from Femtolet or cloudlet to remote cloud
R_{lf}	Rate of transmitting data from local cloud to Femtolet or cloudlet
R_{lr}	Rate of transmitting data from local cloud to remote cloud
R_{rf}	Rate of transmitting data from remote cloud to Femtolet or cloudlet
R_{fcl}	Rate of transmitting data from femtocell to cloudlet
P_{s}	Power in transmitting data per unit time
P_{r}	Power in receiving data per unit time
P_{p}	Power in processing data per unit time
p_{ab}	Probability of storing processed data in remote cloud
S_{m}	Data processing speed of mobile device
S_{f}	Data processing speed of Femtolet or cloudlet
S_{l}	Data processing speed of local cloud server
S_{r}	Data processing speed of remote cloud server
U_{f1}	Uplink failure rate between sensor base station and mobile device
U_{f2}	Uplink failure rate between mobile device and Femtolet or femtocell
D_{f2}	Downlink failure rate between mobile device and Femtolet or femtocell
U_{f3}	Uplink failure rate between Femtolet or cloudlet and local cloud
D_{f3}	Downlink failure rate between Femtolet or cloudlet and local cloud
U_{f4}	Uplink failure rate between Femtolet or cloudlet and remote cloud
D_{f4}	Downlink failure rate between Femtolet or cloudlet and remote cloud
U_{f5}	Uplink failure rate between local cloud and remote cloud
D_{f5}	Downlink failure rate between local cloud and remote cloud
U_{f6}	Uplink failure rate between femtocell and cloudlet
D_{f6}	Downlink failure rate between femtocell and cloudlet

$$
\begin{aligned}
Del_4 = \sum_N Dl_{\mathrm{s}} + \frac{D_{\mathrm{raw}}}{R_{\mathrm{sm}}}(1 + U_{\mathrm{f1}}) + \frac{D_{\mathrm{raw}}}{R_{\mathrm{mf}}}(1 + U_{\mathrm{f2}}) + \frac{D_{\mathrm{raw}}}{R_{\mathrm{fl}}}(1 + U_{\mathrm{f3}}) \\
+ \frac{D_{\mathrm{raw}}}{R_{\mathrm{lr}}}(1 + U_{\mathrm{f5}}) + \frac{D_{\mathrm{raw}}}{S_{\mathrm{r}}} + \frac{D_{\mathrm{proc}}}{R_{\mathrm{rf}}}(1 + D_{\mathrm{f4}})
\end{aligned}
\tag{4}
$$

If the probabilities of data processing inside the edge device, fog device, local cloud and remote cloud are α, β, γ and δ respectively, the delay is given by,

$$Del_{\mathrm{prop}} = \alpha \cdot Del_1 + \beta \cdot Del_2 + \gamma \cdot Del_3 + \delta \cdot Del_4 \tag{5}$$

where $\alpha \le 1, \beta \le 1, \gamma \le 1, \delta \le 1$ and $\alpha + \beta + \gamma + \delta = 1$.

If (femtocell + cloudlet) scenario is used in IoT, the cloudlet performs data processing and the mobile device works as an edge device. The mobile device is connected with the network through a femtocell base station, which is connected with the cloudlet. The data captured by the sensor nodes are transmitted to the mobile device. The mobile device either processes the data and sends the result to the cloudlet via the femtocell, or sends the raw data to the cloudlet via the femtocell. If the cloudlet receives processed data, it stores the data. Otherwise it processes the raw data and stores it. If the cloudlet is unable to process the data, then local or remote cloud is used. It is observed that in (femtocell + cloudlet) scenario an additional communication delay between the femtocell and cloudlet is involved. Therefore, the delay in this case is given as,

$$\begin{aligned} Del_{\mathrm{clet}} = {} & \alpha \cdot \left(Del_1 + \frac{D_{\mathrm{proc}}}{R_{\mathrm{fcl}}}(1 + U_{\mathrm{f6}})\right) + \beta \cdot \left(Del_2 + \frac{D_{\mathrm{raw}}}{R_{\mathrm{fcl}}}(1 + U_{\mathrm{f6}})\right) \\ & + \gamma \cdot \left(Del_3 + \frac{D_{\mathrm{raw}}}{R_{\mathrm{fcl}}}(1 + U_{\mathrm{f6}})\right) + \delta \cdot \left(Del_4 + \frac{D_{\mathrm{raw}}}{R_{\mathrm{fcl}}}(1 + U_{\mathrm{f6}})\right) \end{aligned} \tag{6}$$

