Assessment of the Suitability of Fog Computing in the Context of Internet of Things

Subhadeep Sarkar†, *Student Member*, *IEEE*, Subarna Chatterjee*, *Student Member*, *IEEE*, Sudip Misra‡, *Senior Member*, *IEEE*

†*‡School of Information Technology,
†School of Medical Science and Technology
Indian Institute of Technology, Kharagpur, 721302, India
Email: †subhadeep@smst.iitkgp.ernet.in, *subarna@sit.iitkgp.ernet.in, ‡smisra@sit.iitkgp.ernet.in

***Abstract*—This work focuses on analysis of the fog computing paradigm and its suitability to support the Internet of Things (IoT) – one of the first attempts of its kind. With the rapid increase in the number of Internet-connected devices, the increased demand of real-time, low-latency services is proving to be challenging for the traditional cloud computing framework. Also, our irreplaceable dependency on cloud computing demands the cloud data centers (DCs) always to be up and running which exhausts huge amount of power and yield tons of carbon dioxide (CO_2) gas. In this work, we assess the applicability of the newly proposed *fog computing* paradigm to serve the demands of the latency-sensitive applications in the context of IoT. We model the fog computing paradigm by mathematically characterizing the fog computing network in terms of power consumption, service latency, CO_2 emission, and cost, and evaluating its performance for an environment with high number of Internet-connected devices demanding real-time service. A case study is performed with traffic generated from the 100 highest populated cities being served by eight geographically distributed DCs. Results show that as the number of applications demanding real-time service increases, the fog computing paradigm outperforms traditional cloud computing. For an environment with 50% applications requesting for instantaneous, real-time services, the overall service latency for fog computing is noted to decrease by 50.09%. However, it is mentionworthy that for an environment with less percentage ($\leq 5\%$) of applications demanding for low-latency services, fog computing is observed to be an overhead compared to the traditional cloud computing. Therefore, the work shows that in the context of IoT, with high number of latency-sensitive applications demanding services fog computing outperforms cloud computing, however, it is not a replacement of the same.**

***Index Terms*—Fog computing, Cloud computing, Internet of things (IoT), Service latency, Power consumption, Carbon-dioxide emission**

I. INTRODUCTION

Recent advancements in computer technologies have led to the conceptualization, development, and implementation of cloud computing systems. From its inception, cloud computing has gained widespread popularity due to its applicability in diverse, widespread domains. Cloud computing systems are generally based on data centric networks (DCNs), which are treated as the sole, monopolized hubs responsible for computation and storage. For contemporary cloud-based systems, all service requests and resource demands are analyzed and processed within the data centers (DCs). However, with the steep rise in the number of Internet-connected devices and in the light of the emerging technology of the Internet of things (IoT), the amount of data to be handled by the cloud DCs is paramount. In 2012, global commercialization of IoT-based application systems generated a revenue of \$4.8 trillon [1]. It is statistically estimated that by 2015, around 25 billion autonomous devices will be connected to the Internet. Cisco estimates that due to IoT, the global corporate profits will also increase approximately by 21% [2]. Also, the cloud DCs exhaust massive amount of energy leading to the emission of enormous amount of greenhouse gases (GHGs), especially carbon dioxide (CO_2). This takes a deep toll on the environment.

The technology of IoT is reliant on cloud computing. Also, contemporary data traffic is heavily heterogeneous with huge volume and is generated at high velocity giving rise to "Big" data. Clearly, data from the billions of Internet-connected devices will be voluminous and demands to be processed within the cloud DCs. Most of these IoT infrastructures, such as smart vehicular traffic management systems, smart driving and car parking systems, and smart grids are observed to demand real-time, low-latency services from the service providers. Since conventional cloud computing involves processing, computation, and storage of the data only within DCs, the massive data traffic generated from the IoT devices is anticipated to experience a huge network bottleneck and, in turn, high service latency and poor Quality of Service (QoS). Moreover, in order to process and serve this high number of requests the DCs are required to be up and running around the clock which results in the consumption of enormous amount of energy and massive emission of CO_2. With the DCs made to run both during on-peak and off-peak hours, the power consumption is increasing sharply putting the condition of the environment at stake.

In this work, we analyze the suitability of a recent computing paradigm – *fog computing* to serve the demands of the real-time, latency-sensitive applications in the context of IoT. Fog (From cOre to edGe) computing, a term coined by Cisco in 2012 [3], is a distributed computing paradigm, that empowers the network devices at different hierarchical levels with various degrees of computational and storage capability. These devices are equipped with an 'intelligence' which allows them to examine whether an application request requires the intervention of the cloud computing tier or not. The idea is to serve the requests which demand real-time, low-latency services (e.g. live streaming, smart traffic monitoring, smart parking etc.) by the fog computing devices and the connected work stations and small-scale storage units. However, the requests which demand semi-permanent and permanent storage or require extensive analysis involving historical data-sets (e.g. social media data, photos, videos, medical history, data backups etc.), these devices only act as routers or gateways to redirect the requests to the core cloud computing framework. Therefore, the focus of this work is to assess the suitability of fog computing in the context of real-time services and application requests, and compare its performance against the traditional cloud computing framework under different traffic types.

It should be clearly mentioned, that fog computing is not a replacement of cloud computing, rather these two technologies complement one another. The complementary functions of cloud and fog enable the users to experience a new breed of computing technology that serves the requirements of the real-time, low-latency IoT applications running at the network edge, and also supports complex analysis and long-term storage of data at the core cloud computing module. The work mathematically characterizing fog computing in terms of power consumption, service latency, and we investigate the eco-friendliness of the technology as well.

A. Motivation

The primary storage and computing centers of the cloud computing architecture are the geographically scattered DCs which communicate among themselves through the DCNs. These DCNs are huge consumers of energy and, in turn, generate and emit heavy high amount of CO_2 gas. Motivated by the design of the mathematical model for cloud networks by Zhang *et al.* [4], in this work, we develop the network model for the fog paradigm and assess its performance while supporting the IoT. For the study of the performance of cloud and fog systems, we refer the above work and our work, respectively. With the increase in the number of the IoT devices demanding real-time services from the service providers, tradition cloud computing framework is expected to face the following challenges:

(i) The International Data Corporation (IDC) forecast says that the worldwide market for IoT-based technologies and solutions will grow from \$1.9 trillion in 2013 to \$7.1 trillion in 2020 [5]. With this increase in the number IoT devices, the DCNs encounters a heavy network traffic which affects the service latency by a great extent, and consequently, applications requesting for real-time services would experience a deterioration in the QoS.

(ii) The U.S. Environmental Protection Agency (EPA) report [6] stated that in the year of 2006, the DCs of U.S. consumed about 61 billion kilowatt-hours of power with a total financial expenditure worth \$4.5 billion. It was also observed that in 2007, 30 million worldwide servers were accounted for 100 TWh of the world's energy consumption at a cost of \$9 billion which is expected to rise upto 200 TWh in the next few years [6], [7]. Therefore, it is important to exempt the cloud DCs from being bombarded with service requests, and serve a part of those requests from the network edge. This would relax the load experienced by the DCs and also would serve the latency-sensitive apllication requests in a better way with increased QoS.

(iii) Moreover, with the cloud DCs made to run during around the clock irrespective of the traffic rate, the amount of GHGs emitted remains unreasonably high. Presently, the planet's annual electricity consumption Information-Communications-Technologies (ICT) ecosystem is about $1,500$ TWh – equal to the total electricity produced by Japan and Germany [8]. Controlling the traffic which requires to be directed to the core cloud computing module would thereby reduce the effective up-time of the DCs. This would help in keeping down the amount of GHGs generated by the DCs and help system to be more eco-friendly.

B. Contribution

We discuss the contributions of our work in this subsection. As mentioned earlier, fog computing is not a substitution of cloud computing; rather in this work, we analyze the suitability of fog computing combined with the traditional cloud computing in supporting the ever-increasing demands of the latency-hungry IoT-based applications. The primary contributions of this work are listed below.

(i) Initially, this work constructs the network model of fog computing – one of the first attempts of its kind in this direction. We define the different network devices and networking links within the fog computing architecture and explain the traffic exchange pattern for the same.

(ii) Based on this model, the work mathematically characterizes the performance metrics of fog computing in terms of the service latency, power consumption, CO_2 emission for different renewable and non-renewable energy resources, and the corresponding costs incurred.

(iii) The work also performs a fair and equitable comparative study for both cloud and fog computing systems. We analyze the suitability of the fog computing architecture to support the demands of IoT devices and while serving latency-sensitive applications.

C. Paper Organization

The rest of the paper is organized as follows. Section II describes the work done so far on this domain. In Section III, the detailed architecture of the of the fog computing paradigm is presented. We discuss the details of the fog networking model in Section IV. The performance metrics are presented and modeled in Section V. Section VI presents experimental setup for the case study. In Section VII, the performance evaluation of fog computing paradigm is performed and a comparative study of both fog and cloud models is presented. Finally, the work is concluded in Section VIII.

II. Related Work

In this Section, we present and discuss the prior research works which were done in this domain.

A. Internet of Things

Recent research has spawned the concept of IoT [9], [10] that connects billions of things across the globe to the Internet and enables Machine to Machine (M2M) communication [11] among these devices. Thus, IoT framework is a dynamic and persuasive platform for data storage, computation, and management [12]. Contemporary devices or Internet based systems are gradually converging towards IoT [13]. It is estimated and reported by Cisco that, by 2020, around 50 billion devices will be connected to the Internet [3]. Thus, by 2020, it is estimated that a large number of applications will be required to be processed and served through the technology of IoT [14], [15]. Analyzing contemporary data trends of large volume, heavy heterogeneity, and high velocity ('Big' data), it is also anticipated that a vast majority of these applications are highly latency-sensitive and require real-time processing [16]–[18]. Therefore, to provision the resource management and heavy computational potential to the applications, IoT leans highly on cloud computing [19]–[22]. Consequently, the performance of IoT is profoundly dependent on the ability of cloud platforms to serve billions of devices and their applications, in real-time [12], [23].

