Computer Communications 194 (2022) 378–386

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

A CPS based social distancing measuring model using Edge and Fog computing

Manash Kumar Mondal [a], Riman Mandal [a,*], Sourav Banerjee [b], Utpal Biswas [a], Pushpita Chatterjee [c], Waleed Alnumay [d]

[a] Department of Computer Science and Engineering, University of Kalyani, Kalyani, Nadia 741235, India
[b] Department of Computer Science and Engineering, Kalyani Govt. Engg. College, Kalyani, Nadia, India
[c] Tennessee State University, USA
[d] Computer Science Department, King Saud University, Saudi Arabia

ARTICLE INFO

Keywords:
Cyber–physical system
Social distancing
Covid-19
Fog computing
Edge computing
IoT

ABSTRACT

Cyber–physical system (CPS) is one of the leading topics for research in academic and industry fields. CPS is an integrated system built with a collection of computation, communication, control, and physical elements to solve real-life problems. Lots of research is going on CPS, but in today's point of view, the Covid-19 is one of the most relevant. Nowadays, Covid-19 has become a headache in our society. Social or physical distancing is one of the most useful non-pharmaceutical interventions (NPI) to minimize virus infections. The regular lifestyle of every human being has been changing rapidly. A contactless lifestyle is becoming a necessity day by day. Society is gradually dependent upon smart technological devices for a contactless lifestyle. In the new-normal lifestyle, many new technologies have been introduced. The government also makes some restrictions on human transmission. However, maintaining social distancing is one of the main challenges of our society. There is no such model that effectively helps people to maintain physical distancing. This paper highlights a framework that will guide maintaining physical distance in a social gathering. The proposed CPS-based model is entirely deployed on edge and fog computing architecture. The proposed model calculates the distance among all paired edge devices owned by human beings and informs the user whether the location is safe or not. This fog and edge-based model improves the latency and network usage compared to the cloud computing model.

1. Introduction

Due to the potential significance of the Cyber–Physical System (CPS), it is becoming a current research trend in academia, industry, and government [1]. National Science Foundation (NSF) has granted a considerable amount of funds for the research on CPS [2]. Few prominent institutes and universities like the University of Memphis, UC Berkeley, University of Michigan, Vanderbilt University, University of Notre Dame, University of Maryland, and General Motors Research and Development Center have worked on this research area since 2008 [3]. CPS consists of two functional components; the first is that cutting-edge connectivity ensures the real-time data transfer between the physical system and cyberspace. The second one is intelligent data management, analytical skills, and computational capability of the cyberspace system [4]. Because of these two great functionalities, many applications have been developed based on CPS. In this article, the following model has entirely supported CPS properties.

Covid-19 is a newly identified coronavirus that causes an infectious illness. Most persons infected with the coronavirus will have mild to moderate respiratory symptoms and recover without needing particular therapy. This virus, tentatively named 2019-nCoV, was discovered in people exposed to seafood or wet market in Wuhan, China. The Chinese public health, clinical, and scientific sectors responded quickly, allowing for early detection of the clinical illness and knowledge of the infection's epidemiology [5]. Governments implemented several non-pharmaceutical interventions (NPI) to minimize the Covid-19 disease. Among NPIs, are isolation at home, voluntary home quarantine, the social distancing of the people over 70 years old, the social distancing of the entire population, and closure of educational institutions [6]. Therefore, the social distancing of one of the best life-saving practices during the Covid-19 pandemic.

During the Covid-19 pandemic, the big challenge is to maintain social distancing. Many researchers have shown that effective social

* Corresponding author.
E-mail addresses: manashkumarmondal@gmail.com (M.K. Mondal), mandal.riman@gmail.com (R. Mandal), mr.sourav.banerjee@ieee.org (S. Banerjee), utpal0172@gmail.com (U. Biswas), pchatter@tnstate.edu (P. Chatterjee), wnumay@ksu.edu.sa (W. Alnumay).

https://doi.org/10.1016/j.comcom.2022.07.029
Received 21 November 2021; Received in revised form 21 June 2022; Accepted 20 July 2022
Available online 26 July 2022
0140-3664/

Nomenclature

CAN	Campus Area Networks
CDMA	Code-Division Multiple Access
CoAP	Constrained Application Protocol
C-IoT	Cloud-centric Internet of Things
CPS	Cyber–Physical System
DDS	Data Distribution Service
EPC	Evolved Packet Core
FCFS	First Come First Serve
GPS	Global Positioning System
GSM	Global System for Mobile communication
I2O	Internet-to-Orbit gateway
JML	Java Modeling Language
LAN	Local Area Network
LTE	Long-term evaluation
MQTT	Message Queuing Telemetry Transport
NPI	Non-Pharmaceutical Interventions
QoE	Quality of Experience
QoS	Quality of Service
REST	Representational State Transfer
UMTS	Universal Mobile Telecommunications Service
VM	Virtual Machine
WAN	Wide Area Network
XMPP	Extensible Messaging and Presence Protocol

distancing can reduce the Covid infection rate. Wearing masks, sanitization, cleaning hands with soap, social distancing, etc., significantly impacts infection in society. Human society will be beneficial if such technology successfully monitors the physical distancing using mobile like IoT devices. This article proposes a model based on fog computing architecture that monitors social distancing. Typically, a Cloud-centric Internet of Things (C-IoT) system architecture is built for providing the services of heterogeneous physical objects or things. It commonly works as web-based client–server technology. Embedded systems, middleware, and cloud services are the three key technologies that engage with IoT Systems. The cloud system offers the required storage, control strategies, and processing. Interconnection between Embedded systems front end devices and the cloud has been done by middleware. Finally, an embedded system delivers intelligence to front-end devices. A few limitations in a cloud-based IoT system are the main reason for moving towards fog computing. The most common reasons are Bandwidth, Security, Latency, Resource-bound, and Uninterrupted [7,8]. This article proposed a model for edge and fog computing environments that improves latency and network usage compared to cloud computing environments.

CPS is one of the leading topics for research in academic and industry fields. CPS is an integrated system built with a collection of computation, communication, control, and physical elements to solve real-life problems. Lots of research is going on CPS, but in today's point of view, the Covid-19 is one of the most relevant. Nowadays, Covid-19 has become a headache in our society. Social or physical distancing is one of the most useful non-pharmaceutical interventions (NPI) to minimize virus infections. The regular lifestyle of every human being has been changing rapidly. A contactless lifestyle is becoming a necessity day by day. Society is gradually dependent upon smart technological devices for a contactless lifestyle. In the new-normal lifestyle, many new technologies have been introduced. The government also makes some restrictions on human transmission. However, maintaining social distancing is one of the main challenges of our society. There is no such model that effectively helps people to maintain physical distancing. This paper highlights a framework that will guide maintaining physical distance in a social gathering. The proposed CPS-based model is entirely deployed on edge and fog computing architecture. The proposed model calculates the distance between all paired edge devices owned by human beings and informs the user whether the location is safe or not. This fog and edge-based model improves the latency and network usage compared to the cloud computing model.