If Eqs. (5) and (6) are compared, it is observed $Del_{\mathrm{prop}} < Del_{\mathrm{clet}}$. Hence it is observed that the use of Femtolet reduces the delay than that of using (femtocell + cloudlet) scenario.

The delay while using IoT-F2N is determined using Eq. (5). Then the results are compared with the existing IoT frameworks [22, 23]. Figure 3 compares the delay of the proposed IoT-F2N with the fog computing-based existing IoT architectures. The delay is measured in second (s).

Figure 3 shows that IoT-F2N reduces the delay by approximately 18% than the IoT using (femtocell + cloudlet) based network, 25% than the IoT using fog network [22] and 55% than the existing approach energy-management-as-a-service [23].

4.1.2 Power consumption

If data processing occurs inside the edge device, the power consumption is calculated as,

$$\begin{aligned} P_1 = {} & \sum_N P_{\mathrm{sen}} + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{sm}}}(1 + U_{\mathrm{f1}}) + P_{\mathrm{p}} \cdot \frac{D_{\mathrm{raw}}}{S_{\mathrm{m}}} \\ & + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{mf}}}(1 + U_{\mathrm{f2}}) + P_{\mathrm{s}} \cdot p_{\mathrm{ab}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{fr}}}(1 + U_{\mathrm{f4}}) \end{aligned} \tag{7}$$

If data processing occurs inside the fog device, the power consumption is given as,

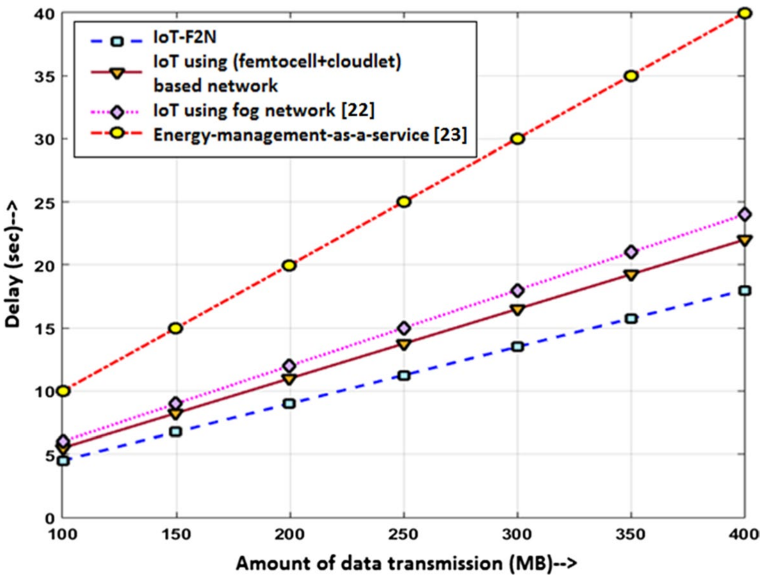


Fig. 3 Delay in proposed IoT-F2N and existing IoT architectures

$$P_2 = \sum_N P_{\mathrm{sen}} + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{sm}}}(1 + U_{\mathrm{f1}}) + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{mf}}}(1 + U_{\mathrm{f2}}) + P_{\mathrm{p}} \cdot \frac{D_{\mathrm{raw}}}{S_{\mathrm{f}}} + P_{\mathrm{s}} \cdot p_{\mathrm{ab}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{fr}}}(1 + U_{\mathrm{f4}}) \tag{8}$$