B. Cloud computing

Over the last few years, cloud computing has been extensively studied and excavated. Cloud computing follows a service oriented architecture and thrives on the principle of virtualization. A good number of works [24]–[28] on cloud computing illustrates the detailed underlying process behind the provisioning of cloud services. The process of complete virtualization of cloud services involves several cloud service providers (CSPs), dispersed across multiple geographical locations, to provision the physical resources as per the user demand. Moreover, cloud systems are based on DCNs. Thus, for every user demand, services from one or more DCNs are supplied in the form of storage or processing. In [29], Xiao *et al.* addressed the problem of design and optimal positioning of DCs to improve the QoS in terms of service latency and cost efficiency. However, the work is strongly affected by the efficiency of the DCNs. In [30], Zheng *et al.* proposed a QoS ranking framework to mitigate the service latency in scenarios of practical service invocations. In another work, Chen *et al.* [31] focused on the problem of latency for video streaming services. The work suggests the usage of a single DC under a single CSP. However, the situation might be hypothetical as in real-life scenarios of IoT, a single DC under a single global CSP may hinder the overall service efficiency due to lack of proper management and shortage of cloud storage. Few works [31], [32] have focused on the problem of enhancing the system performance by optimizing the placement of virtual machines. However, the scope of the work is limited to the QoS of the cloud computing units only and not on the service that is eventually provisioned to the end-users. Tziritas *et al.* [33] addressed process migration to improve the performance of cloud systems and demonstrated experimental results with 1000 processes. However, IoT concerns billions of processes and in such a scenario, process migration within DCs might be of overhead degrading the performance. Similarly, job scheduling techniques to improve QoS were also compared by Chandio *et al.* [34] using 22385 jobs. However, compared to IoT systems, the count is too low less to be considered. Other scheduling techniques that focus on real-time workload scheduling [35] or energy-efficient scheduling [36] have also worked with low scale scenarios comprising of a maximum of 128 Virtual Machine instances and 1000 cloud servers, respectively.

For each of the above works, the DCs form the hub of computing and the DCNs are striked for every single application request. Therefore, with the increase in the number of IoT consumers and with every requests being required to be processed within the DCs, it is likely that the cloud DCNs will encounter a serious difficulty in serving the IoT applications real-time. Additionally, with the increase in the number of latency-sensitive applications, the efficiency of service provisioning will also reduce to a significant extent. Additionally, considering the contemporary state of our environment, it is observed and reported [37]–[41] that as we are more and more advancing towards technology, we are driving our nature to an alarming state. Therefore, it is imperative to simultaneously maintain the eco-friendliness of our surroundings.

C. Fog computing

The contemporary trends in data volume, velocity, and variety and the limitations of cloud computing make it easy to speculate the need to propose new techniques of data management and administration. In this context, Cisco proposed the very recent and revolutionary concept of *fog* computing [3]. Fog computing is defined as a distributed computing infrastructure that is able to handle billions of Internet connected devices. The underlying principle of the technology is "edge computing" in which the services are hosted within the edge devices inclusive of the gateways, routers, and access points. A similar concept of edge computing was also proposed by Lewis in [42]. In [43], Bonomi *et al.* defined the characteristics of the paradigm in terms of latency, location awareness, geographical distribution, mobility, heterogeneity, and the predominant access to wireless devices. However, it was more concerned on the theory and dogma of the computing paradigm. Hong *et al.* [44] designed a programming model to support large-scale IoT applications through mobile fog computing. The model supports the service provisioning to geographically scattered, latency-sensitive applications. The credibility and prospect of fog computing is enhanced by few more works.

In [45], the authors considered various computing paradigms inclusive of cloud computing. The authors inferred the feasibility of building up a reliable and fault-tolerant fog computing platform. Zhu *et al.* [46] focused on optimization of web service through fog computing. The authors examined and inferred the dynamic adaptability of the paradigm. Another work [47] addressed the problem of mitigating the security attacks in cloud-based systems through a fog computing environment. Fog computing was used to launch disinformation attacks and it was observed that the several malicious attacks could be averted. Of late, Zao *et al.* [48] advanced the research to a great extent by exploring the dimension of augmented brain computer interaction. However, most of the works on fog computing have primarily focused on the principles, basic notions, and the doctrines of it. Not many works have contributed in the technical aspect of the paradigm in terms of an implementation perspective.

D. Focus of the paper

This work studies the suitability and applicability of fog computing as a potential platform to support IoT. The novelty of the paper is to model the paradigm of fog computing and perform a comparative study in terms of power consumption, cost and latency with respect to cloud systems. The work further investigates the eco-friendly aspects of the paradigm to judge its appropriateness to serve the world of Internet connected devices.

III. Fog Computing Architecture

In this Section, we present the fog computing architecture and its minute details. It is imperative to mention that fog computing is a non-trivial extension of cloud computing and is based on the principles of edge computing.

A. Assumptions

Fog paradigm is still in its early stage of research and is yet to shape up. We therefore, draw few simplified, yet realistic assumptions.

- Every terminal node (TN) is aware of its absolute geo-spatial position and is able to share the information through technologies such as GPS, GIS, or GNSS.
- The devices of the fog computing tier of are "intelligent" in terms of their computational and storage ability. The devices are also equipped with routing or packet forwarding ability [44].
- The networking devices of the fog computing layer (edge/fog/cloud gateways), as shown in Fig. 2, are self-adjusting to dynamic load sharing in terms of the network, computational, and storage load among themselves.
- Every fog computing devices can support the mobility of the TNs.

B. System Outline

This subsection illustrates the distinct tiers of a generic fog computing architecture. As depicted in Fig. 1, it is essentially a three tier architecture. The tiers are discussed below.

(a) *Tier 1*: This is the bottom-most tier of the architecture. The tier comprises of several TNs. The TNs are majorly smart, wireless sensor nodes that sense heterogeneous physical parameters and transmit the same to the immediate upper tier.

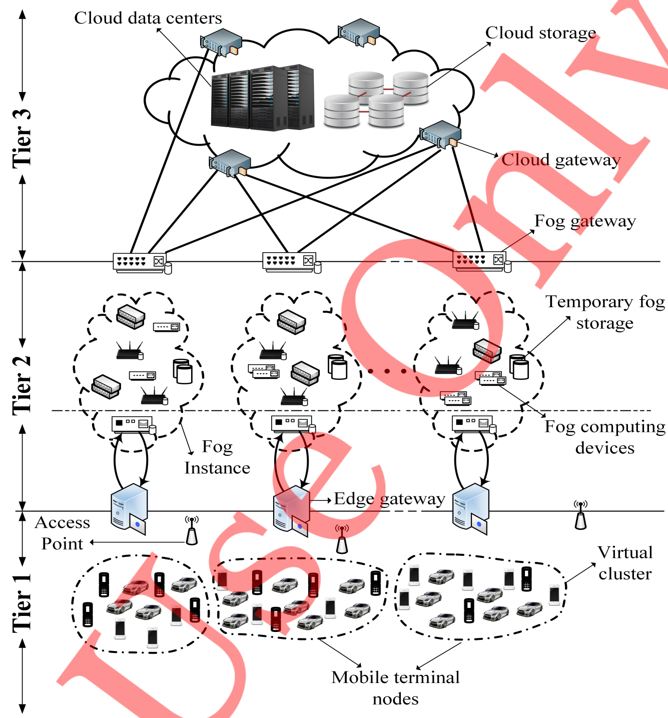


Fig. 1: Fog computing architecture

(b) *Tier 2*: The tier 2 or the middle layer is also known as the fog computing layer. The primary components of this tier are intelligent intermediate devices (such as routers, gateways, switches, and access points) that possess the ability of data storage, computation, routing, and packet forwarding.

(c) *Tier 3*: The uppermost tier is commonly known as the cloud computing tier. Several high-end servers and DCs comprise this tier.

C. Architecture Details

The principle of fog computing architecture is based on edge computing. As already discussed, the bottom-most tier comprising of smart Internet connected TNs are one of the fundamental components of IoT. While the data is transmitted upwards (towards the fog tier) the data is processed within every gateways and intermediate fog devices. The TNs are assumed to form location-based logical clusters, which are termed as Virtual Clusters (VCs). Every VC together form an Edge Virtual private Network (EVPN) that transmits data to multiple Fog Instances (FIs). An FI is conceptualized specific to a geographic location. The mobility of a TN makes the mapping of a TN to an FI flexible and non-static. Within the FIs, the data are processed and analyzed to decide whether it needs to be transmitted to the cloud DCs. Application requests which require storage or historical data based analytics are redirected to the cloud, else, the data are processed within the fog units. The fog devices possess limited semi-permanent storage that allow temporary data storage and serve the latency-sensitive applications in real-time.