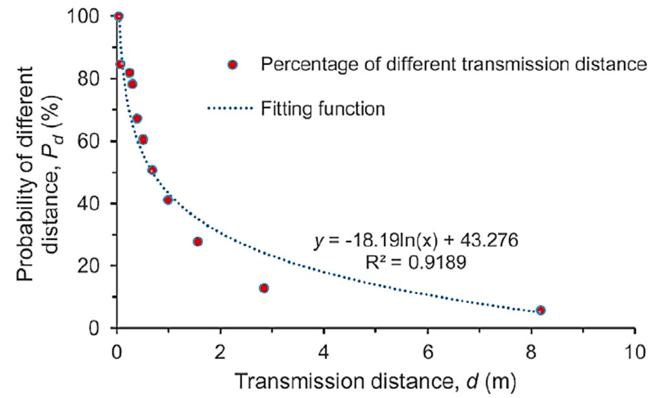

Fig. 1. The relationship between droplet transmission distance and its exposure probability [10].

The rest of the article is arranged in the following manner. Section 2 describes the literature survey of existing works. The proposed system model is presented in Section 3. Section 4 depicts the result analysis and performance evaluation of the proposed model. Finally, conclusion and future scopes are discussed in Section 5.

2. Literature survey:

Social distancing is one of the most important NPIs. Direct interaction among humans is avoided by social distance, as is the possible cross-transmission of virus-carrying droplets from human breathing. Several researchers worked on that. Sun et al. [9] developed a social distancing index P_d based on exhaled droplet circulation and transmission. They introduced the social distancing index by the inspiration of the Wells–Riley model. The researchers suggested a safe social distance of 1.6 to 3 m, entirely dependent on the aerosol transmission of exhaled large droplets while talking. Many researchers suspected that the droplets with small sizes in the nucleus form transmit more distances than regular sizes as airborne.

In the article [11], the researchers describe the impact of social distancing. As per their observations, the daily growth rate reduces up to 5.39% after two to five days, 6.87% after eight to ten days, 8.196% after thirteen to fifteen days, and 9.89% after twenty days. Keskinocak et al. [12] examined the impact of shelter-in-place that provides impressive results in Covid infection rate. Fig. 1 describes the relationship between distance and exposure probability. The X-axis demonstrates the distance in meters; the y-axis represents the probability of infection over distances.

CPS is a physical and engineered system that is disciplined, monitored, coordinated, and unified by a computing and communication module. It provides a coupling of physical and cyberspace. It acts as a bridge between the nan world and wide-area systems [13]. Gunes et al. [14] described a review of related work, focusing on the origins of CPS, its connections to various fields of study, common principles, and practical applications. Lee et al. [15] demonstrate that it can practically realize the deterministic models for distributed cyber–physical systems. Rajkumar et al. [16] focus on tightly integrated computing, communication, management, and physical parts that concentrate on intricate interdependencies and interaction between cyberspace and the physical world. In article [17], a genetic algorithm threshold-based CPS technology has been suggested to protect data during data mining techniques.

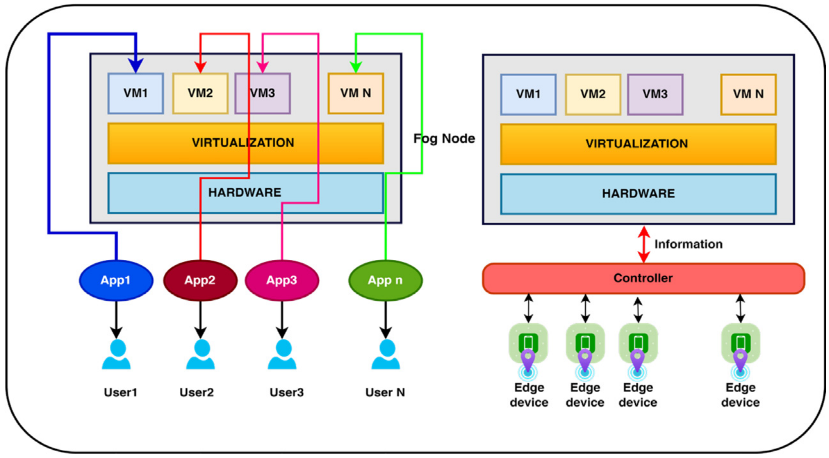

Fig. 2. Overview of a Fog node.

It also showed that the method results provided the same performance and improved security and privacy. Wu et al. [18] proposed an ant colony optimization-based model to secure sensitive and confidential information. Their method produces effective results in terms of secureness. A compact genetic algorithm-based mining model was proposed to reduce the mining progress in supermarket engineering. They used a clustering model to divide the transactions based on correlations that can accurately predict the pattern [19]. In article [20], the researcher proposed a Closed High-utility itemset mining (CHUIM) architecture for distributed databases. Applying a generic composite model they tried to achieve perfectness and correctness. After that, a map-reduce framework has deployed for taken care of such big data for mining.

2.1. General architecture of edge, fog, and cloud computing

Generally, cloud computing follows distributed system paradigms. Thousands of nodes have spread over the world. The computing node is placed within 200 meters of the devices in fog computing. In a word, the processing is done below the cloud and near the edge devices. Edge computing processes the data at the end devices themselves. A general combined architecture is discussed below in Fig. 3.

(a) *Sensor:* A device that detects the physical properties of the environment and converts them into data. A sensor detects an event or change in the environment. Generally, the data generated by sensors are sent to other devices for computation. Many sensors are available in the IoT field. Here we consider location sensors that generate location data in latitude and longitude format. Example: GPS16X-HVS.

(b) *Actuator:* Sensor generated data sent to any computing device for processing. After processing, the results are sent to an actuator. The actuator is a device that acts on the environment based on results. Sometimes a monitor or display device acts as an actuator.

(c) *Fog node:* Fog nodes are the local physical computing resource that offers various services on the deployment of fog computing. It consists of at least one processing element associated with edge devices. The fog node is the nearest computing powerhouse of edge devices [8]. Sometimes the edge device uploads the data to the cloud rather than fog. Fog node contains several properties that efficiently monitor and process the data from edge devices. Fog nodes can be deployed in various locations depending on the needs. Sometimes fog node works like a gateway, an intermediate computer node. It also works as a network device such as a router. Fig. 2 describes the internal working functionality of a fog node. Generally, a fog node uses bare-metal architecture. Virtual machines are directly deployed on the hardware. Virtualization is the upper layer of hardware. This layer provides the required resources to deploy a VM on user request. The controller decides which tasks are to be submitted in which VM.