When data processing occurs inside the local cloud, the power consumption is given as,

$$P_3 = \sum_N P_{\mathrm{sen}} + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{sm}}}(1 + U_{\mathrm{f1}}) + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{mf}}}(1 + U_{\mathrm{f2}}) + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{fl}}}(1 + U_{\mathrm{f3}}) + P_{\mathrm{p}} \cdot \frac{D_{\mathrm{raw}}}{S_{\mathrm{l}}} + P_{\mathrm{r}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{lf}}}(1 + D_{\mathrm{f3}}) + P_{\mathrm{s}} \cdot p_{\mathrm{ab}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{lr}}}(1 + U_{\mathrm{f5}}) \tag{9}$$

If data processing occurs inside the remote cloud, the power consumption is given as,

$$P_4 = \sum_N P_{\mathrm{sen}} + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{sm}}}(1 + U_{\mathrm{f1}}) + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{mf}}}(1 + U_{\mathrm{f2}}) + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{fl}}}(1 + U_{\mathrm{f3}}) + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{lr}}}(1 + U_{\mathrm{f5}}) + P_{\mathrm{p}} \cdot \frac{D_{\mathrm{raw}}}{S_{\mathrm{r}}} + P_{\mathrm{r}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{rf}}}(1 + D_{\mathrm{f4}}) \tag{10}$$

If the probability of data processing inside the edge device, fog device, local cloud and remote cloud are α, β, γ and δ respectively, the power consumption is calculated as,

$$P_{\mathrm{prop}} = \alpha \cdot P_1 + \beta \cdot P_2 + \gamma \cdot P_3 + \delta \cdot P_4 \tag{11}$$

In case of (femtocell + cloudlet) scenario, the delay is calculated using Eq. (6). The power consumption in (femtocell + cloudlet) scenario is given as,

$$\begin{aligned} P_{\mathrm{clet}} = {} & \alpha \cdot (P_1 + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{proc}}}{R_{\mathrm{fcl}}}(1 + U_{\mathrm{f6}})) + \beta \cdot (P_2 + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{fcl}}}(1 + U_{\mathrm{f6}})) \\ & + \gamma \cdot (P_3 + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{fcl}}}(1 + U_{\mathrm{f6}})) + \delta \cdot (P_4 + P_{\mathrm{s}} \cdot \frac{D_{\mathrm{raw}}}{R_{\mathrm{fcl}}}(1 + U_{\mathrm{f6}})) \end{aligned} \tag{12}$$

If Eqs. (11) and (12) are compared, it is observed $P_{\mathrm{prop}} < P_{\mathrm{clet}}$. Hence it is observed that the use of Femtolet reduces the power consumption than that of using (femtocell + cloudlet) scenario.

The power consumption while using IoT-F2N is determined using Eq. (11). Figure 4 compares the power consumption of the proposed IoT-F2N with the fog computing-based existing IoT architectures. The power is measured in Watt (W). Figure 4 shows that IoT-F2N reduces the power by approximately 16% than the IoT using (femtocell + cloudlet)-based network, 43% than the IoT using fog network [22] and 53% than the existing method energy-management-as-a-service [23].

Mathematically it is already proved that proposed IoT-F2N has lower delay and power consumption than the (femtocell + cloudlet)-based network, which is