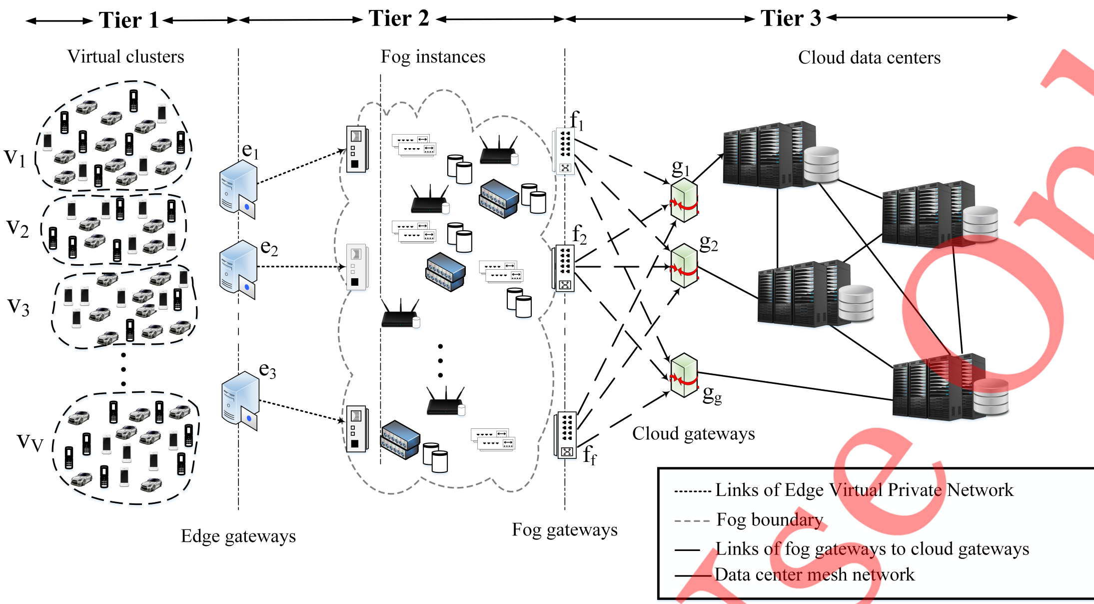


Fig. 2: Networking links and components of fog computing

The cloud computing tier is commonly responsible for permanent storage of huge, voluminous data chunks within its powerful DCs. The DCs are equipped with massive computational ability. However, unlike conventional cloud architecture, the core cloud DCs are not bombarded for every single query. Fog computing enables the cloud tier to be accessed and utilized in an efficient and controlled manner.

IV. Networking Model

Motivated by the works of Zhang *et al.* [4], we develop the network model of the fog computing paradigm. Analogous to the above work, the proposed work designs the metric of fog network model. Any cloud based analytics are excerpted from the above cited work, especially while performing a comparative study of cloud versus fog in Section VII. However, all fog based evaluations are based on the metrics designed in this work.

In this Section, we define the functional components and the data transfer links associated with the fog computing paradigm. The set of N geo-spatially distributed mobile TNs is denoted by $\mathcal{N}$, where $|\mathcal{N}| = N$. At time instant t, we consider these N TNs to be grouped under different VCs. The set of all VCs is denoted by $\mathcal{V}$, such that $\mathcal{V} = \{v_i | i \in [1, V], i \in \mathbb{I}\}$ and $|\mathcal{V}| = V$, where V is the total number of VCs present in the system and v_i denotes the i^{th} VC. The total number of TNs mapped to v_i is denoted by n_i, $\forall i = 1(1)V$. As total coverage of the TNs by the VCs is assumed for any instant of time, we have:

$$\sum_{i=1}^{V} |v_i| = \sum_{i=1}^{V} n_i = N. \tag{1}$$

The data generated by all the constituent TNs within a VC are transported through the edge gateways towards the fog computing tier. The set of all edge gateways present at the lowest tier in the computing architecture is denoted by $\mathcal{E}$, with $|\mathcal{E}| = E$. The data transport links between the VCs and the edge gateways are considered to be of high bandwidth. However, the data transmission bandwidth between the EVPNs and the FIs are considered to be restricted. The set of all fog instances present in the system at any given time is given by $\mathcal{F}$, with $|\mathcal{F}| = F$. All data and queries generated from the applications instances running with the TNs in tier 1 are forwarded through the (e, f) link, i.e., through the link between the EVPN e and the FI f, $\forall e \in \mathcal{E}$ and $\forall f \in \mathcal{F}$.

Let $P_r^{v_i}(t)$ and $P_s^{v_i}(t)$ be the total amount of data, in bytes, that are generated from the VC v_i in time-slot t which demand to be served and stored, respectively. Note that, based on the type of request a stream of bytes are forwarded to the cloud computing tier. If a request demands real-time services, it is processed and served from within the FI without the intervention of the cloud computing core. However, the requests which require intervention of the cloud computing layer for analysis based on historical data-sets and for long-term (semi-permanent or permanent) storage, are redirected to the upper tier after each time-slot. Let $Q_r^{v_i}(t)$ and $Q_s^{v_i}(t)$ be the total number of bytes generated from the VS v_i which is redirected to the cloud computing layer for computation and storage purposes, respectively, in the t^{th} time-slot. Clearly, $P_r^{v_i}(t) \leq Q_r^{v_i}(t)$ and $P_s^{v_i}(t) \leq Q_s^{v_i}(t)$, $\forall i = 1(1)V$.

The fog gateways, located between the fog computing tier and the cloud computing tier, are represented by the set $\mathcal{G}$, where $|\mathcal{G}| = G$. Also, $\forall f \in \mathcal{F}$ and $\forall g \in \mathcal{G}$, the data communication link (f, g), between the FI f and the fog gateway g is also bandwidth constrained. Finally, we discuss the data communication and aggregation involving the cloud computing framework. The byte-stream transmitted by the fog gateways reaches a cloud DC through a channel of limited bandwidth. The set of all DCs at the cloud-end is represented by $\mathcal{K}$, where $|\mathcal{K}| = K$. $\forall g \in \mathcal{G}$ and $\forall c \in \mathcal{K}$, (g, c) the communication link between the cloud gateway g and the DC c is denoted by (g, c) which is the route through which data reach the cloud computing tier from the fog computing tier. However, we assume that at the cloud-end not all DCs are responsible data aggregation at every instant of time. We consider that at any given time instant, data aggregation and subsequent analysis take place only in one DC. Data migration between the cloud DCs takes place through high bandwidth links.

V. Modeling the Performance Metrics

We define the fog tier routing variable (FTRV) as $X^{fog}_{v_i,e,f}(t)$, $\forall v_i \in \mathcal{V}, \forall e \in \mathcal{E}, \forall f \in \mathcal{F}$, which indicates the uplink route through which the data generated from the VC v_i in time-slot t reaches the FI f for real-time processing and temporal storage. However, for application instances which require to be referred to the cloud framework for aggregation, historical analysis, and long-term storage, the cloud tier routing variable (CTRV) is defined as $X^{cld}_{f,g,c,d}(t)$, $\forall f \in \mathcal{F}, \forall g \in \mathcal{G}, \forall c, d \in \mathcal{D}$, which denotes the route through which data from the FI f reaches its destination DC in time-slot t. For the FTRV, the route $v_i \to e \to f$ represents the path along which the data originated from v_i moves to the FI f through the intermediate edge gateway e. Similarly, for the CTRV, $f \to g \to c \to d$ indicates that the data-stream is redirected from the FI f to the cloud DC c through the intermediate fog gateway g, and from there it is once again transferred to the DC d for aggregation and further processing.

For any given route, its corresponding FTRV or CTRV value is set as equal to the proportion of the total data (in bytes) generated in time-slot t which traverse through the route. Clearly, if a given route for data transmission is valid (at least some proportion of the total data generated traverses via the route), the corresponding value of the FTRV or the CTRV is set as non-zero, i.e., $X^{fog}_{v_i,e,f}(t) > \lambda$ or $X^{cld}_{f,g,c,d}(t) > \lambda$, accordingly, and is set to zero, otherwise. λ denotes the proportion of the data which traversed through the concerned route, and $\lambda \in (0, 1]$. Clearly, $\sum_{v_i\in\mathcal{V},e\in\mathcal{E},f\in\mathcal{F}} X^{fog}_{v_i,e,f}(t) = 1$ implies that every byte of data originated from v_i reaches f without any loss, whereas, $\sum_{v_i\in f\in\mathcal{F},g\in\mathcal{G},c,d\in\mathcal{D}} X^{cld}_{f,g,c,d}(t) = 1$ indicates that the amount of data redirected towards the DC d by the FI f has reached its destination without any information loss. At any given time instant t, the set of all feasible FTRVs $(\mathcal{X}^{fog})$ is expressed as:

$$\mathcal{X}^{fog} = \Big\{ X^{fog}_{v_i,e,f}(t) \mid X^{fog}_{v_i,e,f}(t) = [0,1] \text{ and } \sum_{v_i\in\mathcal{V},e\in\mathcal{E},f\in\mathcal{F}} X^{fog}_{v_i,e,f}(t) = 1, \forall v_i \in \mathcal{V}, \forall e \in \mathcal{E}, \forall f \in \mathcal{F} \Big\}. \tag{2}$$

Similarly, at time t, the set of feasible CTRVs $(\mathcal{X}^{cld})$ is given by:

$$\mathcal{X}^{cld} = \Big\{ X^{cld}_{f,g,c,d}(t) \mid X^{cld}_{f,g,c,d}(t) = [0,1] \text{ and } \sum_{v_i\in f\in\mathcal{F},g\in\mathcal{G},c,d\in\mathcal{D}} X^{cld}_{f,g,c,d}(t) = 1, \forall f \in \mathcal{F}, \forall g \in \mathcal{G}, \forall c, d \in \mathcal{D} \Big\}. \tag{3}$$

A. Power Consumption

The total power consumption is divided into three broad categories for application requests which are served by the fog computing tier without inference of the cloud framework, and into four broad categories for requests which are required to be served by the cloud computing tier. Following is the list of factors which are responsible for power consumption during big data handling using the fog computing framework.

1) Data forwarding: While forwarding of data packets, the overall power consumed due to reception of the byte-stream, initial processing required for routing, and subsequent transmission of the same, is categorized as *power consumption due to data forwarding*. For data packets which require real-time, low-latency services, and are processed at the fog computing tier, and the corresponding power consumption due to data forwarding $(\Psi^{fog}_{df}(t))$[1] at time t is computed as:

$$\begin{aligned}
\Psi^{fog}_{df}(t) &= (\gamma_{eg} + \gamma_{fi}) \Big[\sum_{i=1}^{V} \{ P^{v_i}_r(t) - Q^{v_i}_r(t) + P^{v_i}_s(t) \\
&\qquad - Q^{v_i}_s(t) \} \sum_{v_i,e,f} X^{fog}_{v_i,e,f}(t) \Big] \\
&= (\gamma_{eg} + \gamma_{fi}) \Big[\sum_{v_i,e,f} \{ P^{v_i}_r(t) - Q^{v_i}_r(t) + P^{v_i}_s(t) \\
&\qquad - Q^{v_i}_s(t) \} X^{fog}_{v_i,e,f}(t) \Big],
\end{aligned} \tag{4}$$

where γ_{eg} and γ_{fi} represent the amount of energy required per second (power) to forward unit byte of data by the edge gateways and the fog instances, respectively.