(d) *Local data center:* The data center is a giant warehouse composed of storage and processing capabilities that offer various services to users. Big organizations and businesses use that resource to process, storage, and analysis purpose. Local data centers generally provide services to a limited number of users. For example, a local data center can serve the user of a city, or a factory, etc. It has certain limitations as if it is not as powerful as cloud.

(e) *Gateway:* Gateway is a telecommunication entity. It is a piece of hardware or software that allows data transmission from one network to another. There is a difference between gateway and router or switches. Gateways use more than one protocol to communicate among numerous networks. Various types of gateways are available in telecommunications. The most common gateways are the Internet-to-orbit gateway (I2O), cloud storage gateway, and IoT gateway. IoT gateway is most important in these scenarios. IoT gateway act like a protocol converter between the field of IoT devices and the cloud. It also supports real-time control and offline support for the field devices. Two different architectures are used for the data exchange protocol. The bus-based protocol consists of DDS, REST, and XMPP. On the other hand, the broker-based protocol includes AMQP, CoAP, MQTT, and JMI. Sometimes the protocols are classified as message-centric and data-centric.

(f) *Connectively:* Connectivity is one of the necessary factors in this architecture. Without proper connection establishment, nothing can happen. A wired or wireless medium generally attaches devices. Edge devices need to be well associated with some fog nodes uninterruptedly. Connection failure is unexpected in the fog architecture. One of the main reasons to use fog architecture is that the fog nodes are generally deployed closed to the edge devices. Different kinds of fog devices are linked with distinct fog nodes with various networking devices in the middle.

(g) *Base station:* A microcell is a mobile phone network cell served by a low-power cellular base station (tower) that covers a small area, such as a mall, a hotel, or a transit hub. Microcells can cover a range of lesser than 2 km. However, the base station (mobile tower) can serve up to 35 km. Whereas Pico cell 200 m range. Femtocell works within 10 m radius. Microcells generally provide 2G connectivity [21]. Pico cells are also available in GSM, CDMA, UMTS, and LTE. Small cells are low-powered cellular radio access nodes within 10 m to a few km that operate in licensed and unlicensed spectrum.

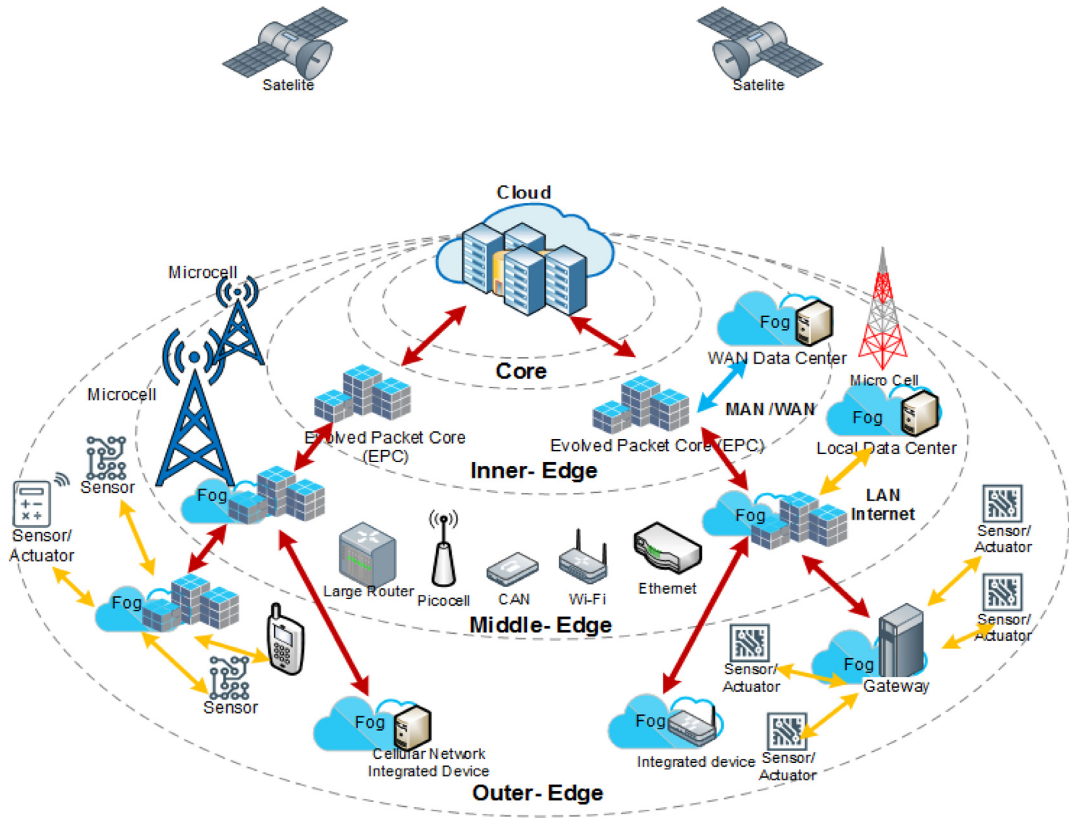

Fig. 3. Complete architecture of edge, Fog and Cloud computing.

(h) *Inner edge:* The inner border is also called near-the-edge [22], consists of worldwide, state-wide, country-wide regional wide area network (WAN) enterprises, internet service provider (ISP), evolved packet core (EPC), and metropolitan area network. Local networks are connected with the global network using the infrastructure of the inner edge. Nowadays, to achieve the quality of experience (QoE), the service providers use geo-distributed caching and processing mechanisms at the data center.

(i) *Middle edge:* The middle edge generally contains two types of networks — local area network and cellular network. Further LANs can be used as campus area networks (CANs), Ethernets, and wireless LANs (WLANs). Femtocell, picocell, microcell, and microcell are belonged to cellular networks [23]. Cellular networks are mainly classified based on services and the range of services

(j) *Outer edge:* The outer edge is the bottom layer of that architecture. This layer is also called a far-edge, extreme-edge, and sometimes mist [24–26]. Mainly three types of devices belong to this layer — integrated devices, IP gateway devices, and constraint devices. The sensor and actuator are those devices generally controlled by a microcontroller. These devices have comparatively low processing power and memory storage. These types of devices belong to the constraint device.