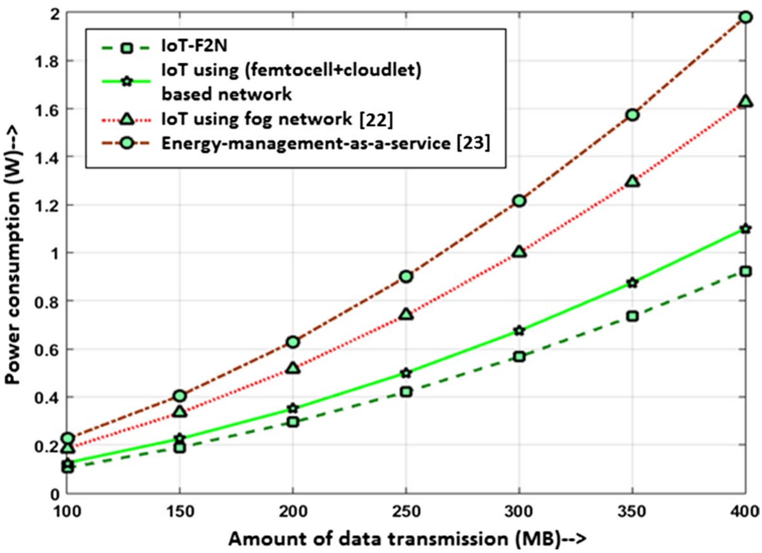


Fig. 4 Power consumption in proposed IoT-F2N and existing IoT architectures

also observed from the theoretical analysis presented in Figs. 3 and 4. In existing fog-based IoT framework [22], the sensor data are processed inside the fog devices. The cloud servers store the data. In energy-management-as-a-service [23], the sensor data are processed inside the fog devices. Different controlling units are used for managing the devices. The cloud servers store the data. But in IoT-F2N the data processing occurs inside the edge device (mobile device) or the fog device (Femtolet) in most cases. If none of them is able, then only the local or remote cloud servers process the data. Hence the storage and processing overload of the remote cloud is reduced. If the data are to be accessed by other parties, then only the data are forwarded and stored in the remote cloud servers. Thus the communication delay and power consumption are reduced than the existing frameworks.

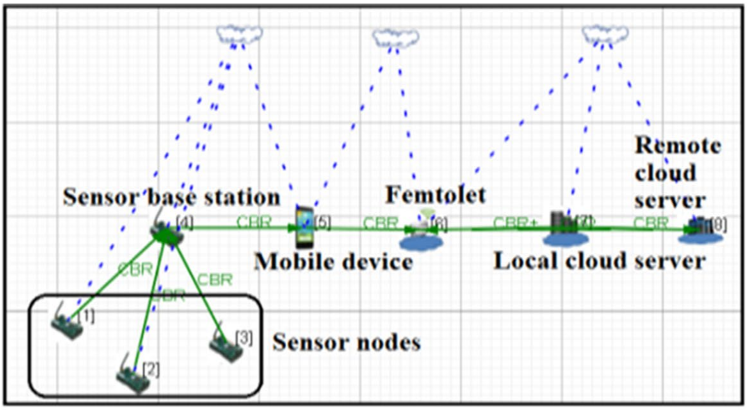
4.2 Scenario simulation using Quanlet

AIoT-F2N scenario is simulated using Qualnet 7 to evaluate its performance with respect to delay, jitter, throughput and energy consumption. Qualnet [48] is a network simulation platform where different types of network scenarios can be created, which contains routers, switches, access points, computers, radios, antennas, mobile devices, and the protocols for data transmission between the nodes of the network. The execution of the created network scenario evaluates the performance of that network in terms of delay, jitter, energy consumption, throughput, etc.

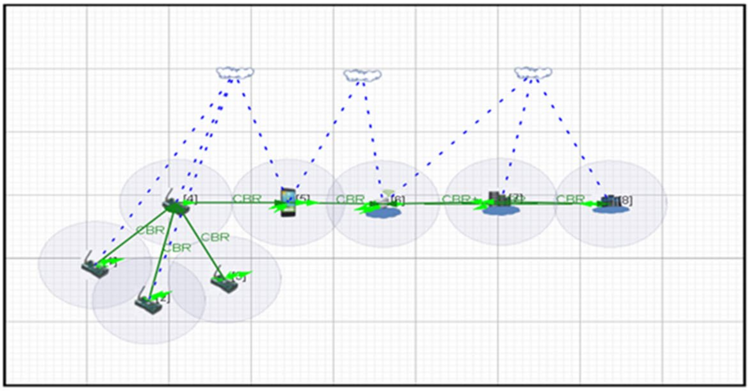
The parameters of simulation are given in Table 3 and the scenario simulated is given in Fig. 5. In the created scenario, nodes 1, 2 and 3 are sensor nodes, node 4 is sensor base station, node 5 is the mobile device (edge device), node 6 is Femtolet (fog device), and nodes 7 and 8 are the local and remote cloud respectively.