Similarly, for data packets which demand to be processed at the core cloud computing module for complex analysis and long-term storage, the corresponding power consumption required for forwarding of the data packets $(\Psi^{cld}_{df}(t))$[2] at time t is expressed as:

$$\begin{aligned}
\Psi^{cld}_{df}(t) &= (\gamma_{eg} + \gamma_{fi}) \Big[\sum_{i=1}^{V} \{ Q^{v_i}_r(t) + Q^{v_i}_s(t) \} \sum_{v_i,e,f} X^{fog}_{v_i,e,f}(t) \Big] \\
&\quad + \gamma_{cl} \Big[\sum_{i=1}^{V} \{ Q^{v_i}_r(t) + Q^{v_i}_s(t) \} \sum_{f,g,c,d} X^{cld}_{f,g,c,d}(t) \Big] \\
&= \sum_{i=1}^{V} \{ Q^{v_i}_r(t) + Q^{v_i}_s(t) \} \Big[(\gamma_{eg} + \gamma_{fi}) \sum_{v_i,e,f} X^{fog}_{v_i,e,f}(t) \\
&\qquad + \gamma_{cl} \sum_{f,g,c,d} X^{cld}_{f,g,c,d}(t) \Big],
\end{aligned} \tag{5}$$

where γ_{cl} is the power required to forward unit byte of data by a cloud gateway.

[1]The notation $\sum_{v_i,e,f}$, as used in the paper essentially means the same as the notation $\sum_{v_i\in\mathcal{V},e\in\mathcal{E},f\in\mathcal{F}}$.

[2]The notation $\sum_{f,g,c,d}$ is used as an alternative for the notation $\sum_{v_i\in f\in\mathcal{F},g\in\mathcal{G},c,d\in\mathcal{D}}$ in the manuscript.

2) Computation: The power consumption due to computation also occur at both fog tier and cloud tier. At the fog computing layer, these computations are mostly real-time, low-latency services which are required to meet the continuous and ever-increasing demand of the TNs, otherwise stated as the IoE devices. Computation and analysis at the fog tier involves the temporarily stored data-sets within the fog computing devices. Let τ denote the time-to-live for every data packet after which it is removed from the temporary fog storage. Therefore, at time t, the computational power consumption at the fog layer $(\Psi_{cp}^{fog}(t))$ depends on the data stored within the FIs from time $(t-\tau)$ till the present time-slot. $(\Psi_{cp}^{fog}(t))$ is mathematically expressed as:

$$\Psi_{cp}^{fog}(t) = \beta^{fog} \sum_{j=t-\tau}^{t} \phi_j^{fog} \sum_{i=1}^{V} \{P_s^{v_i}(j) - Q_s^{v_i}(j)\}, \quad (6)$$

where β^{fog} is the average per byte computational power consumption (obtained as the ratio of the power exhausted for the processing of an instruction to the number of bytes in the instruction). ϕ_j^{fog} is the weight-factor associated with the data-set which is required for analysis, and the magnitude of ϕ_j^{fog} decreases with the increase in the *age* of the data. The magnitude of ϕ_j^{fog} lies within $(0,1]$, with $phi_j^{fog} = 1$ for $j = t$. The expression $\sum_{i=1}^{V}\{P_s^{v_i}(j) - Q_s^{v_i}(j)\}$ indicates the cumulative amount of data which is stored at time t within the temporary fog storage units, for analysis and computation.

For the cloud computing tier, computations and subsequent analysis are highly extensive in terms of the volume of data involved and the complexity associated. At time instant t, the power consumption at the cloud computing framework due to computation and analysis, $\Psi_{cp}^{cld}(t)$, is dependent on the cumulative amount of data which stored within the DCs, starting from the very beginning, i.e., $t = 0$. $(\Psi_{cp}^{cld}(t))$ is computed as:

$$\Psi_{cp}^{cld}(t) = \beta^{cld} \sum_{j=0}^{t} \phi_j^{cld} \sum_{i=1}^{V} Q_s^{v_i}(j), \quad (7)$$

where β^{cld} id the mean computational power required to process unit byte at the cloud-end, and $\sum_{i=1}^{V} Q_s^{v_i}(j)$ is the total amount of data aggregated within a DC for processing and computation, with $\phi_j^{cld} \in [0,1]$. Clearly, for $j = t$, $\phi_j^{cld} = 1$, whereas, as $j \to 0$, $\phi_j^{cld} \to 0$.

3) Storage: The power consumption due to storage, similar to the computational power consumption, depends on the number of bytes of data which is stored in the database for processing and analysis. However, as the storage power consumption is independent of the *age* of the data, unlike the computational power, this term is void of any such weight-factors. The mathematical expression for the storage power consumption at time t for the fog tier $(\Psi_{st}^{fog}(t))$ is given by:

$$\Psi_{st}^{fog}(t) = \alpha^{fog} \sum_{j=0}^{t} \sum_{i=1}^{V} \{P_s^{v_i}(j) - Q_s^{v_i}(j)\}. \quad (8)$$

Similarly, for the cloud computing tier, the storage power consumption $(\Psi_{st}^{cld}(t))$ is computed as:

$$\Psi_{st}^{cld}(t) = \alpha^{cld} \sum_{j=0}^{t} \sum_{i=1}^{V} Q_s^{v_i}(j), \quad (9)$$

where α^{fog} and α^{cld} represent the per byte per unit time energy consumption to storage data within the databases in the fog tier and the cloud DC, respectively.

4) Data migration: The power consumption due to data migration is only associated with the cloud computing layer. In other words, applications which require real-time services data migration is irrelevant as they are served by the fog computing tier without the intervention of the cloud computing framework. However, for applications which demand complex and historical analysis of data, migration of data from different geo-spatially distributed DCs towards the aggregator DC is required. As the aggregator DC may vary in every time-slot, at any given time-slot t, it is important to migrate all data to the aggregator DC for subsequent processing. At time t, the overall migration cost $(\Psi_{mg}^{cld}(t))$ within the cloud computing framework is given by:

$$\Psi_{mg}^{cld}(t) = \begin{cases} \sum_{c\in\mathcal{D}} \sum_{d\in\mathcal{D}} \eta_{cd} \sum_{j=0}^{t-1} \phi_j^{cld} \sum_{i=1}^{V} Q_s^{v_i}(j), & \text{if } \mathcal{A}_t \neq \mathcal{A}_{t-1} \\ 0 & \text{, otherwise} \end{cases}, \quad (10)$$

where η_{cd} is the per byte power consumption for cost migration of data from DC c to DC d, $c, d \in \mathcal{D}$. $\mathcal{A}_t$ denotes the aggregator DC at time-slot t. Therefore, $\mathcal{A}_t = \mathcal{A}_{t-1}$ implies that the aggregator DC in time t is same as it was in the previous time-slot, indicating no additional data migration is required.

Therefore, at time t, the overall power consumption for applications which are served by the fog computing tier, $\Psi^{fog}(t)$ is computed as:

$$\Psi^{fog}(t) = \Psi_{df}^{fog}(t) + \Psi_{cp}^{fog}(t) + \Psi_{st}^{fog}(t). \quad (11)$$

Whereas, for application seeking intervention of the core cloud computing, the cumulative power consumption at time t is calculated as:

$$\Psi^{cld}(t) = \Psi_{df}^{cld}(t) + \Psi_{cp}^{cld}(t) + \Psi_{st}^{cld}(t) + \Psi_{mg}^{cld}(t). \quad (12)$$

B. Service Latency

Service latency for a request sent by an application instance running within a TN is basically its response time, and is computed as the sum of the transmission latency and the processing latency for the request. As mentioned earlier, the communication among the different TNs within a EVPN incur insignificant latency as sufficient bandwidth-support is provided within every EVPN for the purpose. Similarly, inter-DC communication and data migration, which take place along the (c, d) link are also considered to invoke negligible latency due to the high-bandwidth, $\mathbb{W}(c, d)$, dedicated available within the cloud-core, $\forall c, d \in \mathcal{D}$ [4]. However, the bottleneck of communication is formed by the links (e, f) and (f, g), $\forall e \in \mathcal{E}, \forall f \in \mathcal{F}, \forall g \in \mathcal{G}$. The bandwidth between the edge gateways and the FIs, $\mathbb{W}(e, f)$ is limited, and at any given time instant, as the active number of TNs in tier 1 increases, the service latency also increases. The communication links between fog gateways and the cloud gateways are bandwidth constrained ($\mathbb{W}(f, g)$) as well. In this Section we discuss the service latency for the applications, and mathematically formulate the expression for the same. The service latency is divided into two subheads as mentioned below.

1) Transmission Latency: Let δ_{ef} and δ_{fg} are the delays in unit byte data transmission from a EVPN to the corresponding FI, and from an fog gateway to the cloud gateway, respectively. For applications which are served by the cloud computing tier, the transmission latency at time t, ($\delta_{tr}^{fog}(t)$), is computed as:

$$\delta_{tr}^{fog}(t) = \delta_{ef} \sum_{i=1}^{V} \{P_r^{v_i}(t) + P_s^{v_i}(t) - Q_r^{v_i}(t) - Q_s^{v_i}(t)\}. \quad (13)$$

Similarly, for requests which are handled at the cloud computing tier, the corresponding transmission latency ($\delta_{tr}^{cld}(t)$) is expressed as:

$$\delta_{tr}^{cld}(t) = (\delta_{ef} + \delta_{fg}) \sum_{i=1}^{V} \{Q_r^{v_i}(t) + Q_s^{v_i}(t)\}. \quad (14)$$

Therefore, in presence of the fog computing tier, the mean transmission latency at time t (Δ_{tr}^{fog}) is computed as:

$$\Delta_{tr}^{fog}(t) = \frac{\delta_{ef} \sum_{i=1}^{V} \{P_r^{v_i}(t) + P_s^{v_i}(t)\} + \delta_{fg} \sum_{i=1}^{V} \{Q_r^{v_i}(t) + Q_s^{v_i}(t)\}}{\sum_{i=1}^{V} \{P_r^{v_i}(t) + P_s^{v_i}(t)\}} \quad (15)$$

On the contrary, in the traditional cloud computing framework, the corresponding mean transmission latency ($\Delta_{tr}^{cld}(t)$) is given as:

$$\Delta_{tr}^{cld}(t) = \frac{\delta_{eg} \sum_{i=1}^{V} \{P_r^{v_i}(t) + P_s^{v_i}(t)\}}{\sum_{i=1}^{V} \{P_r^{v_i}(t) + P_s^{v_i}(t)\}} \quad (16)$$

where δ_{eg} is the latency associated with unit byte data transmission from a TN to the cloud DC. Clearly, from the triangle inequality, we have $\delta_{eg} \geq \delta_{ef} + \delta_{fg}$.