2.2. Application model deployed in iFogSim

An application model interface is necessary that provides an interface to the user. That application model can manage the functionality of the sensor–actuator also. Fig. 4 describes the application model deployed in the iFogSim simulator. Every client module consists of four modules data cleaning, distance calculations, safe location calculation, and location control.

The sensor generates raw data (location) and sends it to the data-cleaning module to clean up the raw data and make it proper location data format. The data cleaning module is deployed inside the edge device itself. The edge device cleans the data and sends it to the fog layer for further processing. The fog devices process the data into two different modules in the fog layer. The distance calculation module calculates the distance among all the edge devices. The safe location calculation module calculates each device's potential safe locations and sends the data to the location control module. The location-control module is also deployed in the edge device itself. This module recommended the user the safest place to maintain a threshold distance.

3. Proposed system model

3.1. Problem formulation

Over CPS conceptualization, many researchers have developed several theoretical and practical models for Smart Manufacturing, Air Transportation, Emergency Response, Intelligent Transportation, Critical Infrastructure, Health Care and Medicine, and Robotic Service. Some researchers also suggested a few models of CPS to solve real-life problems. In this article, we deal with a Covid-19 using CPS. During Covid-19 pandemic, humans are getting affected by the virus nearby. We are experiencing a new normal lifestyle during Covid and post-Covid. Hotels, bars, restaurants, theaters, shopping complexes, and indoor games are open with few Covid restrictions [27]. The study [10] says that the Covid virus can fly up to 8 m. There is no such technical model available in society that can guide us to maintaining social distancing up to the experts' prescribed distance. Measuring social distancing is not easy at all using mobile-like devices. The required computational capability is not available in mobile-like devices for continuous monitoring. If thousands of people gathered in a place that will generate a large amount of data in every instance. The processing of that large volume of data requires massive computational capability.

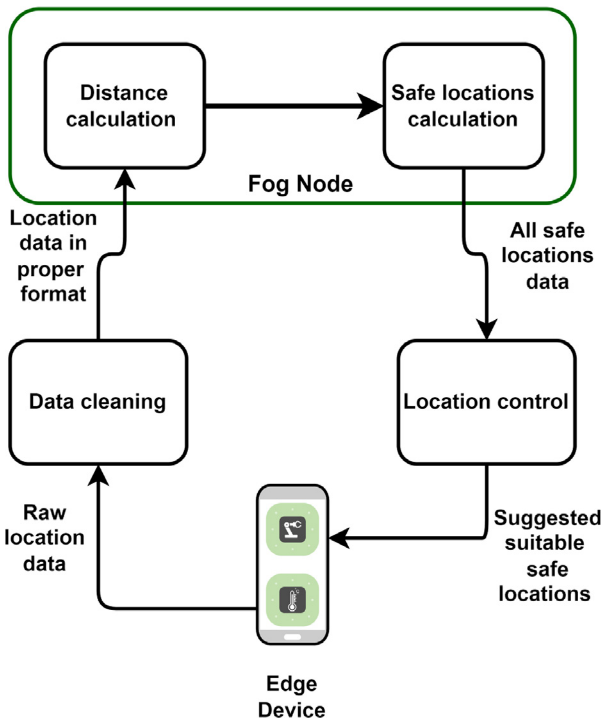

Fig. 4. Application model in iFogSim.

To fulfill this requirement, we must consider a technology that can handle it efficiently. Generally, the cloud was the only reliable solution for providing these services. However, there are a few drawbacks; their cloud is not suitable for such latency-sensitive devices.

3.2. Proposed model

This article proposes a social distancing measuring model that is entirely deployed in fog and edge computing architecture, and it reduces the latency and network usage compared to the cloud. Cloud provides several services to users like computation, storage, infrastructure, software, etc. Cloud computing has some limitations, like latency, bandwidth, and operational cost. Fog computing is one of the alternative solutions that overcome the drawbacks of the cloud. It can also deal with that large amount of data with minimal bandwidth and latency of nearly zero. In our article, many sensors and actuators are there at the bottom layer. This bottom layer is the edge layer. Among those components, the sensor takes the data from the environment. The actuator makes action on the environment based on the results processed by the system. A GPS module received accurate geographic location in latitude and longitude values. Thousands of edge devices are consciously generating location data at the same time. The edge device is not so capable of doing such a substantial computational itself. Therefore, it needs to send the data to a place that can provide enough infrastructure. That infrastructure was provided by fog. After processing this data, calculate the distance between each edge device pair in fog and send the results back to the edge devices. Each of the GPS-enabled devices is an edge device (see Fig. 6).

In this model, the devices are uninterruptedly linked to some satellites for location data generation. Among thirty-one satellites minimum of four satellites are required to calculate a particular location on earth. It should also connect the GPS-enabled edge devices to the internet for transferring data. A GPS consists of three major things receiver, satellite, and ground station. Firstly receiver receives the signal from the satellites. A minimum of four satellites are required to calculate the location on the earth's surface. All four satellites communicate with a ground station with a known location (lati, long.) Three satellites provide information on the three-dimensional space: the x-axis, y-axis, and z-axis. The fourth satellite gives information about time. Concerning the location of the ground station and the time required to receive the signal by receivers, together calculate the receiver device's location [28] (see Fig. 5). The GPS modules generate the current position in latitude and longitude format like at1(lati, long), at2(lati, long), etc. For example, a device can generate location data like (22.9880962, 88.4482013). So N number of devices will generate N set of location data in every instance of time. Therefore, for a single instance of a time tk, N device will generate data like:

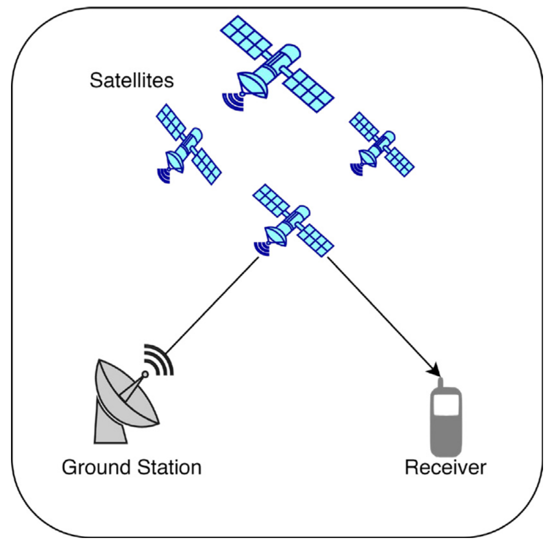

Fig. 5. Satellite communication of an edge device.

$$\mathrm{a}_{tk}(\mathrm{lati},\mathrm{long}),\mathrm{b}_{tk}(\mathrm{lati},\mathrm{long}),\mathrm{c}_{tk}(\mathrm{lati},\mathrm{long}),\ldots,$$
$$\mathrm{n}-1_{tk}(\mathrm{lati},\mathrm{long}),\mathrm{n}_{tk}(\mathrm{lati},\mathrm{long}).$$

That much amount of data needs to send to the fog layer at any instant. The number of VMs (virtual machines) created in the fog equals users (edge devices) in the environment. For every instance of a time, the same number of locations (latitude, longitude) send to fog. In every virtual machine, calculate the distance among all user locations.

