Computers, Materials & Continua
DOI: 10.32604/cmc.2023.029340
Article

Tech Science Press

Hybrid Mobile Cloud Computing Architecture with Load Balancing for Healthcare Systems

Ahyoung Lee[1], Jui Mhatre[1], Rupak Kumar Das[2] and Min Hong[3,*]

[1]Department of Computer Science, Kennesaw State University, Marietta, GA, USA
[2]Department of Computer Science, University of Minnesota Duluth, Duluth, MN, USA
[3]Department of Computer Software Engineering, Soonchunhyang University, Asan, Korea
*Corresponding Author: Min Hong. Email: mhong@sch.ac.kr
Received: 02 March 2022; Accepted: 26 May 2022

Abstract: Healthcare is a fundamental part of every individual's life. The healthcare industry is developing very rapidly with the help of advanced technologies. Many researchers are trying to build cloud-based healthcare applications that can be accessed by healthcare professionals from their premises, as well as by patients from their mobile devices through communication interfaces. These systems promote reliable and remote interactions between patients and healthcare professionals. However, there are several limitations to these innovative cloud computing-based systems, namely network availability, latency, battery life and resource availability. We propose a hybrid mobile cloud computing (HMCC) architecture to address these challenges. Furthermore, we also evaluate the performance of heuristic and dynamic machine learning based task scheduling and load balancing algorithms on our proposed architecture. We compare them, to identify the strengths and weaknesses of each algorithm; and provide their comparative results, to show latency and energy consumption performance. Challenging issues for cloud-based healthcare systems are discussed in detail.

Keywords: Mobile cloud computing; hybrid mobile cloud computing; load balancing; healthcare solution

1 Introduction

Healthcare is a fundamental part of every individual's life. The healthcare industry is developing very rapidly with the help of advanced technologies. To ensure healthcare systems are more accessible to people, many researchers are trying to build different healthcare solutions. Nowadays, smartphones, as IoT mobile devices, are more capable of dealing with diverse types of applications to complete their tasks. Healthcare applications on mobile devices can exchange data through communication interfaces (e.g., application programming interfaces (APIs)) between patients/system users and healthcare service providers. However, the limitations of computing resources (e.g., CPU, storage, and processing power) in mobile devices mean that is not possible to run all application processes in these resource-constrained mobile devices. Thus, to overcome resource limitations, mobile devices integrated with

cloud paradigms as a mobile cloud computing architecture offer efficiency-enhancing usability of the mobile device. For example, users can access healthcare facilities and get the task outputs by offloading computation activities from mobile devices with their hardware limitations to cloud-based techniques. Therefore, in this paper, we propose a hybrid mobile cloud computing (HMCC) architecture for healthcare applications. The HMCC architecture contains one or more private clouds where patient's information can be stored and analyzed, and one or more public clouds for easy access to the healthcare system for patients. HMCC provides a workload balancing algorithm for the proper utilization of resources.

With the increase in population, healthcare systems are becoming major challenges in today's world. According to the World Health Organization (WHO) [1], at least half of the global population is unable to access essential health services, and 930 million people worldwide spend 10 percent of their income on healthcare for themselves or their families. Because conventional healthcare systems are very costly and time consuming, it is not always possible for residents of poor countries to get proper healthcare. Also due to poor transportation systems, people from underdeveloped or developing countries are unable to get quick treatment from their health centers.

These healthcare problems can be alleviated through the use of cloud computing techniques. Cloud computing has been widely revolutionized by incorporating computing technologies. Thus, using cloud computing provides the main benefits of: (1) enhancing the usability of existing IoT resources, (2) allowing users to access hardware components, such as storage or CPU, as well as software components, at any time from any location, (3) providing high-capacity networks, as well as low-cost computing and storage services, and (4) guaranteeing high-accuracy results, as well as requiring less human interaction.

According to cloud deployment models [2], there are four types of clouds, namely public clouds, private clouds, hybrid clouds, and community clouds. The public cloud infrastructure is designed to be available to the public or large industrial cloud service providers (e.g., Amazon Elastic Compute Cloud (EC2), Google Cloud, and Microsoft Azure) to sell cloud services. But in the public cloud, many different attacks happen easily as anyone has access to it. The private cloud infrastructure has the same performance as the public cloud, but it is operated solely for private organizations, and to provide cloud services only to their authorized users. The development of a private cloud may require inflated cost. The hybrid cloud infrastructure is a construction of more than one cloud (private, community, or public) depending on the purpose of an organization. It may require load-balancing solutions between clouds. However, there are many advantages of the hybrid cloud, such as flexibility, scalability, and reliability. For example, a healthcare system has patient data that may be extremely sensitive and private–these data are stored on private cloud servers, and the healthcare system can interconnect with applications on public clouds as a software service. The community cloud infrastructure is shared by several different organizations who share common concerns (e.g., security requirements, compliance considerations, and system policy).

Based on the National Institute of Standards and Technology (NIST) cloud computing reference architecture [2], this cloud model consists of three service models: Infrastructure as a Service (IaaS) for computing architecture including data storage, virtualization, server and network; Platform as a Service (PaaS) for supporting programming language execution environments, including operating system, web server and database; and Software as a Service (SaaS) for supporting on-demand services for users, such as such as Microsoft 365 and Adobe Creative Cloud. Fig. 1 illustrates the three-layer service of cloud computing architecture.

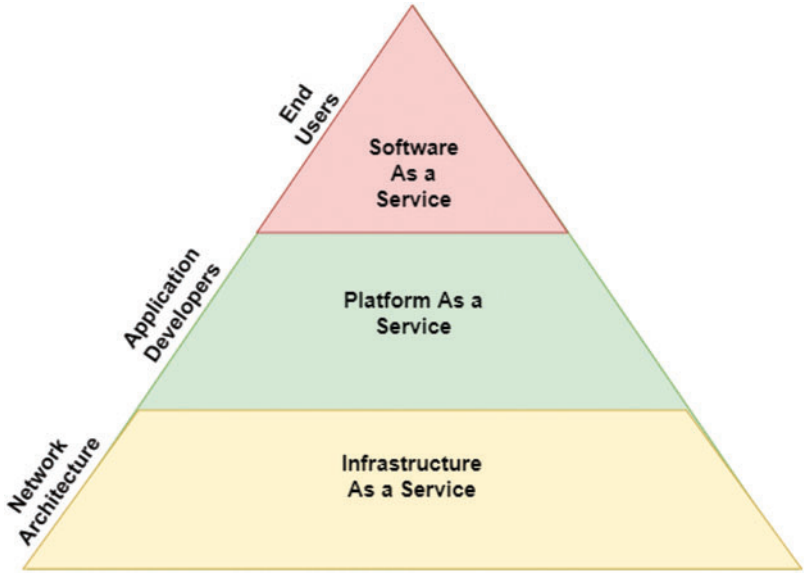


Figure 1: Overview of the service layering cloud computing architecture

However, cloud computing resources based on the cloud computing architecture in Fig. 1 are limited, and very costly. So, resource utilization is required to gain the maximum profit from cloud computing. Services, for example, healthcare systems, require real-time data analysis and solutions. They require a very quick response from cloud computing. In those cases, load balancing is highly beneficial to the cloud environment, where massive workloads can be equally distributed among different servers. Load balancers can determine workload, and can distribute the workloads among other servers. Load balancing provides a prominent level of service availability, and improves the response time. Without load balancing, some virtual servers might have zero traffic, while others become overloaded. Therefore, we describe why a load balancing algorithm is required for the improvement of service availability and response time of mobile cloud computing.