2) Processing Latency: Processing latency for an application instance request within the fog computing tier is defined as the time required to serve the request after analyzing the data accumulated during the previous τ time-slots within the fog computing devices. Mathematically, at time t, the processing latency, $\delta_{pr}^{fog}(t)$, within the fog computing tier is expressed as:

$$\delta_{pr}^{fog}(t) = \left(P_r^{v_i}(t) - Q_r^{v_i}(t)\right)\zeta^{fog} \sum_{j=t-\tau}^{t} \phi_j^{fog} \sum_{i=1}^{V} \{P_s^{v_i}(j) - Q_s^{v_i}(j)\}. \quad (17)$$

Similarly, for applications which are referred to the cloud computing layer for processing purpose, the processing latency is defined in terms of the time required to analyze the cumulative migrated data-sets from the cloud DCs within the aggregator DC. The processing latency ($\delta_{pr}^{cld}(t)$) within the cloud computing core is, therefore, computed as:

$$\delta_{pr}^{cld}(t) = Q_r^{v_i}(t)\zeta^{cld} \sum_{j=t-\tau}^{t} \phi_j^{cld} \sum_{i=1}^{V} Q_s^{v_i}(j). \quad (18)$$

where ζ^{fog} and ζ^{cld} represent the per byte processing latency at the fog computing and cloud computing tiers, respectively. The mean processing delay at time t for a fog computing environment is:

$$\Delta_{pr}^{cld}(t) = \Big[\left(P_r^{v_i}(t) - Q_r^{v_i}(t)\right)\zeta^{fog} \sum_{j=t-\tau}^{t} \phi_j^{fog} \sum_{i=1}^{V} \{P_s^{v_i}(j) - Q_s^{v_i}(j)\} + Q_r^{v_i}(t)\zeta^{cld} \sum_{j=t-\tau}^{t} \phi_j^{cld} \sum_{i=1}^{V} Q_s^{v_i}(j)\Big] \Big/ \sum_{i=1}^{V} \{P_r^{v_i}(t) + P_s^{v_i}(t)\}. \quad (19)$$

However, the average transmission latency, $\Delta_{pr}^{cld}(t)$, at time t for a traditional cloud computing framework is given as:

$$\Delta_{pr}^{cld}(t) = \frac{P_r^{v_i}(t)\zeta^{cld} \sum_{j=t-\tau}^{t} \phi_j^{cld} \sum_{i=1}^{V} P_s^{v_i}(j)}{\sum_{i=1}^{V} \{P_r^{v_i}(t) + P_s^{v_i}(t)\}}. \quad (20)$$

Finally, to compute the mean service latency, we simply add the mean transmission latency and the mean processing latency for the corresponding computing framework.

VI. Case Study: Simulation Setup

This Section illustrates the network setup, power consumption, CO_2 emission rate, and the cost variables corresponding to the fog computing framework.

A. Network Topology

The essential nodes of the fog computing network includes the sets of the TNs $\mathcal{N}$, the FIs $\mathcal{F}$, and the cloud DCs $\mathcal{D}$. We consider a global deployment of these essential nodes. For this purpose, the work considers 100 highest populated cities across the globe [49], the respective population of people using Internet services [50], and the corresponding geographic location of the cities [51], as shown in Fig. 3. The matrix $L_c[1..100][1..100]$ stores the relative Euclidean distance between any pair of cities. The TNs within a particular city are logically grouped to form a VC.

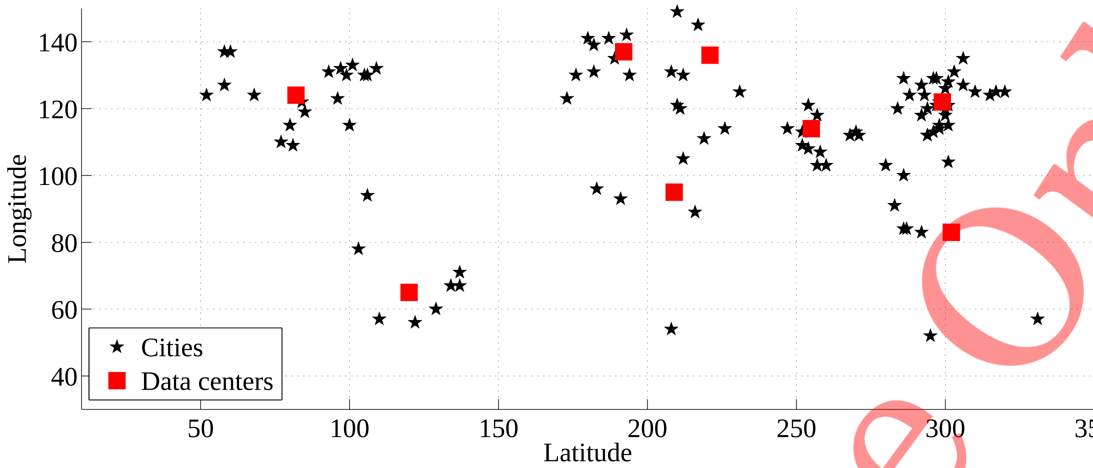
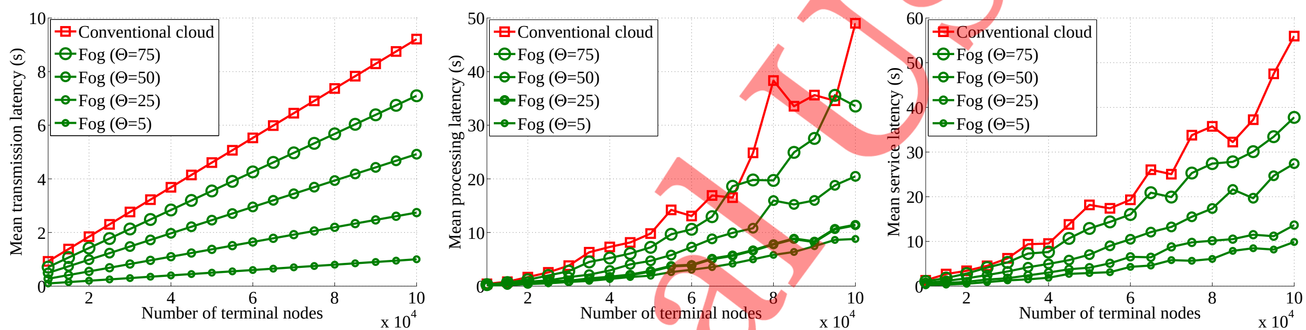


Fig. 3: Global deployment scenario

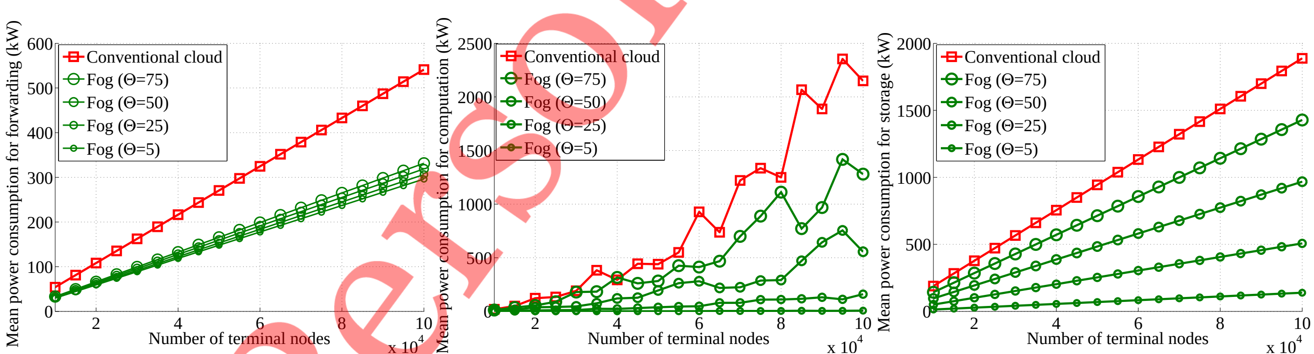


(a) Tranamission latency vs No. of TNs

(b) Processing latency vs No. of TNs

(c) Service latency vs No. of TNs

Fig. 4: Analysis of service latency



(a) Power consumption due to data forwarding

(b) Power consumption due to computation

(c) Power consumption due to data storage

Fig. 5: Analysis of power consumption

B. Network Traffic

The data traffic generated from the cities (or VCs) is proportional to the population of Internet users of the corresponding city. Data from the EVPNs are transmitted to the fog computing tier in form of packets. These packet sizes typically vary between a minimum of 34 bytes (header + FCS only) to a maximum of 65550 bytes. The instruction size is taken as 64 bits. Communication among the TNs within a EVPN and that among the DCs within the cloud computing core are assumed to take place through bandwidth unconstrained channels, as mentioned earlier. However, the capacity of the (e, f) links, $\forall e \in \mathcal{E}, \forall f \in \mathcal{F}$ is considered to be 1 Gbps. On the other hand, the (f, g), $\forall f \in \mathcal{F}, \forall g \in \mathcal{G}$ link capacity is taken as 10 Gbps.