The proposed model consists of several modules. The modules are described below:

- *Edge device with GPS module:* The edge device is the bottom layer of this architecture. These devices are associated with a GPS module and little processing microcontroller-controlled devices. The primary purpose of this device is to calculate the device's location. Generally, a GPS module generates raw data; it needs to be clean and obtained in a proper format before processing this data. After cleaning raw data, it is ready for further processing in the next layer. The GPS module generates raw data at five-millisecond intervals. An actuator module is also installed inside the edge device and then provides feedback to the user after getting the result from the fog node.
- *The fog node:* The fog layer plays between the edge and cloud layers in the middle. This layer collects the edge layer's data and processes it inside the fog node. The objective of this fog layer is to process the data without a significant delay. The cloud faces a considerable delay as compared to the fog. Two modules have been deployed in the fog node: distance calculation and safe location calculation. After calculating the secure locations, the data send to the edge device for further processing.

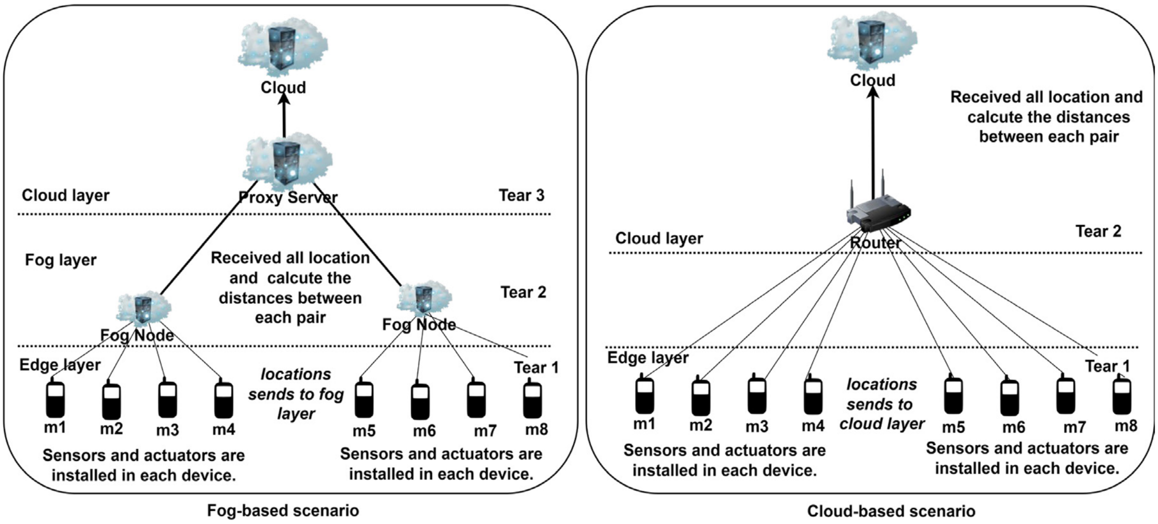

Fig. 6. (a) Topology of Fog-based model in iFogSim (b) Topology of Cloud-based model in iFogSim.

- *The cloud datacenter:* After processing all of the data and getting feedback from the fog layer, we need to store the data. The cloud layer uses for storing purposes in this architecture. All the communication between the fog nodes and the cloud happens via a proxy server. The fog node transfers the data to the cloud after some scheduled time intervals.
- *Application:* An application program by which the user can control the sensor and actuator. Using the user can on and off or modify the working functionality of the edge devices. It is used for monitoring the sensor and actuator also. For a mobile-like edge-device, the GPS module implies a sensor, and the display unit act as an actuator. A fog-based application deploys in Fig. 4 using iFogSim. The addition of one edge device on that particular model is quite simple. The devices should be connected by internet connection and send the data to the fog from the application module. The fog broker is responsible for the new virtual machine deployed on which host. Generally, FCFS policy is followed. Load balancing is also another overhead for resource allocation cases. A new location data (latitude, longitude) is appended to the old size data set. So the new data set contains one more data than the previous. The new location device is assigned to a new virtual machine on a host that calculates the distances with all device's location data sets. Deletion of a device in this model is also simple. The edge sensor, which generates location data, can be disconnected. That device will not send anything. The new data set contain one less number location data than the previous one. The virtual machine will also terminate at the host, which was previously deployed for that deleted device. The new data set will be taken care of by the broker. From the application program, the user can also disconnect a device manually.

3.3. Algorithm for distance calculation

A data set is generated by collecting all sensor location data. The same data set is sent to the fog node for processing. The fog broker deploys the same number of virtual machines (VMs) as the number of edge devices (users). The mapping between the number of edge devices and VMs is one to one. The VMs are responsible for all types of processing-related activities. The researchers introduce a few algorithms for calculating the actual distance between two locations on the earth's surface. In this article, the following algorithm is used. This algorithm calculates the distance between two points using location data [29,30].

Algorithm

Input: *Two location in (lati, and long.)*
Output: *Distance between two points*

```
Step1: R=6373.00
Step2: lat1= radians(latitude ), lon1= radians(latitude )
Step3: lat2= radians(latitude ), lon2= radians(latitude )
Step4: dlon(lon2-lon1), dlat(lat2-lat1)
Step5: a = sin(dlat / 2)**2 + cos(lat1) * cos(lat2) * sin(dlon / 2)**2
Step6: c = 2 * atan2(sqrt(a), sqrt(1 - a))
Step7: distance = R * c* 1000
Step8: print ("Result:", distance)
Step9: if distance <=7:
Step10:print ( "Unsafe Feedback message")
Step11:print (" Give instruction")
Step12: else
Step13:print ("Safe Feedback message")
```

3.4. Workflow diagram

A Cyber–physical system integrates an embedded system, networks, and communications. It works like the backbone of modern IoT systems. Along with this, fog and edge computing also play a significant role. Clients/users are seamlessly connected by GPS module-enabled embedded systems with the satellites in this model. Every user generates some information. That information is nothing but the user's geographic location (lati. long). After that, the user submits the data to the fog broker in iFogSim Simulator. After aggregating data from all the users, the fog broker sent the data to the fog node, called fog device in simulator. The number of virtual machines called AppModule in iFogSim is the same as many clients. Distance between each pair of the user is computed in the VMs. This entire thing happens between start-simulation and stop-simulation methods. The result is generated and sent back to users.