The remainder of this paper is organized as follows. Section 2 provides a detailed literature survey of existing healthcare systems with cloud computing approach, highlighting the need for mobile cloud computing in health care systems. Section 3 details the proposed architecture and load balancing algorithms; then Section 4 discusses the implementation environment and analyzes the results. Based on our implementation and literature survey, Section 5 highlights open challenges; Section 6 then concludes the paper.

2 Related Work

2.1 Healthcare Systems with Cloud Computing Approaches

Health care services incorporate recent advancements in technology. Various applications leveraging remote photoplethysmography techniques, such as the remote analysis of patients using video and web cameras, are widely used. 3D remote computed tomography (CT) is another remote technique used nowadays for imaging and health care automation [3]. Many health and fitness devices are available on the market, such as smart watches, diabetes testing kits, pedometers, smart mats, heart rate monitors, and smart baby monitors. Most of these devices are Internet of Things (IoT) devices, and have little in-memory and onboard computation power. But the amount of data generated

by these devices' accounts for big data. Complex machine learning algorithms need to be run to provide their results [4]. These devices are connected to mobile phones whose computation power is utilized to generate results. But even then, the problem of lack of resources persists. Further improvisation to resolve this issue is made by leveraging the power of edge computing and cloud computing [3−8]. Reference [3] proposed an architecture that has two applications, traffic offloading, and radio network information services. References [3,5] further highlighted the issue of data privacy in distributed computing introduced due to edge computing. Reference [9] proposed a user-centric secure edge computing architecture using blockchain technology to secure the users' data records during data distribution over edge servers. Latency is a major problem, and other problems in using edge computing are also highlighted, like data abstraction for transmission, lack of reliable robotic automation, network load balancing, and intelligent scheduling algorithms.

Reference [10] proposed a hybrid healthcare solution, where a patient's profiles and health data are stored in a server in the hospital by the patient's mobile or home computer. Physicians examine these data through a hybrid cloud-based system, and decide if the patient needs to be admitted or not. All data are encrypted by a secured cryptographic technology. Reference [11] proposed a secured hybrid cloud solution for healthcare information systems using Windows Azure as a public provider, and virtual environment Hyper-V as the private cloud. The main characteristics of these proposed solutions are availability, authenticity, and flexibility. The system creates a virtual private network that uses certificates to authenticate the clients. This virtual private network allows users to access their desired intranet securely when they are on the public Internet.

Reference [12] proposed an efficient Hybrid cloudlet-based mobile cloud computing model. This model helped to reduce consumed power and time delay. Here, mobile devices relate to a cloudlet if it is available, instead of an enterprise cloud server. If the service is unavailable in the connected cloudlet, a routing system transfers the task to the nearest cloudlet. In the case of unavailability of a cloudlet system, the mobile user needs to use an enterprise cloud server. Reference [13] proposed a mobile cloud-based food calorie measurement system. The authors applied a food recognition algorithm using a Support Vector Machine (SVM) classifier. Forty distinct categories of food and fruits were used in this experiment data. While 50% of data were used as train data, the remaining 50% were used as test data. The average accuracy was 99%.

Collaboration among various healthcare systems remains an issue. Various pharmacies, hospitals, clinics, emergency services, and insurance companies all follow different naming systems. Reference [7] gives a system with a semantic gateway at the network edge for rest API, which can be used for the collaboration of health systems. Other add-ons performed include local storage, security, data analysis, data compression, and standardization. Reference [8] proposed a novel approach for an energy-efficient task offloading to edge servers. Interaction among edge servers and wireless body area network (WBAN) users was formulated as a Stackelberg game, since users compete for edge servers. The alternating direction method of multipliers (ADMM)-based algorithm was used to find Stackelberg equilibrium in a distributed environment, such that edge servers were selected by users in an energy-efficient manner. Deep learning as a service provided by cloud infrastructure was used to provide customer prediction services, which raises privacy concerns among customers of sharing their personal data with untrusted organizations. Such data sharing and privacy concerns must be dealt with in healthcare applications. Reference [14] proposed a novel low expansion rate homomorphic encryption scheme with packing and unpacking methods using a convolutional neural networks (LeHE4SCNN) approach. It was scalable, privacy-preserving, and communication efficient in terms of response time and usage cost. On similar lines of privacy conservation in health care systems, Ref.

[9] proposed a blockchain-based technique with patient-centric personal health records using patient consent. This is a steppingstone for patient-centric data management in healthcare systems.

An IoT-based healthcare application using PaaS prototype was proposed in Ref. [15] for hybrid cloud and fog environments. This prototype enables the provisioning of IoT applications, while existing PaaS solutions do not support provisioning different applications with components spanning cloud and fog.

2.2 Why Healthcare Systems Need Mobile Cloud Computing

Mobile cloud computing (MCC) is built based on concepts of cloud computing and mobile computing as the combination of cloud computing technologies with mobile devices, to bring rich computational resources to mobile users [16]. The main purpose of MCC is to enable execution of an excessive number of mobile applications on mobile devices, to provide high availability and reliability through the Internet. Thus, healthcare systems based on MCC offer more efficient services to both service providers and users to achieve express access and use health services anywhere and anytime. MCC raises the healthcare service level more efficiently by providing high-quality and low-cost healthcare services to patients, because the healthcare data are tracked, and transmitted into mobile devices. Its main benefit to doctors is to analyze the healthcare data in real-time, so that stakeholders (e.g., patients, doctors, and hospitals) could have the latest update for a disease or infection.

However, as is well know the size of a mobile device is usually small, the maximum capacity of computation, storage, and power is always limited. Thus, high-volume data processing in healthcare systems based on MCC is managed efficiently and synchronized into a distributed execution of cloud computation and mobile device. This MCC-based solution is a computation offloading, such that the data processing part would be sent to the cloud servers to be integrated, and then once the data execution tasks have been completed, sent back to the mobile device. From our earlier research [17], there are main advantages and challenges of mobile cloud computing related to healthcare systems.

***Benefits of mobile cloud computing*:** MCC has four main benefits in healthcare systems, as follow: (1) MCC supplies multiple types of cloud platforms to execute rich applications on mobile devices. (2) It supplies real-time data accessibility and high-scale computing capability to support a large volume of healthcare data analysis and processing on time. (3) It provides a customized payment method as a pay-as-you-go method that allows users to be charged based on usage of resources, rather than the traditional provisioning method for a certain number of resources that might or might not be used. (4) It provides high availability and reliable communication between mobile users and the healthcare systems, so that mobile devices can access cloud services over mobile networks, or access points to users can be accessed from any location. Therefore, healthcare system users should get access from any location and any time in the world through MCC.