C. Terminal Nodes

The total number of TNs in the system is treated as a variable, within the range $[10,000,\ 100,000]$, to assess the system performance against varied network conditions. The number of TNs in each of the 100 cities is taken as proportional to the population of Internet users of the city. The TNs transmit their data through access points distributed within each city. All TNs within a city are considered to send their data to a single FI.

D. Data Centers

The number of DCs present worldwide is considered to be 8. Based on the clustering of the city-population, the location of the DCs are determined. The pair-wise Euclidean distances between the DCs are stored in the matrix $L_d[1..8][1..8]$. Every DC is assumed to accommodate varied number of IT components within the discrete set $\{16,000, 32,000, 64,000, 128,000\}$, based on the network traffic is processes.

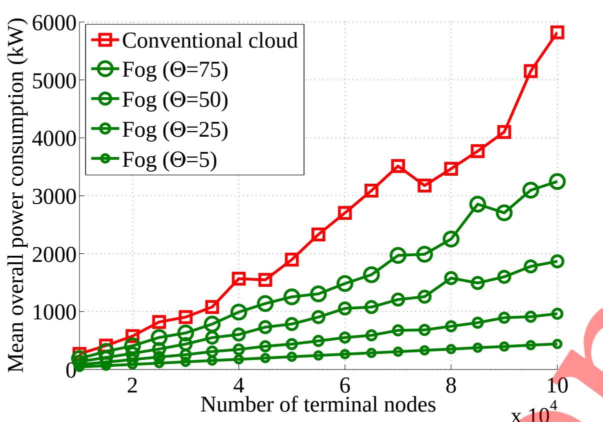


Fig. 6: Overall power consumption vs No. of TNs

E. Power Consumption

The power consumed by the different network, computing, and storage elements are used to compute the overall power consumption by the system under the discussed scenario. The power consumed by each 1 Gbps and 10 Gbps router is taken as 20 W and 40 W, respectively. Each 3-layer network switch and storage component consumes 350 W and 600 W, respectively. The fog computing devices, within an FI, consumes a total of 3.7 W collectively. Power consumption by the cloud DCs are taken to be proportional to the IT elements within those, and taken from the range $\{9.7, 19.4, 38.7, 77.4\}$ MW.

F. CO_2 Emission

The sources of the energy supplied to the cloud DCs determines the amount the CO_2 gas produced. Table I depicts the CO_2 emission rate for the different energy sources taken for analysis [52]:

TABLE I: CO_2 emission rates

Type of energy source	Energy source	CO2 emission rate (in g/kWh)
Non-renewable	Coal	960
	Diesel	778
	Natural gas	443
Renewable	Geothermal	38
	Hydroelectric	10
	Wind	9

G. Cost

For analysis of the cost both in terms of the operational cost and the CO_2 emission penalty, we define the cost of the different component elements. Each 1 Gbps and 10 Gbps router port costs \$50/year. The Cost for each server is taken as \$4,000/year [53]. The uploading cost for each byte of data is taken as \$ 12, and the storage cost per GB of data is taken uniformly from within the range $0.45 - 0.55$ \$/hour. The electricity cost is taken as uniformly distributed between \$30/MHWh and \$70/MWh [53]. The penalty for CO_2 emission, however, was taken in higher degree. The penalty was decided to be \$1,000/ton CO_2 emitted.

VII. Case Study: Performance Evaluation

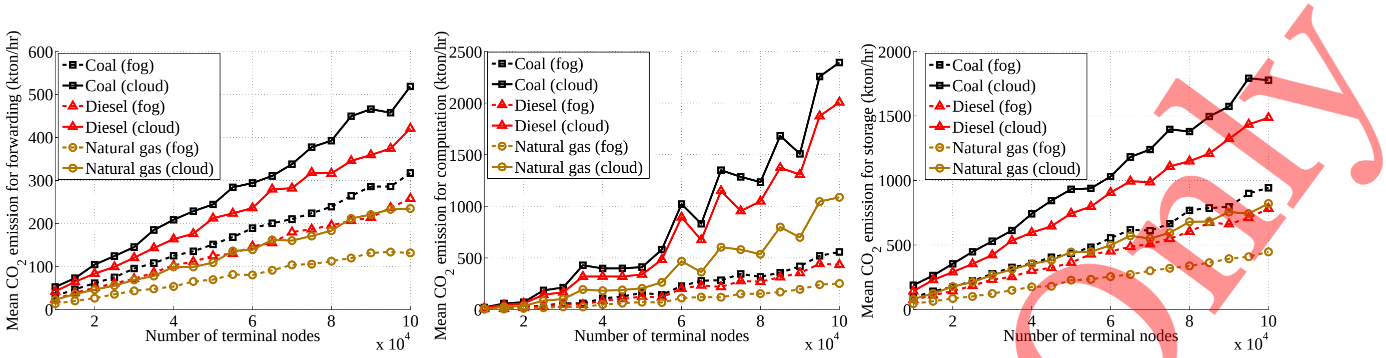
In details of the results obtained in terms of the performance metrics are presented and analyzed in this Section. The simulation setup is mentioned in details in Section VI. Additionally, we compare the performance of the fog computing paradigm wit that of the traditional cloud computing architecture, and present a thorough study against the same.

A. Service Latency

First, we analyze the variation of the service latency (transmission latency + processing latency) with the number of TNs present in tier 1. We define the ratio of the total bytes transmitted to the fog computing tier to the number of bytes referred to the cloud computing core as the cloud transmission ratio, mathematically represented by the variable Θ. With the change in the magnitude of Θ, within the range $[0.05, 0.75]$, we plot the transmission latency and propagation latency for both fog computing and the conventional cloud computing architectures, and observe the change in the corresponding services latencues.

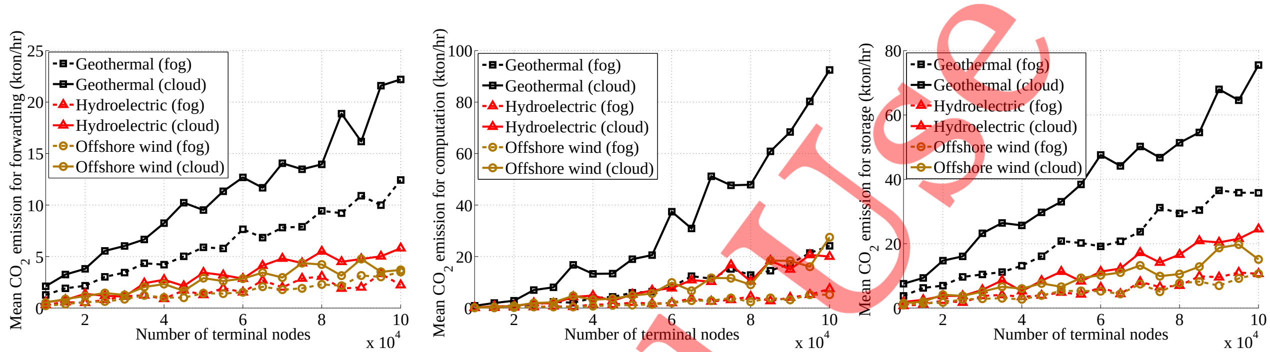
In Fig. 4(a) and Fig. 4(b), the mean transmission latency and mean processing latency are plotted, separately, against variable number of TNs, for different magnitudes of Θ. It is observed that with the decrease in the magnitude of Θ, as more number of application requests demand real-time and latency-sensitive services, the mean transmission latency and the mean processing latency are observed to diminish. Also, with the increase in the number of TNs, the tranmission and processing latencies increase.

Consequently, the overall service latency for both these computing paradigm follow the same pattern. In summary, for a very low percentage of applications which demand real-time services, i.e., for a very high magnitude of Θ ($\Theta \simeq 100$), the



(a) CO_2 emission due to data forwarding (b) CO_2 emission due to computation (c) CO_2 emission due to data storage

Fig. 7: Analysis of CO_2 emission for non-renewable energy sources



(a) CO_2 emission due to data forwarding (b) CO_2 emission due to computation (c) CO_2 emission due to data storage

Fig. 8: Analysis of CO_2 emission for renewable energy sources

service latency in fog computing becomes almost same with that in a cloud computing environment.

B. Power Consumption

The power consumption for due to the individual effects of data forwarding, computation, and data storage, and due to their collective effect is analyzed in this subsection. As shown in Fig. 5(a), although with the increase in the number of TNs in the lowest tier, the mean power consumption due to data forwarding increases linearly, the impact of the change in the power consumption for different magnitudes of Θ is observed to be very low. However, compared to a conventional cloud computing framework, this mean power consumption was significantly less.

In Fig. 5(b) and Fig. 5(c), the variation in the mean power consumption due to computation and storage, respectively, are shown against the change in the number of TNs. Similar inferences are drawn from the two figures, as the power consumption in both these cases are noticed to decrease by a significant margin as the magnitude of Θ is decreased.

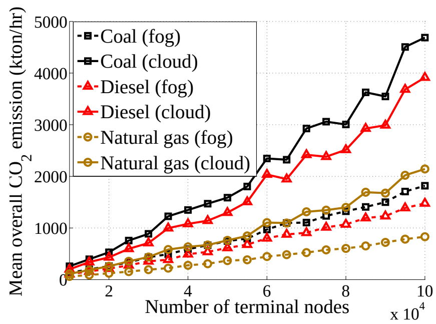
In presence of the fog computing tier, the mean cumulative power consumption, as shown in Fig. 6, is always less than that in a conventional cloud computing. We observe that compared to the power consumption due to data forwarding, the amount of power consumed due to computation and storage are considerably high. For $\Theta = 75$, the overall mean power consumption is calculated to be 42.2% less in fog computing.