The proposed model consists of an edge device that generates raw location data using a sensor. The fog node also communicates with each other using internetworks. Data is cleaned at the edge device and then sent to a fog node for processing. In this model, data processing refers to the distance calculation between edge devices. All the fog nodes coordinate before sending the feedback to the fog device. The outcomes and location data are stored at the cloud data center for future use (see Fig. 7).

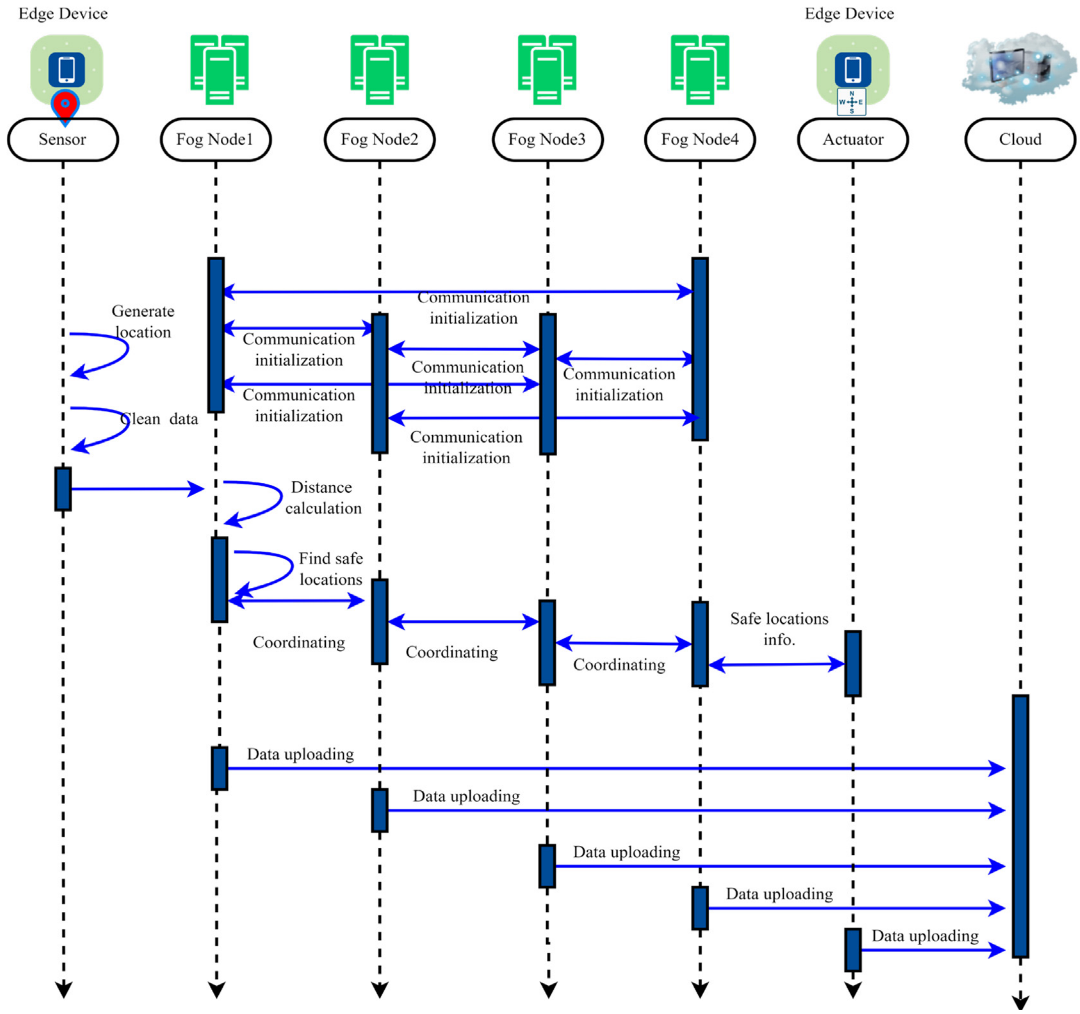

Fig. 7. Workflow diagram of proposed model.

4. Result analysis and performance evaluation:

4.1. Experimental setup:

This model is deployed in a most popular fog-computing simulator called iFogSim [31,32]. The iFogSim is an extension of another extensively used cloud simulator environment, CloudSim [33]. An efficient algorithm is used for distance calculation between two devices. For simulation, we deploy two models, one in a cloud environment and another in fog. Heterogeneous system configurations need to be considered and simulated to get unbiased outcomes. Config-1, Config-2, Config-3, Config-4, and Config-5 are the five configurations based on number of edge devices connected to the system. Table 3 shows the configurations of devices for simulations. The fog-based model follows three-tier architecture, whereas the cloud-based model has only two — the proper table represents the equipment used in each tier.

Table 1 illustrates the entire system configuration describing the CPU length, RAM size, Uplink bandwidth, Downlink bandwidth, Deployment level of fog, proxy server, and cloud. Additionally, this table also describes power consumptions and processing capability matrices. Each edge device generates the location data in five-millisecond intervals.

Table 2 illustrates a few crucial parameters regarding the cloud-based model. The cloud-based model has only two tires, the edge, and cloud tiers. In this model, the cloud receives the data from the edge devices via some routers. Table 2 describes the system configurations of cloud and router only. It includes the CPU length, RAM size, Uplink bandwidth, Downlink bandwidth, Level of deployment, RatePerMIPS, Busy power, Idle power, etc.

Table 1
System configurations fog-based scenario.

Parameter	Cloud	Proxy Server	Fog
Level	0	1	2
CPU length (in MIPS)	44800	2800	2800
RAM (MB)	40000	2800	4000
Uplink bandwidth (in MB)	100.00	10000.00	10000.00
Downlink bandwidth (in MB)	10000.00	10000.00	10000.00
RatePerMIPS	0.01	0.0	0.0
Busy power (in Watt)	16 × 103	107.339	107.339
Idle power (in Watt)	16 × 83.25	83.43	83.43

4.2. Performance evaluation

The result of the simulation is described in Table 3. The first column represents configuration details. Column 2 represents the number of edge devices associated with experiments. The latency parameter of each configuration in the fog scenario is depicted in column 3. Similarly, column 4 describes the latency details in the cloud scenario. Columns 5 and 6 represent the network usage in fog and cloud scenarios, respectively.