Table 3 Simulation parameters

Layer	Parameter	Value
Physical layer	Radio type	802.15.4 radio, 802.11b radio
	Energy model	Mica-motes
	Antenna model	Omni directional
	Packet reception model	PHY 802.15.4 reception model, PHY 802.11b reception model
MAC layer	MAC protocol	802.15.4, 802.11
Network layer	Routing protocol	AODV
	Network protocol	IPV4
Transport layer	Maximum segment size	512 bytes
CBR properties	Item size	256–1024 bytes
Scenario properties	Simulation time	300 s



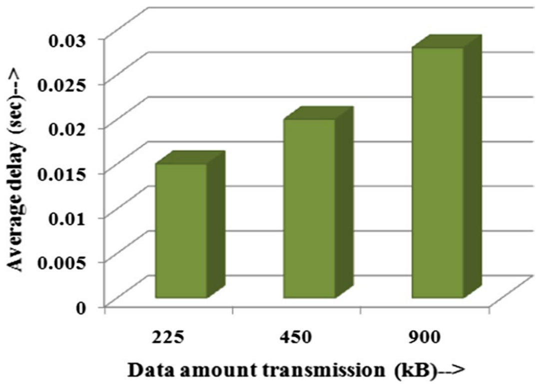
(a) Simulation scenario of IoT-F2N before experiment



(b) Simulation scenario of IoT-F2N during experiment

Fig. 5 Simulated model of IoT-F2N

Fig. 6 Average delay in IoT-F2N



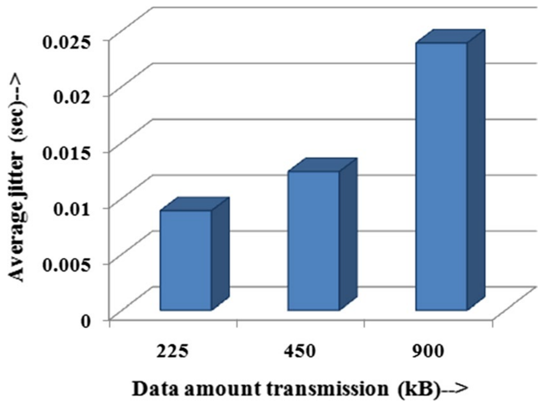


Fig. 7 Average jitter in IoT-F2N

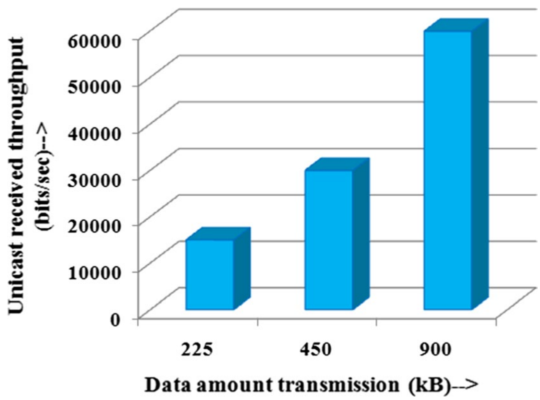


Fig. 8 Unicast received throughput in IoT-F2N

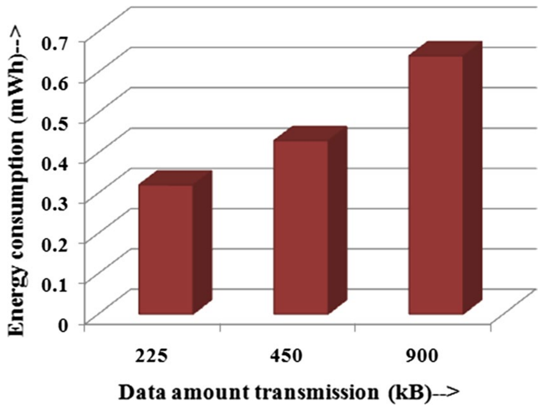
Average delay Delay refers to the time period of data transmission from the sender to the receiver. Figure 6 shows that the average delay in proposed IoT-F2N paradigm is 0.01–0.03 s approximately for 225–900 kB data transmission between the nodes.