***Challenges with mobile cloud computing*:** There are four main challenges to mobile cloud computing-based solutions [18,19], as follow: (1) Network availability: Due to the mobility nature, mobile devices may be disconnected from one domain network, and re-connected to another domain network. To solve these interruptions of cloud services, mobile devices require a high stable connection technology to provide seamless communication in MCC. (2) Latency reduction: Latency is the measurement of delay when a request is returned to its original user. This also occurs within the MCC environment, such that delays can arise anywhere from the edge mobile devices to the end-servers in a cloud data center. Thus, techniques for reducing the average end-to-end delay are required in real-time healthcare applications in the MCC environment. (3) Resource availability: Some resources are costly and limited in the MCC environment. For example, the capacity of network bandwidth

is limited, the energy capacity of both mobile devices and servers in healthcare database centers is limited, and rich on-demand data requirements could be made at the same time from many users. Thus, highly efficient usage of resources is essential to achieve a minimized processing time and response time in MCC. (4) Security and privacy: Protection of user data remains one of the most important technical issues in MCC. Most mobile devices store confidential user information, such as medical records, payment information, and other personal privacy data, which are shared with the MCC infrastructures. Thus, healthcare systems based on MCC require an optimal privacy and authentication solution for healthcare data.

Load balancing for resource management: As the demand for using clouds from IoT devices is significantly increasing day-by-day, load balancing is a key solution to managing performance and resource utilization in cloud systems. A load balancing solution is important to effectively allocate cloud resources to enhance the performance of cloud computing. It can control the scheduling of incoming requests among available back-end servers in the MCC environment. It can divide the workload of a server among clustered servers to ensure efficient resource utilization and the rapid analysis of results. It ensures that all back-end servers of the healthcare systems in MCC are equally loaded to guarantee high quality-of-service to end users. And it is necessary to effectively offload data traffic, when congestion happens between clouds and edge mobile devices. Thus, an efficient load balancing solution is concerned with the following purposes: improving resource usage efficiency, fending off overload and breakdown, enhancing service availability, and restraining downtime. In general, we should consider the main criteria while designing a load balancing solution in MCC as follow. It should generate less overhead; it should keep the latest load information; it should balance the system uniformly; it should run on a dedicated system; its migration should take minimum downtime; and its network communication should be reliable and fast. Research in load balancing and task scheduling has been conducted for a long time, and has been used in various applications. CPU process scheduling, batch process scheduling, and token passing are many applications. Edge servers and cloud servers frequently get computation tasks. Increased accessibility of applications for mobile devices has increased network traffic on mobile clouds. Task offloading, and then task scheduling at edge servers, is the latest area under research.

Energy efficient scheduling on federated Edge cloud (ESFEC) in Ref. [20] provides two heuristic-based algorithms for task placement: (ESFEC-migration first), and (ESFEC-energy first). These variations of the same algorithm place services based on migration or energy consumption criteria to be conserved. Whenever a new task arrives, a service placement manager is initiated, which analyses the actual traffic requirements, and allocates the virtual machine (VM) running services to one of the edge servers. Service monitoring is conducted periodically, to check if the CPU utilization of any VM does not exceed the threshold value. If CPU utilization exceeds that value, then migration is initiated, and again the task is scheduled to another machine. A genetic algorithm-based graph coloring approach is applied for scheduling purposes in Ref. [21]. This paper considers the tasks as vertices of graph $G = (V, E)$, and the reachability of other tasks as edges. Further, graph coloring is done for this graph. The color of vertex (task) represents which edge server would be selected to run the task. This approach helps in reducing the number of edge servers for execution, and optimizing it. The graph coloring is done using a genetic algorithm that includes fitness calculations, followed by selection, crossover, and mutation. Eventually, the algorithm returns an optimized task to edge mapping in the form of a colored graph. Reference [22] proposes a novel HEELS algorithm based on the glowworm movement algorithm. They generate clusters of tasks that are allocated per edge server. Since task scheduling is an NP-hard problem, reinforcement learning based solutions are also provided. Reference [23] uses deep deterministic policy gradient-based scheduling. A set of tasks and resources are considered as the

state of the system, and policy is learned to find optimal action of mapping edge server to the given task set. This algorithm is designed to reduce time and energy consumed in task computation.

Among multiple issues highlighted for edge computing in healthcare systems, we propose an architecture to address the following issues:

- **Security:** We separate the private cloud of hospitals from the public cloud with different privileges to each cloud, so that user data is not compromised.
- **Load balancing:** A lot of work is done in the task of offloading health care systems to edge servers. But since the task to be deployed is more computation centric for health systems (ones from IoT devices), they need a proper load balancing strategy. We use load balancing strategies in our architecture.
- **User mobility:** Mobility remains an issue for most of the edge-enabled applications. When users move across the accessibility of edge servers, task migration is required, so that the user can access the task results from the new location.
- **Edge server overloading:** For high computation and data intensive applications, edge servers are often overloaded when network traffic is high. For some applications that are delay sensitive, it is not always possible to offload the computation to cloud servers. Task migration to other edge servers through collaboration is another important aspect of our proposed architecture. Moreover, if edge servers are not available, we propose to use virtual edge servers [24] as well.

3 The Proposed Hybrid Mobile Cloud Computing (HMCC) Architecture

3.1 System Design

Our proposed hybrid cloud has at least one private cloud, and at least one public cloud. The internal structures of the two types of clouds are consistent with each other. In our proposed architecture, private clouds are used to store and process medical data within the health organization. This allows Information Technology (IT) staff to have more control over stored medical data. Only physicians and IT staff have direct access to private clouds. Users can get their health data and diagnosis updated from that private cloud, but they need to be authorized members of that system. The public cloud is open to both physicians and patients. The patient's and doctor's simplified profiles are there. Our main objective is to maintain connectivity while the user is moving from one place to another place, and incorporating an efficient load balancing algorithm for the maximum usage of the cloud resources. We describe the scenario of this model in Fig. 3 with its architecture in Fig. 2 as follows:

- The mobile user will be connected to the main cloud through edge-clouds using mobile data, or an access point (Wi-Fi, hotspot, etc.) when the region is covered by an existing edge-cloud (or if connectivity is available, directly to the main cloud).
- When the user is at the edge of an edge cloud, and about to move out from that edge, they are connected to the nearest available edge cloud. The system migrates the ongoing process from the previously connected edge cloud to a newly connected edge cloud through the main cloud with minimum interruption.
- If the number of service requests in a specific edge cloud is more than its threshold level, the system automatically connects the upcoming services to the nearest available edge-cloud by dynamic load-balancing algorithms.
- If no edge-cloud is available during the mobility of user and after crossing the threshold of a particular edge-cloud, the system automatically creates a new edge-cloud.
- When the last user leaves the edge cloud, the system automatically drops the unused edge cloud.

- The load balancing algorithm produces minimum data overhead, and a management system measures and controls the data overhead of the system.