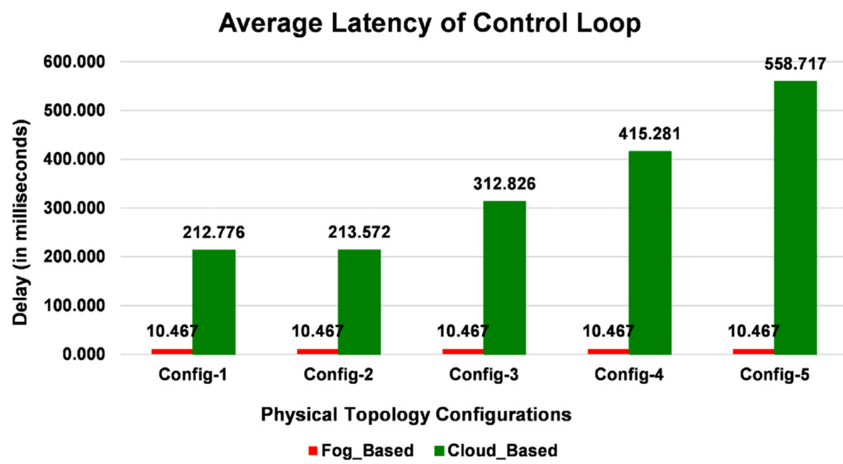

Fig. 8. Comparison of latency based matrices.

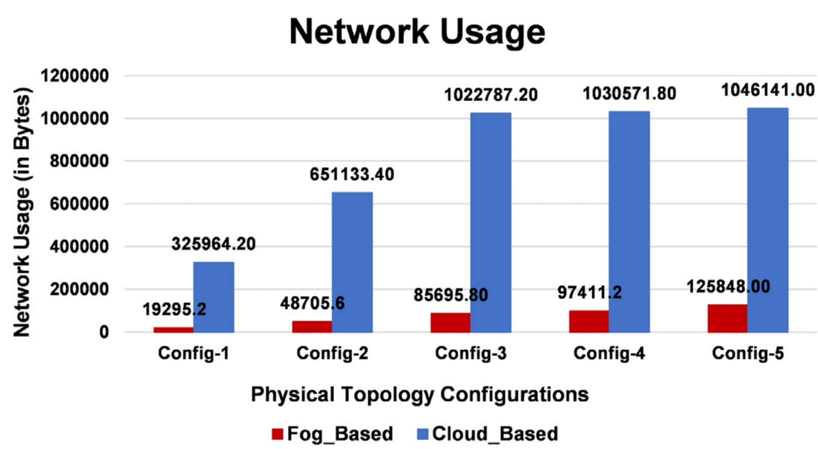

Fig. 9. Comparison of network usages based matrices.

Table 2
System configurations for cloud-based scenario.

Parameter	Cloud	Router
Level	0	1
CPU length (in MIPS)	44800	2800
RAM (MB)	40000	4000
Uplink bandwidth (in MB)	100.00	10000.00
Downlink bandwidth (in MB)	10000.00	10000.00
RatePerMIPS	0.01	0.0
Busy power (in Watt)	16 × 103	107.339
Idle power (in Watt)	16 × 83.25	83.43

4.2.1. Application loop delay (latency):

Application loop delay or latency is the time required for two consecutive processing of locations for one device. Eq. (1) represents the calculations of application loop delay (latency) as follows [32].

$$Latency = \delta + \gamma + \varepsilon + \rho \tag{1}$$

Where δ represents the time taken for data cleaning, γ represents the time taken for distance calculation, ε denotes the time taken for calculating all the location, and ρ represents the time taken by location control module. Fig. 8 illustrates the delay in milliseconds for all the configurations. The red bar represents the fog-based results, whereas the green bar denotes the results of cloud-based scenarios.

4.2.2. Total network usage

If the number of fog devices increases, network usage will also increase. However, the network load on fog is lower than the network load on the cloud. The result clearly shows that the fog-based system improves the outcome. Network usage can calculate as [33] in Eq. (2), and Fig. 9 clearly illustrate it. The x-axis represents the network usage, and the y-axis denotes the topology configurations of systems.

$$Network\ usage = Latency \times \zeta \tag{2}$$

Where ζ represent tupleNWSize in iFogSim.

5. Conclusion

The prime goal of the current study was to reduce latency and network usage while achieving social distance. This CPS-based proposed model's topology has been fully deployed and traced on the Java-based simulator called iFogSim. For the experiments, five different sets of system configurations have been considered. Each system configuration is deployed and simulated in both fog-edge and cloud scenarios. These findings have significant implications for understanding how the fog and edge-based model results show substantial improvements in latency and network usage over the cloud model. Since the study was limited to simulation, it was impossible to address real-world data. This CPS-inspired model can extend in the future by deploying in real-world scenarios. The research is limited by the lack of information on data security. User data security is one of the significant challenges that it can address. The researcher has not applied any encryption algorithm to the location data of every end-user while sending it to the fog layer. The study should be repeated using potential research directions like resource management, energy consumption, efficient task allocation policies, etc.

Table 3
Simulation results.

Configurations	No of devices	Latency (Fog) (in Milliseconds)	Latency (Cloud) (in Milliseconds)	Network Usage (Fog) (in Bytes)	Network Usage (Cloud) (in Bytes)
Config-1	8	10.467	212.776	19295.20	325964.20
Config-2	16	10.467	213.572	48705.60	651133.40
Config-3	28	10.467	312.826	85695.80	1022787.20
Config-4	32	10.467	415.281	97411.20	1030571.80
Config-5	40	10.467	558.717	125848.00	1046141.00

CRediT authorship contribution statement

Manash Kumar Mondal: Conceptualization, Methodology, Software. **Riman Mandal:** Conceptualization, Software, Writing – original draft. **Sourav Banerjee:** Resources, Investigation, Writing – review & editing. **Utpal Biswas:** Supervision. **Pushpita Chatterjee:** Validation, Writing – review & editing. **Waleed Alnumay:** Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The work of Dr. Alnumay is funded by Researchers Supporting Project number (RSP-2021/250), King Saud University, Riyadh, Saudi Arabia.