C. CO_2 Emission

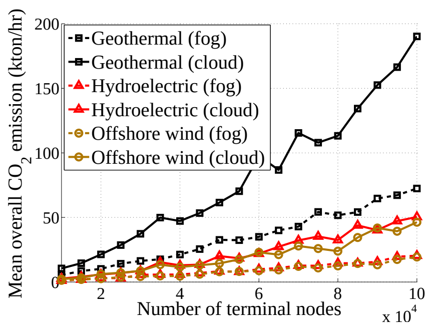
The impact of power consumption on the environment is depicts through the amount of CO_2 gas emission in the process. As mentioned in Section VI, we divide our analysis based on the type of the energy source – non-renewable and renewable. For analysis of CO_2 emission in both cloud and fog systems, the predominant factor that we have assumed is the power consumption of the DCs in each case. The experiment considers distinct multiplicative factors for various non-renewable and renewable energy sources (Table I).

In Fig. 7, variation in the mean CO_2 emission for each of the individual factors, viz., data forwarding, computation, and data storage is plotted against variable number of TNs. As indicated in Fig. 7(a),the average CO_2 emission for packet forwarding is notably high in cloud computing paradigm compared to fog computing paradigm when coal, diesel, and natural gas are considered. In Fig. 7(a), it is observed that, for the same non-renewable energy source type, the mean CO_2 emission stands greener compared to conventional cloud systems while taking computation into account. Lastly, even for the purpose of storage, we observe a significant reduction in CO_2 emission in Fig. 7(c).

We now examine the case of green computation for renewable energy sources, as shown in Fig. 8. For the purpose of study, we have assumed three types of renewable energy sources – geothermal, hydroelectric, and offshore wind. Similar to Fig. 7, it is observant that fog computing platform are distinctly greener in trems of CO_2 emission compared to cloud platforms, considering packet forwarding (Fig. 8(a)), computation (Fig. 8(b)), and storage (Fig. 8(c)) into account.



(a) Total CO_2 emission for renewable sources



(b) Total CO_2 emission for non-renewable sources

Fig. 9: Comparison of total CO_2 emission for different sources

A combined analysis of Figs. 7 and 8 is presented in Figs. 9(a) and 9(b), respectively. For the sake of data aggregation we have assumed $\Theta = 75$. For non-renewable energy sources, we find that the over CO_2 emission is decreased by 59.26% for coal, 57.58% for diesel, and 55.56% for natural gas through fog computing, compared to cloud computing. On the other hand, for renewable energy sources, we observe that fog computing achieves a reduction of 56.94% for geothermal energy, 55.79% for hydroelectric energy, and 54.95% for offshore wind energy.

D. Cost

In this Section, we analyze the cost incurred in both cloud and fog computing environments for various operations.

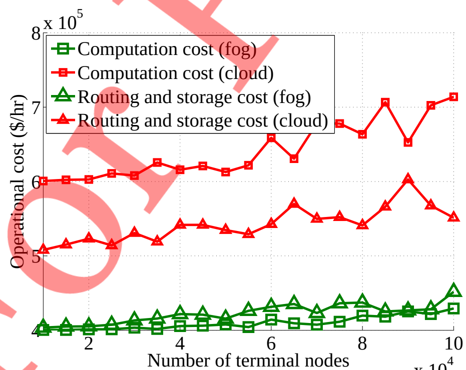
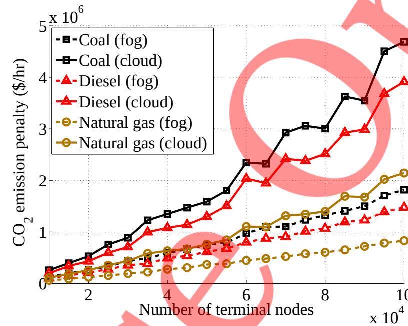
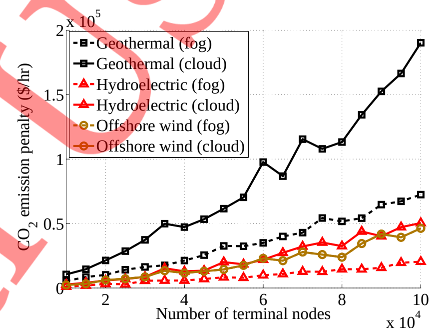


Fig. 10: Operational cost per year

Fig. 10 indicates the standard cost incurred. Based on the power consumption due to computation, we evaluate the cost for computation. Similarly, we also evaluate the cost for routing and storage. Both the costs are observed to be considerably higher in cloud environments than fog environments.



(a) CO_2 emission penalty for non-renewable energy sources



(b) CO_2 emission penalty for renewable energy sources

Fig. 11: Analysis of cost for non-renewable energy sources

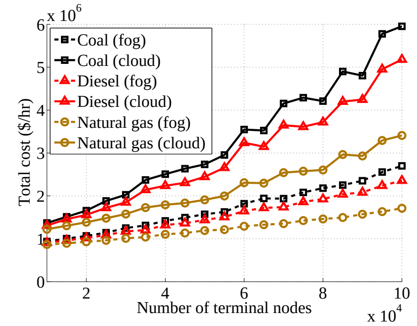
In Fig. 11, we evaluate the penalty to be paid by the TNs for consumption of renewable and non-renewable energy sources. From Fig. 11(a) we observe that the penalty is considerably high for coal in cloud environments, whereas, in fog environments, it is remarkably low. Similarly, for diesel and natural gas, the penalty are much lower in fog environments than cloud platforms.

The analysis of penalty for consumption of renewable energy sources is shown in Fig. 11(b). The penalty due to consumption of geothermal energy is much higher in cloud systems. The other types viz. hydroelectric and offshore wind energy penalty are also reduced in fog platforms, compared to cloud platforms.

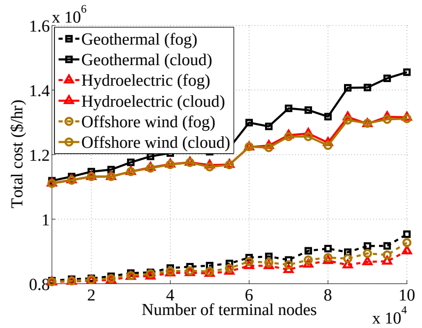
Thus, the effects of Figs. 10 and 11 are combined in Fig. 12 to obtain the total incurred cost. It is observed that both for non-renewable (Fig. 12(a)) and renewable (Fig. 12(b)) energy sources the fog based systems exhibit a cheaper nature compared to cloud based systems.

VIII. Conclusion

The work focuses on analyzing the suitability of fog computing within the framework of IoT. The goal of this paper is to develop a mathematical model of fog computing and assess its applicability in the context of IoT where it is pivotal



(a) Total cost for non-renewable energy sources



(b) Total cost for renewable energy sources

Fig. 12: Analysis of cost for renewable energy sources

to meet the demands of the latency-sensitive applications running at the network-edge. The work further performs a comparative performance evaluation of cloud computing with that of fog computing for an environment with high number of Internet-connected devices demanding real-time services. Results clearly depict the enhanced performance of fog computing both in terms of the provisioned QoS and eco-friendliness under such situations. We eventually justify fog paradigm as an improved, eco-friendly computing platform that can support IoT better compared to the existing cloud computing paradigm.

In the future, we plan to extend this work by proposing a working fog computing prototype to support real-time implementation. The results of real network traffic and CO_2 emission rate of the DCs can be utilized to strengthen the model and support the future 'green' IoT technologies.

References

[1] L. Piras. (2014, March) A brief history of the internet of things [info-graphic]. [Online]. Available: http://www.psfk.com/2014/03/internet-of-things-infographic.html

[2] Internet of things market forecast:. Cisco. [Online]. Available: http://postscapes.com/internet-of-things-market-size

[3] (2014, Jan.) Cisco delivers vision of fog computing to accelerate value from billions of connected devices. Press release. Cisco. [Online]. Available: http://newsroom.cisco.com/release/1334100/Cisco-Delivers-Vision-of-Fog-Computing-to-Accelerate-Value-from-Billions-of-Connected-Devices-utm-medium-rss

[4] L. Zhang, C. Wu, Z. Li, C. Guo, M. Chen, and F. Lau, "Moving big data to the cloud: An online cost-minimizing approach," *IEEE Journal on Selected Areas in Communications*, vol. 31, pp. 2710–2721, Dec. 2013.

[5] D. Lund, C. MacGillivray, V. Turner, and M. Morales, "Worldwide and regional internet of things (iot) 20142020 forecast: A virtuous circle of proven value and demand," International Data Corporation (IDC), Tech. Rep., 2014.

[6] D. Bouley, "Estimating a data center's electrical carbon footprint," April 2010. [Online]. Available: https://www.insight.com/content/dam/insight/en-US/pdfs/apc/apc-estimating-data-centers-carbon-footprint.pdf

[7] "Report to congress on server and data center energy efficiency public law 109−431," U.S. Environmental Protection Agency ENERGY STAR Program, Tech. Rep., July 2007.

[8] M. P. Mills, "The cloud begins with coal," Digital Power Group, Tech. Rep., August 2013.

[9] D. He and S. Zeadally, "An analysis of rfid authentication schemes for internet of things in healthcare environment using elliptic curve cryptography," *IEEE Internet of Things Journal*, vol. 2, no. 1, pp. 72–83, Feb 2015.

[10] I. F. Akyildiz, M. Pierobon, S. Balasubramaniam, and Y. Koucheryavy, "The internet of bio-nano things," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 32–40, March 2015.

[11] A. Aijaz and A. Aghvami, "Cognitive machine-to-machine communications for internet-of-things: A protocol stack perspective," *IEEE Internet of Things Journal*, vol. 2, no. 2, pp. 103–112, April 2015.

[12] A. Ortiz, D. Hussein, S. Park, S. Han, and N. Crespi, "The cluster between internet of things and social networks: Review and research challenges," *IEEE Internet of Things Journal*, vol. 1, no. 3, pp. 206–215, June 2014.

[13] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645 – 1660, 2013.

[14] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An information framework for creating a smart city through internet of things," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 112–121, April 2014.