Average jitter Jitter refers to the difference between the delay of the arrival of the present packet and that of the previous packet. Figure 7 shows that the average jitter in proposed IoT-F2N paradigm is 0.005–0.025 s approximately for 225–900 kB data transmission between the nodes.

Unicast received throughput The successful message delivery rate is referred as the throughput. Figure 8 shows that the unicast received throughput in proposed IoT-F2N paradigm is 10,000–60,000 bits/s approximately for 225–900 kB data transmission between the nodes.

Energy consumption The total energy consumption in transmitting and receiving data is determined. Figure 9 shows that the energy consumption in proposed

Fig. 9 Energy consumption in IoT-F2N



IoT-F2N paradigm is 0.2–0.6 mW h approximately for 225–900 kB data transmission between the nodes.

5 Conclusions and future scope

This paper has proposed an IoT paradigm using Femtolet-based fog network. The proposed paradigm is referred as IoT-F2N. In IoT-F2N the sensor nodes collect sensor data and send to the mobile device that works as edge device. The edge devices are registered under the fog device Femtolet. Femtolet is an indoor base station with data storage and processing ability. The mobile device sends the sensor data to the Femtolet. The processed sensor data are maintained inside the fog device Femtolet. If the data are to be accessed by some other parties then only Femtolet sends the data to the remote cloud servers. If the Femtolet is incapable to process, then either the local or the remote cloud servers process the sensor data. Theoretical analysis shows that IoT-F2N reduces the delay and power by approximately 18% and 16%, respectively, than the IoT using femtocell plus cloudlet-based network. The theoretical analysis also illustrates that IoT-F2N reduces the delay and power by approximately 25% and 43%, respectively, than the existing fog-based IoT model. Thus the proposed IoT-F2N is referred as an energy-efficient architectural model.

For faster response to the user request in IoT-F2N, multilevel data processing will be required. In that case some processing will be performed inside the Femtolet, and rest will be performed inside the local or remote cloud server. In such circumstances, efficient multilevel sensor data processing algorithm as well as efficient resource allocation mechanism will be required. Utility computing aggregates server, network, and storage systems into a single and centrally managed pool of resources. As IoT, Femtolet and fog computing are integrated in IoT-F2N, providing a secured and trustworthy IoT-F2N is also an emerging research field.

Acknowledgements Authors are grateful to Department of Science and Technology (DST) for sanctioning a research Project entitled “Dynamic Optimization of Green Mobile Networks:

Algorithm, Architecture and Applications" under Fast Track Young Scientist Scheme Reference No. SERB/F/5044/2012-2013, DST-FIST for SR/FST/ETI-296/2011 and Melbourne-Chindia Cloud Computing (MC3) Research Network.