References

[1] X. Yu, Y. Xue, Smart grids: A cyber–physical systems perspective, Proc. IEEE 104 (5) (2016) 1058–1070, http://dx.doi.org/10.1109/JPROC.2015.2503119.
[2] J. Shi, J. Wan, H. Yan, H. Suo, A survey of cyber–physical systems, in: 2011 Int. Conf. Wireless Communications and Signal Processing, WCSP, IEEE, 2011, pp. 1–6.
[3] J. Sprinkle, U. Arizona, S. Sastry, CHESS: Building a cyber–physical agenda on soli foundations, 2008, Presentation Report.
[4] J. Lee, B. Bagheri, H.-A. Kao, A cyber–physical systems architecture for industry4.0-based manufacturing systems, Manuf. Lett. 3 (2015) 18–23.
[5] S. Perlman, Another decade, another coronavirus, New England J. Med. (2020) 760–762.
[6] N. Ferguson, D. Laydon, G. Nedjati Gilani, N. Imai, K. Ainslie, M. Baguelin, et al., Report 9: Impact of non-pharmaceutical interventions (NPIs) to reduce COVID19 mortality and healthcare demand, 2020.
[7] M. Chiang, T. Zhang, Fog and IoT: An overview of research opportunities, IEEE Internet Things J. 3 (6) (2016) 854–864.
[8] IEEE Standard Association, FOG - Fog computing and networking architecture framework, 2018, [Online] http://standards.ieee.org/develop/wg/FOG.html. (Accessed 2 April 2018).
[9] C. Sun, Z. Zhai, The efficacy of social distance and ventilation effectiveness in preventing COVID-19 transmission, Sustainable Cities Soc. 62 (2020) 102390.
[10] C. Sun, Z. Zhai, The efficacy of social distance and ventilation effectiveness in preventing COVID-19 transmission, Sustainable Cities Soc. 62 (2020) 102390.
[11] C. Courtemanche, J. Garuccio, A. Le, J. Pinkston, A. Yelowitz, Strong social distancing measures in the United States reduced the COVID-19 growth rate: Study evaluates the impact of social distancing measures on the growth rate of confirmed COVID-19 cases across the United States, Health Affairs 39 (7) (2020) 1237–1246.
[12] P. Keskinocak, B.E. Oruc, A. Baxter, J. Asplund, N. Serban, The impact of social distancing on COVID19 spread: State of Georgia case study, PLoS One 15 (10) (2020) e0239798.
[13] R.R. Rajkumar, I. Lee, L. Sha, J. Stankovic, Cyber-physical systems: The next computing revolution, in: Proc. 47th Design Automation Conf., ACM, 2010, pp. 731–736.
[14] V. Gunes, S. Peter, T. Givargis, F. Vahid, A survey on concepts, applications, and challenges in cyber–physical systems, KSII Trans. Internet Inf. Syst. 8 (12) (2014) 4242–4268, http://dx.doi.org/10.3837/tiis.2014.12.001.
[15] E.A. Lee, The past, present and future of cyber–physical systems: a focus on models, Sensors (Switzerland) 15 (3) (2015) 4837–4869, http://dx.doi.org/10.3390/s150304837.
[16] R.R. Rajkumar, I. Lee, L. Sha, J. Stankovic, Cyber-physical systems: The next computing revolution, in: Proc. 47th Design Automation Conf., ACM, 2010, pp. 731–736.
[17] J.M.T. Wu, G. Srivastava, A. Jolfaei, M. Pirouz, J.C.W. Lin, Security and privacy in shared HitLCPS using a GA-based multiple-threshold sanitization model, IEEE Trans. Emerg. Top. Comput. Intell. (2021).
[18] J.C.W. Lin, G. Srivastava, Y. Zhang, Y. Djenouri, M. Aloqaily, Privacy-preserving multiobjective sanitization model in 6G IoT environments, IEEE Internet Things J. 8 (7) (2020) 5340–5349.
[19] J.C.W. Lin, Y. Djenouri, G. Srivastava, U. Yun, P. Fournier-Viger, A predictive GA-based model for closed high-utility itemset mining, Appl. Soft Comput. 108 (2021) 107422.
[20] J.C.W. Lin, Y. Djenouri, G. Srivastava, Efficient closed high-utility pattern fusion model in large-scale databases, Inf. Fusion 76 (2021) 122–132.
[21] D. Lopez-Perez, I. Guvenc, G. de la Roche, M. Kountouris, T.Q. Quek, J. Zhang, Enhanced intercell interference coordination challenges in heterogeneous networks, IEEE Wirel. Commun. 18 (3) (2011) 22–30.
[22] H.P. Sajjad, K. Danniswara, A. Al-Shishtawy, V. Vlassov, Spanedge: Towards unifying stream processing over central and near-the-edge data centers, in: Proceedings of the IEEE/ACM Symposium on Edge Computing, SEC, IEEE, 2016, pp. 168–178.
[23] K. Tammemäe, A. Jantsch, A. Kuusik, J.-S. Preden, E. Õunapuu, Self-aware fog computing in private and secure spheres, in: Fog Computing in the Internet of Things, Springer International Publishing, 2018, pp. 71–99.
[24] J.S. Preden, K. Tammemae, A. Jantsch, M. Leier, A. Riid, E. Calis, The benefits of self-awareness and attention in fog and mist computing, Computer 48 (7) (2015) 37–45.
[25] M. Liyanage, C. Chang, S.N. Srirama, MePaaS: Mobile-embedded platform as a service for distributing fog computing to edge nodes, in: Proceedings of the 17th International Conference on Parallel and Distributed Computing, Applications and Technologies, PDCAT-16, Guangzhou, China, December 16–18, 2016, pp. 73–80.
[26] K. Tammemäe, A. Jantsch, A. Kuusik, J.-S. Preden, E. Õunapuu, Self-aware fog computing in private and secure spheres, in: Fog Computing in the Internet of Things, Springer International Publishing, 2018, pp. 71–99.
[27] M. Zajenkowski, P.K. Jonason, M. Leniarska, Z. Kozakiewicz, Who complies with the restrictions to reduce the spread of COVID-19?: Personality and perceptions of the COVID-19 situation, Pers. Individ. Differ. 166 (2020) 110199.
[28] R. Santerre, Impact of GPS satellite sky distribution.(18), 1991.
[29] James Inman, [1821]. Navigation and Nautical Astronomy: For the Use of British Seamen, third ed., W. Woodward, C. & J. Rivington, London, UK, 1835, Retrieved 2015-11-09. Fourth edition.
[30] Geometrical determination of the perpendicular to the meridian drawn by Jacques Cassini, Mem. Acad. R. Sci. Paris (1733) 406–416, in French.
[31] R. Mahmud, S. Pallewatta, M. Goudarzi, R. Buyya, IFogSim2: An extended iFogSim simulator for mobility, clustering, and microservice management in edge and fog computing environments, 2021, arXiv preprint arXiv:2109.05636.
[32] H. Gupta, A. Vahid Dastjerdi, S.K. Ghosh, R. Buyya, IFogSim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments, Software: Pract. Exp. 47 (9) (2017) 1275–1296.
[33] R.N. Calheiros, R. Ranjan, A. Beloglazov, C.A. De Rose, R. Buyya, Cloudsim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms, Softw. - Pract. Exp. 41 (1) (2011) 23–50.