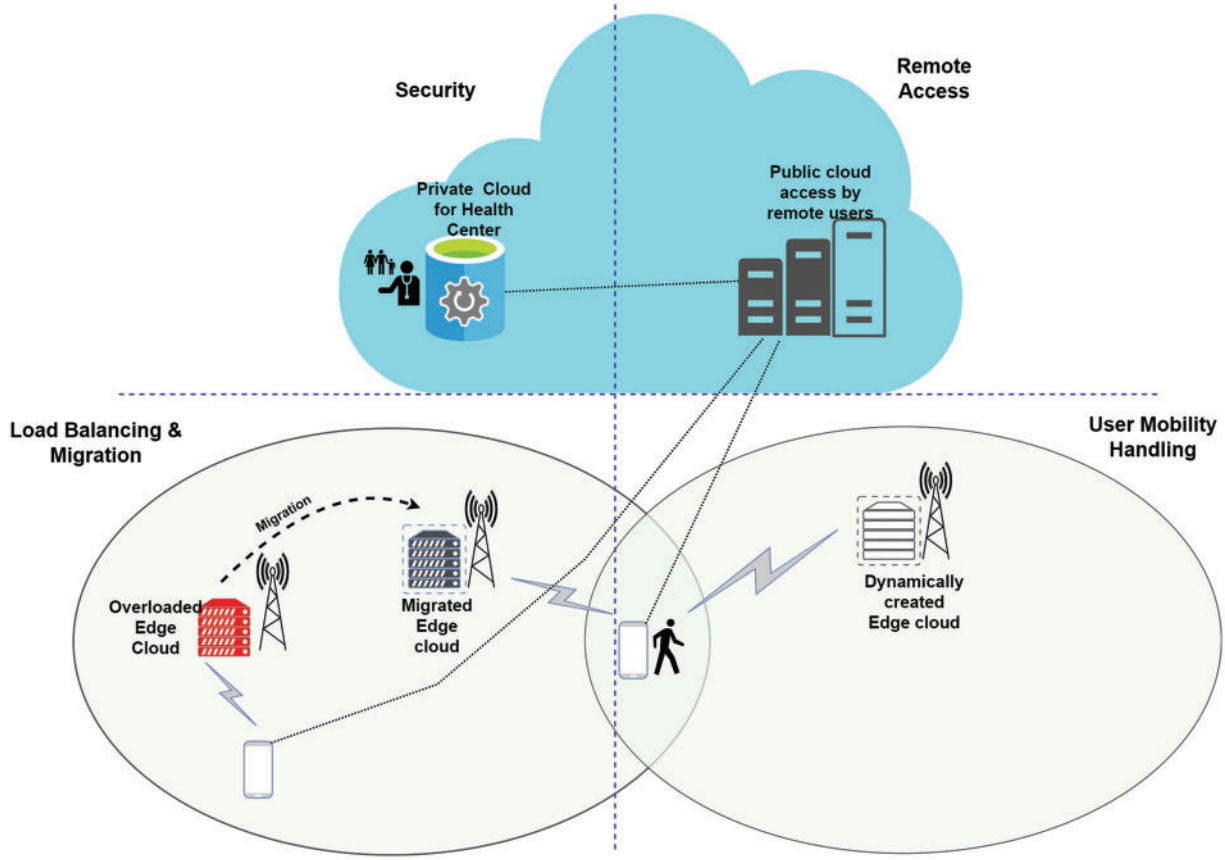


Figure 2: The proposed hybrid mobile cloud computing (HMCC) architecture

3.2 How Load Balancing is Used to Optimize Latency and Energy

After the task offloading decision is made, the tasks from users are divided into two major categories; a set of tasks is offloaded to remote servers (edge and/or cloud), while other sets are computed locally. For those who are offloaded, they need one more level of optimization, which will help to further reduce the latency and energy consumption. This level is load balancing. Assume there are M edge servers available, and N tasks are offloaded to edge servers, then the mapping to N tasks to M edge servers needs to be optimized, so that there is minimal waiting time for each task, and resources at edge servers could suffice to meet the needs of mapped tasks. Moreover, task deadlines should also be met with minimal migration overhead, reducing the possibility of overloading edge servers.

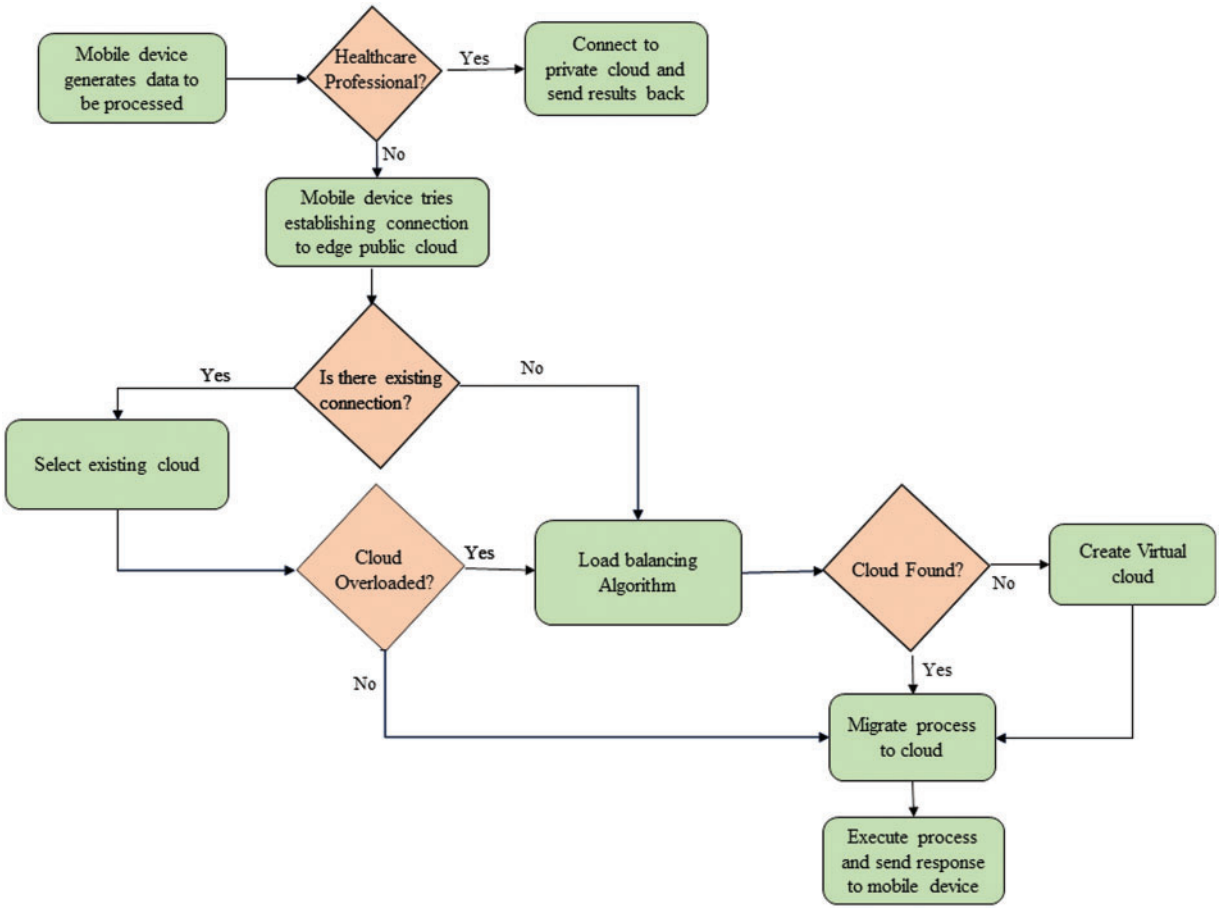


Figure 3: Flowchart showing the process of remote task execution using a hybrid mobile cloud computing (HMCC)