[15] C. Perera, C. Liu, S. Jayawardena, and M. Chen, "A survey on internet of things from industrial market perspective," *IEEE Access*, vol. 2, pp. 1660–1679, 2014.

[16] J. Wei, "How wearables intersect with the cloud and the internet of things : Considerations for the developers of wearables." *IEEE Consumer Electronics Magazine*, vol. 3, no. 3, pp. 53–56, July 2014.

[17] L. Wang and R. Ranjan, "Processing distributed internet of things data in clouds," *IEEE Cloud Computing*, vol. 2, no. 1, pp. 76–80, Jan 2015.

[18] X. Zheng, P. Martin, K. Brohman, and L. D. Xu, "Cloudqual: A quality model for cloud services," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1527–1536, May 2014.

[19] H. Yue, L. Guo, R. Li, H. Asaeda, and Y. Fang, "Dataclouds: Enabling community-based data-centric services over the internet of things," *IEEE Internet of Things Journal*, vol. 1, no. 5, pp. 472–482, Oct 2014.

[20] M. Nitti, R. Girau, and L. Atzori, "Trustworthiness management in the social internet of things," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 5, pp. 1253–1266, May 2014.

[21] X. Zheng, P. Martin, K. Brohman, and L. D. Xu, "Cloud service negotiation in internet of things environment: A mixed approach," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1506–1515, May 2014.

[22] P. B. S. U. Khan and T. Engel, "Energy-efficient high-performance parallel and distributed computing, journal of supercomputing," *Journal of Supercomputing*, vol. 60, pp. 163–164, 2012.

[23] K. Bilal, S. Malik, S. Khan, and A. Zomaya, "Trends and challenges in cloud datacenters," *IEEE Cloud Computing*, vol. 1, no. 1, pp. 10–20, May 2014.

[24] B. Sotomayor, R. S. Montero, I. Llorente, and I. Foster, "Virtual infrastructure management in private and hybrid clouds," *IEEE Internet Computing*, vol. 13, no. 5, pp. 14–22, Sept 2009.

[25] Q. Duan, Y. Yan, and A. Vasilakos, "A survey on service-oriented network virtualization toward convergence of networking and cloud computing," *IEEE Transactions on Network and Service Management*, vol. 9, no. 4, pp. 373–392, December 2012.

[26] H. Liang, L. Cai, D. Huang, X. Shen, and D. Peng, "An smdp-based service model for interdomain resource allocation in mobile cloud networks," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 5, pp. 2222–2232, Jun 2012.

[27] T. Noor, Q. Sheng, A. Ngu, and S. Dustdar, "Analysis of web-scale cloud services," *IEEE Internet Computing*, vol. 18, no. 4, pp. 55–61, July 2014.

[28] A. Dastjerdi and R. Buyya, "Compatibility-aware cloud service composition under fuzzy preferences of users," *IEEE Transactions on Cloud Computing*, vol. 2, no. 1, pp. 1–13, Jan 2014.

[29] J. Xiao, H. Wen, B. Wu, X. Jiang, P.-H. Ho, and L. Zhang, "Joint design on dcn placement and survivable cloud service provision over all-optical mesh networks," *IEEE Transactions on Communications*, vol. 62, no. 1, pp. 235–245, January 2014.

[30] Z. Zheng, X. Wu, Y. Zhang, M. Lyu, and J. Wang, “Qos ranking prediction for cloud services,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 6, pp. 1213–1222, June 2013.

[31] W. Chen, J. Cao, and Y. Wan, “Qos-aware virtual machine scheduling for video streaming services in multi-cloud,” *Tsinghua Science and Technology*, vol. 18, no. 3, pp. 308–317, June 2013.

[32] M. Mishra, A. Das, P. Kulkarni, and A. Sahoo, “Dynamic resource management using virtual machine migrations,” *IEEE Communications Magazine*, vol. 50, no. 9, pp. 34–40, September 2012.

[33] N. Tziritas, S. U. Khan, C.-Z. Xu, T. Loukopoulos, and S. Lalis, “On minimizing the resource consumption of cloud applications using process migrations,” *Journal of Parallel and Distributed Computing*, vol. 73, pp. 1690–1704, 2013.

[34] A. A. Chandio, K. Bilal, N. Tziritas, Z. Yu, Q. Jiang, S. U. Khan, and C.-Z. Xu, “A comparative study on resource allocation and energy efficient job scheduling strategies in large-scale parallel computing systems,” *Cluster Computing*, vol. 17, 2014.

[35] F. Zhang, J. Cao, W. Tan, S. U. Khan, K. Li, and A. Y. Zomaya, “Evolutionary scheduling of dynamic multitasking workloads for big-data analytics in elastic cloud,” *IEEE Transactions on Emerging Topics in Computing*, 2013.

[36] D. Kliazovich, S. Arzo, F. Granelli, P. Bouvry, and S. Khan, “e-STAB: Energy-efficient scheduling for cloud computing applications with traffic load balancing,” in *2013 IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing Green Computing and Communications (GreenCom)*, Aug 2013, pp. 7–13.

[37] B. Z. M. Feng and C.-H. Lung, “A green computing based architecture comparison and analysis,” in *IEEE/ACM International Conference on Cyber, Physical and Social Computing*, 2010.

[38] L. Xu, C. Li, L. Li, Y. Liu, Z. Yang, and Y. Liu, “A virtual data center deployment model based on the green cloud computing,” 2014.

[39] N. N. Naik, K. Kanagala, and J. P. Veigas, “Achieving green computing by effective utilization of cloud resources using a cloud os,” in *IEEE International Conference on Emerging Trends in Computing, Communication and Nanotechnology*, 2013.

[40] T. T. Huu and C.-K. Tham, “An auction-based resource allocation model for green cloud computing,” in *IEEE International Conference on Cloud Engineering*, 2013.

[41] K. Bilal, S. U. R. Malik, O. Khalid, A. Hameed, E. Alvarez, V. Wijaysekara, R. Irfan, S. Shrestha, D. Dwivedy, M. Ali, U. S. Khan, A. Abbas, N. Jalil, and S. U. Khan, “A taxonomy and survey on green data center networks,” *Future Generation Computer Systems*, vol. 36, no. 0, pp. 189 – 208, 2014.

[42] G. A. Lewis, “Mobile computing at the edge,” in *Proceedings of the 1st International Conference on Mobile Software Engineering and Systems*, 2014.

[43] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog computing and its role in the internet of things,” in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*. ACM, 2012, pp. 13–16.

[44] K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwälder, and B. Koldehofe, “Mobile fog: A programming model for large-scale applications on the internet of things,” in *Proceedings of the Second ACM SIGCOMM Workshop on Mobile Cloud Computing*, 2013, pp. 15–20.

[45] H. Madsen, G. Albeanu, B. Burtschy, and F. Popentiu-Vladicescu, “Reliability in the utility computing era: Towards reliable fog computing,” in *Systems, Signals and Image Processing (IWSSIP), 2013 20th International Conference on*, July 2013, pp. 43–46.

[46] J. Zhu, D. S. Chan, M. S. Prabhu, P. Natarajan, H. Hu, and F. Bonomi, “Improving web sites performance using edge servers in fog computing architecture,” in *IEEE Seventh International Symposium on Service-Oriented System Engineering*, 2013.

[47] S. Stolfo, M. Salem, and A. Keromytis, “Fog computing: Mitigating insider data theft attacks in the cloud,” in *Security and Privacy Workshops (SPW), 2012 IEEE Symposium on*, May 2012, pp. 125–128.

[48] J. Zao, T. T. Gan, C. K. You, S. Rodriguez Meendez, C. E. Chung, Y. T. Wang, T. Mullen, and T. P. Jung, “Augmented brain computer interaction based on fog computing and linked data,” in *International Conference on Intelligent Environments*, 2014.

[49] T. Brinkoff. (2013, January) City population. [Online]. Available: http://www.citypopulation.de/

[50] (2013, January) Internet world stats. [Online]. Available: http://www.internetworldstats.com/

[51] Google map. Google. [Online]. Available: https://www.google.com/maps

[52] B. K. Sovacool, “Valuing the greenhouse gas emissions from nuclear power: A critical survey,” *Energy Policy*, vol. 36, no. 8, pp. 2950–2963, Aug. 2008.

[53] F. Larumbe and B. Sanso, “A tabu search algorithm for the location of data centers and software components in green cloud computing networks,” *IEEE Transactions on Cloud Computing*, vol. 1, pp. 22–35, 2013.

Subhadeep Sarkar (S’13) is presently pursuing his Ph.D. from the School of Medical Science and Technology, Indian Institute of Technology Kharagpur, India. He is also working as a Senior Research Fellow in School of Information Technology, Indian Institute of Technology Kharagpur in a project funded by Department of Electronics and Information Technology, Government of India. He received his B.Tech. degree in Computer Science and Technology from West Bengal University of Technology, India in 2012. His current research interests include networking and communication aspects of Wireless Body Area Networks (WBANs), Cloud Computing, and Fog Computing.

Subarna Chatterjee (S’13) is a TCS Research Scholar pursuing her Ph.D. from the School of Information Technology, Indian Institute of Technology, Kharagpur, India. She received her B.Tech. degree in Computer Science and Technology from West Bengal University of Technology, India in 2012. Her current research interests include networking and communication aspects of Cloud Computing in Wireless Sensor Networks.

Dr. Sudip Misra (SM’13) is an Associate Professor in the School of Information Technology at the Indian Institute of Technology Kharagpur. Prior to this he was associated with Cornell University (USA), Yale University (USA), Nortel Networks (Canada) and the Government of Ontario (Canada). He received his Ph.D. degree in Computer Science from Carleton University, in Ottawa, Canada. He has won eight research paper awards in different conferences. He was awarded the IEEE ComSoc Asia Pacific Outstanding Young Researcher Award at IEEE GLOBECOM 2012. He was also awarded the Canadian Governments prestigious NSERC Post Doctoral Fellowship and the Humboldt Research Fellowship in Germany.