References

1. Gubbi J, Buyya R, Marusic S, Palaniswami M (2013) Internet of Things (IoT): a vision, architectural elements, and future directions. Future Gener Comput Syst 29(7):1645–1660
2. Mukherjee A, De D (2014) Femtocell based green health monitoring strategy. In: XXXIth URSI General Assembly and Scientific Symposium (URSI GASS). IEEE, pp 1–4
3. De D, Mukherjee A, Ray A, Roy DG, Mukherjee S (2016) Architecture of green sensor mobile cloud computing. IET Wirel Sens Syst 6(4):109–120
4. Chandrasekhar V, Andrews JG, Muharemovic T, Shen Z, Gatherer A (2009) Power control in two-tier femtocell networks. IEEE Trans Wirel Commun 8(8):4316–4328
5. Mukherjee A, Bhattacherjee S, Pal S, De D (2013) Femtocell based green power consumption methods for mobile network. Comput Netw 57(1):162–178
6. Mukherjee A, De D, Roy DG (2019) A power and latency aware cloudlet selection strategy for multi-cloudlet environment. IEEE Trans Cloud Comput 7(1):141–154
7. Mahmud R, Kotagiri R, Buyya R (2018) Fog computing: a taxonomy, survey and future directions. In: Yang LT (ed) Internet of everything. Springer, Singapore, pp 103–130
8. Mukherjee A, De D (2016) Femtolet: a novel fifth generation network device for green mobile cloud computing. Simul Model Pract Theory 62:68–87
9. Barbarossa S, Sardellitti S, Di Lorenzo P (2013) Joint allocation of computation and communication resources in multiuser mobile cloud computing. In: 14th Workshop on Signal Processing Advances in Wireless Communications (SPAWC). IEEE, pp 26–30
10. Mukherjee A, Deb P, De D, Buyya R (2018) C2OF2N: a low power cooperative code offloading method for femtolet-based fog network. J Supercomput 74(6):2412–2448
11. Deb P, Mukherjee A, De D (2019) Design of green smart room using fifth generation network device femtolet. Wirel Pers Commun 104(3):1037–1064
12. De D, Mukherjee A (2015) Femto-cloud based secure and economic distributed diagnosis and home health care system. J Med Imaging Health Inform 5(3):435–447
13. De D, Mukherjee A, Sau A, Bhakta I (2016) Design of smart neonatal health monitoring system using SMCC. Healthc Technol Lett 4(1):13–19
14. Ahmad M, Amin MB, Hussain S, Kang BH, Cheong T, Lee S (2016) Health fog: a novel framework for health and wellness applications. J Supercomput 72(10):3677–3695
15. Sekhar PK, Wignes F (2016) Trace detection of research department explosive (RDX) using electrochemical gas sensor. Sens Actuators B Chem 227:185–190
16. Rawat P, Singh KD, Chaouchi H, Bonnin JM (2014) Wireless sensor networks: a survey on recent developments and potential synergies. J Supercomput 68(1):1–48
17. Giammarini M, Isidori D, Pieralisi M, Cristalli C, Fioravanti M, Concettoni E (2016) Design of a low cost and high performance wireless sensor network for structural health monitoring. Microsyst Technol 22(7):1845–1853
18. Huh J-H, Kim T-J (2019) A location-based mobile health care facility search system for senior citizens. J Supercomput 75(4):1831–1848
19. Durao F, Carvalho JFS, Fonseka A, Garcia VC (2014) A systematic review on cloud computing. J Supercomput 68(3):1321–1346
20. Abdelmaboud A, Jawawi DNA, Ghani I, Elsafi A, Kitchenham B (2015) Quality of service approaches in cloud computing: a systematic mapping study. J Syst Softw 101:159–179
21. Madria S, Kumar V, Dalvi R (2014) Sensor cloud: a cloud of virtual sensors. IEEE Softw 31(2):70–77
22. Sarkar S, Misra S (2016) Theoretical modelling of fog computing: a green computing paradigm to support IoT applications. IET Netw 5(2):23–29
23. Faruque A, Abdullah M, Vatanparvar K (2016) Energy management-as-a-service over fog computing platform. IEEE Internet Things J 3(2):161–169