Load balancing algorithms are normally two categories: static and dynamic. Static algorithms are suitable for low traffic data and this traffic is equally distributed over all servers. But when any server gets overloaded, migration does not depend on the current state of the system. Dynamic algorithms consider the current state of the system and distribute workload based on that. We studied static algorithms like round-robin, weighted round-robin, min-min, and max-min static algorithms [25]. For these algorithms, system state changes are not considered for load balancing and algorithm execution time is large for increasing the size of tasks. Dynamic algorithms such as active clustering, honeybee foraging, and ant colony optimization [25] are solutions to a dynamic environment. These general dynamic algorithms provide cost and time efficient results. Nowadays, machine learning based load balancing solutions are getting a lot of attention such as an energy efficient scheduling on federated edge cloud based on reinforcement learning (ESFEC-RL) [20], deep deterministic policy gradient (DDPG) [23], and graph coloring (GRAPH) [21]. Although algorithms such as ESFEC-RL [20] find optimal costs, execution time to converge to minimal cost is high. Real-time result calculation often fails due to high convergence time. Hence, we studied the heuristic-based, static an energy efficient scheduling on federated edge cloud based on energy first (ESFEC-EF) [20] algorithm, for analyzing load balancing algorithms in terms of latency and energy efficiency in this paper. To compare the benefits of dynamic algorithms, we considered comparing the results of two other dynamic algorithms

based on machine learning algorithms (DDPG and GRAPH). Tab. 1 presents the summary of benefits and limitations of the three load balancing algorithms.

Table 1: Benefits and limitations of load balancing based on our evaluation algorithms

Load balancing algorithm	Type	Benefits	Limitations
Energy efficient scheduling on federated edge cloud based on energy first (ESFEC-EF) [20]	Static	• This algorithm has lower energy consumption in high traffic networks. • Despite its limitations, service migration overhead is reduced due to VM selection based on minimum overhead and maximum CPU utilization.	• The number of service migrations increases with the increase in traffic.
Deep deterministic policy gradient (DDPG) [23]	Dynamic	• It can handle a more dynamic environment and scales well with substantial number of request/servers. • It optimizes resource placement and task dispatching process lowering the overall latency and energy consumption.	• It takes more iterations and hence more running time to optimize the balancing problem. But the results are much better than static algorithms. • Equilibrium should be attained among number of iterations and optimization objective value.
Graph coloring (GRAPH) [21]	Dynamic	• It provides scalability to edge servers and cloud servers by increasing the CPU utilization. • It helps with load balancing to reduce the overall latency of the system and reduce the network traffic. • This method is good for dense traffic networks.	• Unlike latency, energy consumption for higher task sizes is more. This is due to more transmission and migration costs incurred to lower the latency.

After observing the benefits and limitations of three load balancing algorithms in Tab. 1, we tested them on our proposed system. ESFEC-EF [20] is a heuristic-based algorithm, DDPG is a reinforcement-based algorithm in [23], and GRAPH is a graph coloring-based algorithm using a genetic algorithm fitness function in [21]. Both Refs. [21] and [23] are machine learning based algorithms with differences in learning approaches. Reference [23] uses reinforcement learning based algorithm which leverages the learning capability of DDPG technique to tackle network variation and find load balancing solution. Whereas Ref. [21] uses a genetic algorithm for solving the optimization problem and reducing complexity during graph coloring which is meant for allocation of workload to

edge servers. These algorithms find optimal load balancing strategies such as reducing the time and energy requirements of the system. For our proposed system, latency is the maximum time required to compute all the tasks offloaded to edge servers by using all or few available edge servers as per the algorithm strategy. If the load balancing algorithm decides to schedule n tasks to edge server m that has a frequency f_m and takes c_m cycles per byte then the time taken to use all N tasks t_m is given as:

$$t_m = \sum_{n=0}^{N} \frac{B_n * c_m}{f_m} \ . \tag{1}$$

The total time T taken by our system to compute all tasks on M edge servers from all N users is given by:

$$T = \sum_{m=0}^{M} t_m \ . \tag{2}$$

The total energy E consumed when executing these tasks on edge servers is given as:

$$E = \sum_{m=0}^{M} B_k * c_m * z, \tag{3}$$

where, $z = 4/3 * E_i$, and E_i is the energy consumed by the edge server when it is idle. Though the energy calculation remains the same irrespective of scheduling policy, they stand important, because it is required to check the energy thresholds of every edge server when scheduling new tasks to it. This avoids the chances of the edge server becoming overloaded. For edge servers with different computation capabilities, and different battery capacities, an optimal strategy is designed to maintain a minimum latency and energy equilibrium.

4 Simulation Environment, Result Analysis and Evaluations

We compare three load balancing algorithms using our proposed architecture. For our proposed a hybrid mobile cloud computing (HMCC) architecture, load balancing for multiple tasks that are offloaded to edge servers is evaluated using an energy efficient scheduling on federated edge cloud based on energy first (ESFEC-EF) algorithm from Ref. [20], a deep deterministic policy gradient (DDPG)-based scheduling in Ref. [23], and graph coloring (GRAPH) with a genetic algorithm fitness function for scheduling [21].

4.1 Simulation Environment Setting

We simulate the working for each algorithm using Python on 11 Gen Intel® Core™ i7-11370H @ 3.30 GHz, 2,995 MHz, 4 cores, and 8 logical processors. Tab. 2 shows the assumed values of parameters for evaluation.

Table 2: Simulation parameters

Parameter	Description	Initialization
M	Maximum number of edge servers available	20
N	Number of user devices scheduling task	[40, 90, 140, ..., 390]
B	Size of each task	Random (2 – 200) MB

(Continued)

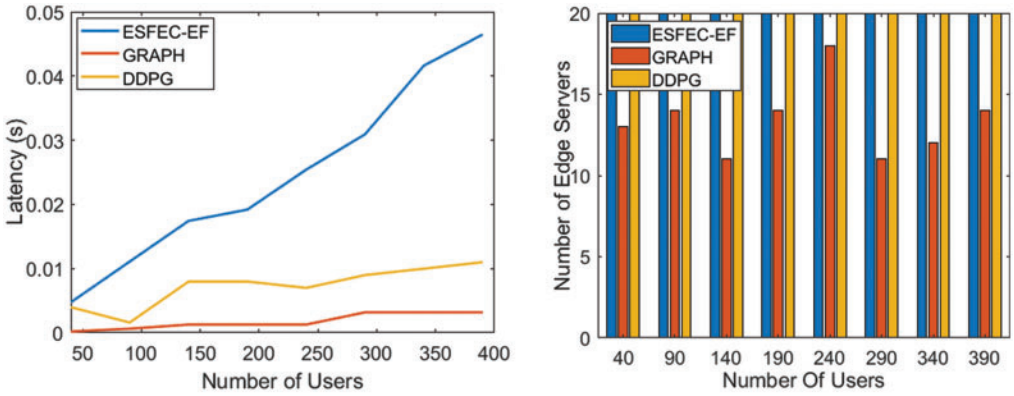
Table 2: Continued

Parameter	Description	Initialization
c	Computation cycles to process one byte on CPU	1
f	CPU frequency	5 GHz
E_i	Energy consumed by CPU when idle	75 W

4.2 Result Analysis and Evaluation

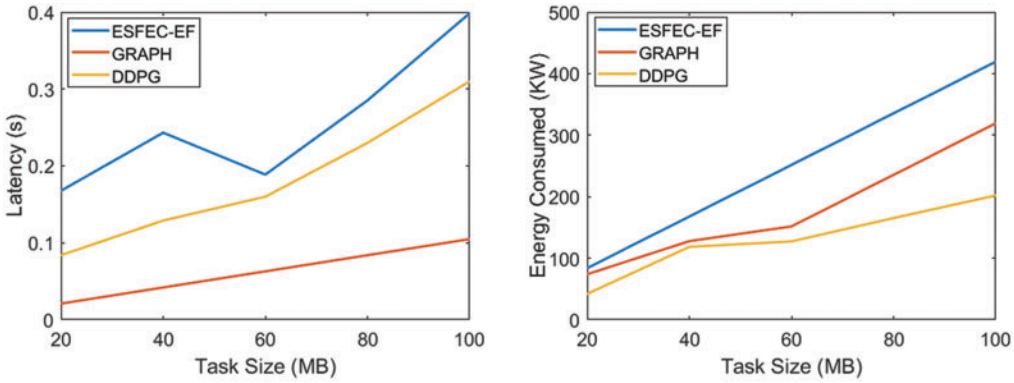
For increasing number of users with minimal task size (2 MB), we compare the latency consumed shown in Fig. 4a. We also show how the optimally graph coloring-based algorithm GRAPH chooses edge servers in Fig. 4b. GRAPH, graph coloring-based algorithm and DDPG-based algorithm are machine learning algorithms. They eventually learn, but they provide better results after learning Fig. 4a shows the lower latency of scheduling strategy provided by GRAPH and DDPG algorithms, as compared to the ESFEC-EF heuristic-based algorithm. The rate of change in latency for increased users is also less for DDPG and GRAPH algorithm, as compared to ESFEC-EF algorithm; this shows the post-learning effect of ML algorithms and better performance. Additionally, GRAPH algorithm helps to optimally allocate the workload to edge servers, which helps in further lowering latency. When compared to ESFEC-EF, latency observed when using DDPG is 20% and 6% when we use GRAPH. Fig. 4b shows the distribution of tasks on edge servers, and the number of edge servers where tasks are scheduled. We have maintained a maximum of 20 edge servers available for tasks of diverse sizes. ESFEC-EF and DDPG algorithm use all 20 servers for migration. From the simulation results, we observe that GRAPH, a graph coloring-based algorithm, uses a smaller number of edge servers than the other two ESFEC-EF and DDPG algorithms from small to a considerable number of users. A notable feature here is, for a network with lower edge servers and requiring cost cutting but having stringent latency constraints, GRAPH algorithm gives better results.

Figs. 5a and 5b show resources consumed (latency and energy) for varying task sizes. We assume 40 users to be present on the network. In Fig. 5a, we observe that GRAPH, graph coloring-based algorithm gives the best results by obtaining a scheduling strategy with minimal time (75% less time as compared to ESFEC-EF) requirement among the three whereas DDPG consumes 25% less time as compared to ESFEC-EF. But GRAPH consumes more energy than DDPG-based algorithm. DDPG-based algorithm maintains equilibrium to minimize latency and energy more as compared to ESFEC-EF and GRAPH algorithms. This reinforcement learning algorithm of DDPG learns with time and provides better results for higher task sizes. With an increase in task size, the rate growth of time and energy requirement is slower for DDPG-based algorithm than GRAPH and ESFEC-EF algorithms, and both DDPG and GRAPH have less energy consumptions than ESFEC-EF; in particular, DDPG consumes significantly less energy about 50% energy compared with ESFEC-EF, and GRAPH consumes 75% of energy as ESFEC-EF for large task sizes from (60 to 100) MB. After comparing the three algorithms shown Tab. 3, the following are **our major observations:**



(a) Latency for varying number of users (b) Edge servers used for varying number of users

Figure 4: Performance comparison of three load balancing algorithms for varying number of users



(a) Latency comparison for varying task size (b) Energy consumption for varying task size

Figure 5: Figure shows energy consumed in KW for increasing task sizes in MB

Table 3: Impact analysis and the limitations of existing models

Type of architecture	Proposed approaches	Limitations
Mobile application [26–31]	• Development of web applications and mobile applications for remote access to healthcare professionals. • These applications are intended for chronic diseases that need immediate diagnosis and treatment, like congestive heart failure, arrhythmia, and hypoxia. • They use human–computer interface designing, text messaging, SMS, calls, text document sharing, etc.	• This approach lacks real-time collection of physiological signals and readings from patients. • It does not support video calling features, or wireless communication. These features limit the scope of healthcare systems in wireless form. • The number of patients reviewed per day remains less. • It requires patients to have skills to collect and input data into applications. Thus, it remains useless in the case of emergency.

(Continued)

Table 3: Continued

Type of architecture	Proposed approaches	Limitations
Mobile cloud computing [32–36]	• This approach uses cloud servers for high traffic healthcare applications. • Improvement of network level resources promotes availability and computation heavy healthcare applications to be used. • This approach resolves several serious issues concerning security, data protection and ownership, quality of services, and mobility. • This leads to expansion of capabilities and benefits and the overcoming of limitations, such as limited memory and CPU power.	• This approach requires mobile devices to communicate with cloud servers directly. • Though the computation cost is reduced due to task offloading, the cost of transmission increases. This leads to latency, as well as energy consumption. • With the increase in IoT devices in healthcare systems, there is a need for low energy consuming solutions.
Hybrid edge-cloud computing [37–41]	• Edge computing offers useful computing resources at the edge of the network to maintain low-latency and real-time computing. • Computing solutions are provided at the edge of the network.	• All patient data is stored on the same, which is accessed by healthcare professionals and patients from public network. This introduces security concerns. • With readily available IoT devices, increased accessibility to mobile networks, network traffic increases, and edge servers are often overloaded.
Our proposed hybrid mobile cloud computing (HMCC) architecture	• We separate the data stores and access rights for health professionals and public network, which deals with the concern of security. • The major focus of our paper also lies in dynamic load balancing algorithms to optimize latency and energy consumption. • This solution is suitable for low battery IoT devices, and mobile devices. • Virtual edge cloud creation also helps when the capacity of available edge servers is all used up. • Hybrid edge and cloud computing and load balancing architecture is a basic architecture for 5G-based metaverse applications.	• Despite the promise of latency and energy optimization, our proposed HMCC architecture needs to provide its adoption with application-specific requirements, such as data rates and real-time communication in terms of bandwidth limitation, coexistence with other cloud computing technologies, scalability, coverage, and security–for the future of IoT connectivity.