24. Sarkar S, Chatterjee S, Misra S (2018) Assessment of the suitability of fog computing in the context of Internet of Things. IEEE Trans Cloud Comput 6(1):46–59
25. Zhang H, Xiao Y, Bu S, Niyato D, Yu FR, Han Z (2017) Computing resource allocation in three-tier IoT fog networks: a joint optimization approach combining Stackelberg game and matching. IEEE Internet Things J 4(5):1204–1215
26. Satyanarayanan M, Bahl P, Caceres R, Davies N (2009) The case for VM-based cloudlets in mobile computing. IEEE Pervasive Comput 4:14–23
27. Roy DG, De D, Mukherjee A, Buyya R (2017) Application-aware cloudlet selection for computation offloading in multi-cloudlet environment. J Supercomput 73(4):1672–1690
28. Flores H, NarayanaSrirama S (2014) Mobile cloud middleware. J Syst Softw 92:82–94
29. Brogi A, Forti S (2017) QoS-aware deployment of IoT applications through the fog. IEEE Internet Things J 4(5):1185–1192
30. White G, Nallur V, Clarke S (2017) Quality of service approaches in IoT: a systematic mapping. J Syst Softw 132:186–203
31. Zhang P, Zhou M, Fortino G (2018) Security and trust issues in Fog computing: a survey. Future Gener Comput Syst 88:16–27
32. Dsouza C, Ahn G-J, Taguinod M (2014) Policy-driven security management for fog computing: preliminary framework and a case study. In: Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration (IEEE IRI 2014). IEEE, pp 16–23
33. Kulkarni S, Saha S, Hockenbury R (2014) Preserving privacy in sensor-fog networks. In: 9th International Conference for Internet Technology and Secured Transactions (ICITST-2014). IEEE, pp 96–99
34. Khan MA, Salah K (2018) IoT security: review, blockchain solutions, and open challenges. Future Gener Comput Syst 82:395–411
35. Reyna A, Martín C, Chen J, Soler E, Díaz M (2018) On blockchain and its integration with IoT. Challenges and opportunities. Future Gener Comput Syst 88:173–190
36. Mukherjee B, Wang S, Lu W, Neupane RL, Dunn D, Ren Y, Su Q, Calyam P (2018) Flexible IoT security middleware for end-to-end cloud–fog communication. Future Gener Comput Syst 87:688–703
37. Aazam M, Zeadally S, Harras KA (2018) Offloading in fog computing for IoT: review, enabling technologies, and research opportunities. Future Gener Comput Syst 87:278–289
38. Dehury CK, Sahoo PK (2016) Design and implementation of a novel service management framework for IoT devices in cloud. J Syst Softw 119:149–161
39. Srinivasan CR, Rajesh B, Saikalyan P, Premsagar K, Yadav ES (2019) A review on the different types of Internet of Things (IoT). J Adv Res Dyn Control Syst 11(1):154–158
40. Hamdan O, Shanableh H, Zaki I, Al-Ali AR, Shanableh T (2019) IoT-based interactive dual mode smart home automation. In: International Conference on Consumer Electronics (ICCE). IEEE, pp 1–2
41. Adeogun RO, Rodriguez I, Razzaghpour M, Berardinelli G, Christensen PH, Mogensen P (2019) Indoor occupancy detection and estimation using machine learning and measurements from an IoT LoRa-based monitoring system. In: 2019 Global Iot Summit (giots)
42. Shridhar VS (2019) The India of Things: Tata communications' countrywide IoT network aims to improve traffic, manufacturing, and health care. IEEE Spectr 56(2):42–47
43. Al-Turjman F (2019) 5G-enabled devices and smart-spaces in social-IoT: an overview. Future Gener Comput Syst 92:732–744
44. Mutlag AR, Ghani MKA, Arunkumar N, Mohamed MA, Mohd O (2019) Enabling technologies for fog computing in healthcare IoT systems. Future Gener Comput Syst 90:62–78
45. Adhikari M, Gainey H (2019) Energy efficient offloading strategy in fog-cloud environment for IoT applications. Internet Things 6:100053
46. Mukherjee A, De D (2018) Octopus algorithm for wireless personal communications. Wirel Pers Commun 101(1):531–565
47. Mukherjee A, De D, Buyya R (2019) E2R-F2N: energy-efficient retailing using a femtolet-based fog network. Softw Prac Exp 49(3):498–523
48. https://www.scalable-networks.com/qualnet-network-simulation