- We observe that machine learning based algorithms perform better than heuristic algorithms.
- When energy and latency constrained load balancing, DDPG-based algorithm gives better results.
- For networks with less edge servers but requiring lower latency, GRAPH algorithm proves useful.

5 Open Challenges

Healthcare systems have certain constraints in terms of load computation, bandwidth, and security. Based on our literature survey, simulation of state-of-the-art algorithms for load balancing, observations, and our comparison results, we have noted some **open challenges**, as follow:

- **Exponential rate of increment of latency**: We observe in Fig. 5a that for increasing task sizes, there is exponential growth of latency for all three algorithms. This is unacceptable for 5G network applications, and it is not possible to implement virtualization in health care applications.
- **Exponential rate of increment of energy:** We observe in Fig. 5b that for increasing task sizes, there is exponential growth of energy consumption for all three algorithms. For health applications on mobile devices, energy consumption could become an issue, due to battery drainage of IoT devices.
- **Security:** Though we have secured the private cloud at health care centers, and collaborated data access by public and private cloud is also secured, data from IoT devices or mobile devices that arrive in the private cloud for the first time from users need to be secured.
- **Energy conservation:** Wireless sensor networks play an especially significant role in healthcare systems. Moreover, energy conservation at the sensor level is crucial. Contribution to energy harvesting wireless sensor networks like the one in Ref. [42] should also be incorporated in healthcare cloud-based architecture.

6 Conclusions

In this paper, we proposed a hybrid mobile cloud computing (HMCC) architecture based on combined edge and cloud computing for healthcare applications. We designed it by keeping in mind concerns such as security, increasing traffic, latency, and energy consumption issues. Separation of public and private cloud ensures that vulnerable patient data is secured from the external public network. For managing network traffic for high throughput with minimal latency and energy consumption, we resorted to load balancing techniques. We compared static, as well as dynamic, load balancing algorithms with our architecture, and observed that dynamic algorithms provide better results. Dynamic algorithms, such as the graph coloring (GRAPH) algorithm, prove perfect when the number of edge servers in the network is less as compared to the workload, whereas the deep deterministic policy gradient (DDPG) algorithm is useful when both latency and energy conservation are of importance. We compare the results of these dynamic algorithms to the static algorithm, the energy efficient scheduling on federated edge cloud based on energy first (ESFEC-EF), and show better results are achieved using the dynamic approach. The edge cloud concept is a principal factor for the patients to gain access to the cloud from anywhere at any time. This architecture can form the basis for metaverse-based healthcare applications. Thus, our next goal is to provide a detailed implementation of HMCC architecture, and propose a dynamic load-balancing algorithm to support applications to function in the metaverse, and further adding blockchain technology to ensure secure transmission of encrypted patient data over the network [43].

Funding Statement: This research was supported by the Bio and Medical Technology Development Program of the National Research Foundation (NRF) funded by the Korean government (MSIT) (No. NRF-2019M3E5D1A02069073) and was also supported by the Soonchunhyang University Research Fund.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] World Health Organization, "Primary health care on the road to universal health coverage: 2019," *Global Monitoring Report*, Geneva, Switzerland, (2020). https://www.who.int/publications/i/item/9789240004276.

[2] F. Liu, J. Tong, J. Mao, R. Bohn, J. Messina *et al.,* "Nist cloud computing reference architecture," *NIST Special Publication*, vol. 500, no. 2011, pp. 1–28, 2011.

[3] J. Zhang, D. Li, Q. Hua, X. Qi and Z. Wen, "3D remote healthcare for noisy CT images in the internet of things using edge computing," *IEEE Access*, vol. 9, pp. 15170–15180, 2021.

[4] M. Aazam, S. Zeadally and E. F. Flushing, "Task offloading in edge computing for machine learning-based smart healthcare," *Computer Networks*, vol. 191, pp. 108019, 2021.

[5] B. Rabeya, A. R. Uzzal, A. A. Omar, M. Z. A. Bhuiyan and M. S. Rahman, "HIDE chain: A user-centric secure edge computing architecture for healthcare IoT devices," in *IEEE INFOCOM 2020-IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS)*, Toronto, ON, Canada, pp. 376–381, 2020.

[6] S. U. Amin and M. S. Hossain, "Edge intelligence and internet of things in healthcare: A survey," *IEEE Access*, vol. 9, pp. 45–59, 2020.

[7] T. Sigwele, Y. F. Hu, M. Ali, J. Hou, M. Susanto *et al.,* "An intelligent edge computing based semantic gateway for healthcare systems interoperability and collaboration," in *IEEE 6th Int. Conf. on Future Internet of Things and Cloud (FiCloud)*, Barcelona, Spain, pp. 370–376, 2018.

[8] P. K. Bishoyi and S. Misra, "Enabling green mobile-edge computing for 5G-based healthcare applications," *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 3, pp. 1623–1631, 2021.

[9] H. Kim, S. Lee, H. Kwon and E. Kim, "Design and implementation of a personal health record platform based on patient-consent blockchain technology," *KSII Transactions on Internet and Information Systems*, vol. 15, no. 12, pp. 4400–4419, 2021.

[10] Y. S. Lee, N. Bruce, T. Non, E. Alasaarela and H. Lee, "Hybrid cloud service-based healthcare solutions," in *2015 IEEE 29th Int. Conf. on Advanced Information Networking and Applications Work-Shops*, Gwangju, Korea (South), pp. 25–30, 2015.

[11] R. C. Chioreanu, C. V. Mihael, S. T. Lăcrămioara and S. T. Vasile, "Implementing and securing a hybrid cloud for a healthcare information system," in *11th Int. Symp. on Electronics and Telecommunications (ISETC)*, Timisoara, Romania, pp. 1–4, 2014.

[12] A. T. Lo'ai and S. Habeeb, "An integrated cloud-based healthcare system," in *Fifth Int. Conf. on Internet of Things: Systems, Management and Security*, Valencia, Spain, pp. 268–273, 2018.

[13] P. Pouladzadeh, P. Kuhad, S. V. B. Peddi, A. Yassine and S. Shirmohammadi, "Mobile cloud-based food calorie measurement," in *IEEE Int. Conf. on Multimedia and Expo Workshops (ICMEW)*, Chengdu, China, pp. 1–6, 2014.

[14] Y. Bai, Y. Feng and W. Wu, "Privacy-preserving and communication-efficient convolutional neural network prediction framework in mobile cloud computing," *KSII Transactions on Internet and Information Systems*, vol. 15, no. 12, pp. 4345–4363, 2021.

[15] O. Bibani, C. Mouradian, S. Yangui, R. H. Glitho, W. Gaaloul *et al.,* "A demo of IoT healthcare application provisioning in hybrid cloud/fog environment," in *IEEE Int. Conf. on Cloud Computing Technology and Science (CloudCom)*, Luxembourg, Luxembourg, pp. 472–475, 2016.

[16] M. Othman, S. A. Madani and S. U. Khan, "A survey of mobile cloud computing application models," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 393–413, 2013.

[17] R. K. Das and A. Lee, "A study of load-balancing solutions of mobile cloud computing for next-generation mobile applications," in *Proc. of the Int. Conf. on Research in Adaptive and Convergent Systems*, New York, NY, United States, pp. 119–123. 2020.

[18] R. Sharma, S. Kumar and M. C. Trivedi, "Mobile cloud computing: Bridging the gap between cloud and mobile devices," in *5th Int. Conf. and Computational Intelligence and Communication Networks*, Mathura, India, pp. 553–555, 2013.
[19] S. Al-Janabi, I. Al-Shourbaji, M. Shojafar and M. Abdelhag, "Mobile cloud computing: Challenges and future research directions," in *10th Int. Conf. on Developments in eSystems Engineering (DeSE)*, Paris, France, pp. 62–67, 2017.
[20] Y. Jeong, E. Maria and S. Park, "Towards energy-efficient service scheduling in federated edge clouds," *Cluster Computing*, pp. 1–13, 2021. https://doi.org/10.1007/s10586-021-03338-9.
[21] J. B. Lim and D. W. Lee, "A load balancing algorithm for mobile devices in edge cloud computing environments," *Electronics*, vol. 9, no. 4, pp. 686, 2020.
[22] Y. Dong, G. Xu, Y. Ding, X. Meng and J. Zhao, "A 'joint-me' task deployment strategy for load balancing in edge computing," *IEEE Access*, vol. 7, pp. 99658–99669, 2019.
[23] X. Wei, A. M. Rahman, D. Cheng and Y. Wang, "Joint optimization across timescales: Resource placement and task dispatching in edge clouds," *IEEE Transactions on Cloud Computing*, pp. 1–1, 2021.
[24] M. Laroui, H. I. Khedher, H. Moungla, H. Afifi and A. E. Kamal, "Virtual mobile edge computing based on IoT devices resources in smart cities," in *ICC 2020-2020 IEEE Int. Conf. on Communications (ICC)*, Dublin, Ireland, IEEE, pp. 1–6, 2020.
[25] M. Singh, P. Nandal and D. Bura, "Comparative analysis of different load balancing algorithm using cloud analyst," in *Int. Conf. on Recent Developments in Science, Engineering and Technology*, Singapore, Springer, pp. 321–329, 2017.
[26] S. W. Davis and I. Oakley-Girvan, "Achieving value in mobile health applications for cancer survivors," *Journal of Cancer Survivorship*, vol. 11, no. 4, pp. 498–504, 2017.
[27] W. Y. S. Chou, A. Prestin, C. Lyons and K. Wen, "Web 2.0 for health promotion: Reviewing the current evidence," *American Journal of Public Health*, vol. 103, no. 1, pp. e9–e18. 2013.
[28] A. Ala, E. Lee, N. Alnosayan, S. Chatterjee, L. Houston-Feenstra *et al.,* "Designing patient-centered mHealth technology intervention to reduce hospital readmission for heart-failure patients," in *48th Hawaii Int. Conf. on System Sciences*, Kauai, HI, USA, IEEE, pp. 2886–2895, 2015.
[29] S. Krishna, S. A. Boren and E. A. Balas, "Healthcare via cell phones: A systematic review," *Telemedicine and e-Health*, vol. 15, no. 3, pp. 231–240, 2009.
[30] E. R. Buhi, T. E. Trudnak, M. P. Martinasek, A. B. Oberne, H. J. Fuhrmann *et al.,* "Mobile phone-based behavioural interventions for health: A systematic review," *Health Education Journal*, vol. 72, no. 5, pp. 564–583, 2013.
[31] S. Devi and S. Roy, "Physiological measurement platform using wireless network with android application," *Informatics in Medicine Unlocked*, vol. 7, pp. 1–13, 2017.
[32] M. R. Rahimi, J. Reza, J. Ren, C. H. Liu, A. V. Vasilakos *et al.,* "Mobile cloud computing: A survey, state of art and future directions," *Mobile Networks and Applications*, vol. 19, no. 2, pp. 133–143, 2014.
[33] X. Wang and Z. Jin, "An overview of mobile cloud computing for pervasive healthcare," *IEEE Access*, vol. 7, pp. 66774–66791, 2017.
[34] L. A. Tawalbeh, R. Mehmood, E. Benkhlifa and H. Song, "Mobile cloud computing model and big data analysis for healthcare applications," *IEEE Access*, vol. 4, pp. 6171–6180, 2017.
[35] A. E. Youssef, "A framework for secure healthcare systems based on big data analytics in mobile cloud computing environments," *International Journal of Ambient Systems and Applications*, vol. 2, no. 2, pp. 1–11, 2014.
[36] D. B. Hoang and L. Chen, "Mobile cloud for assistive healthcare (MoCAsH)," in *2010 IEEE Asia-Pacific Services Computing Conf.*, Hangzhou, China, IEEE, pp. 325–332, 2010.
[37] G. Muhammad, F. Mohammed. A. M. Alsulaiman and B. Gupta, "Edge computing with cloud for voice disorder assessment and treatment," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 60–65, 2018.
[38] S. Lanfang, X. Jiang, R. Huixia and Y. Guo, "Edge-cloud computing and artificial intelligence in internet of medical things: Architecture, technology and application," *IEEE Access*, vol. 8, pp. 101079–101092, 2020.

[39] N. Hassan, S. Gillani, E. Ahmed, I. Yaqoob and I. Muhammad, "The role of edge computing in internet of things," *IEEE Communications Magazine*, vol. 56, no. 11, pp. 110–115, 2018.
[40] D. D. Sánchez-Gallegos, A. Galaviz-Mosqueda, J. L. Gonzalez-Compean, S. Villarreal-Reyes, A. E. Perez-Ramos *et al.,* "On the continuous processing of health data in edge-fog-cloud computing by using micro/nanoservice composition," *IEEE Access*, vol. 8, pp. 120255–120281, 2018.
[41] Z. Yang, B. Liang, and W. Ji, "An intelligent end–edge–cloud architecture for visual IoT-assisted healthcare systems," *IEEE Internet of Things Journal*, vol. 8, no. 23, pp. 16779–16786, 2021.
[42] Y. Ge, Y. Nan and Y. Chen, "Maximizing information transmission for energy harvesting sensor networks by an uneven clustering protocol and energy management," *KSII Transactions on Internet and Information Systems*, vol. 14, no. 4, pp. 1419–1436, 2020.
[43] P. N. Srinivasu, A. K. Bhoi, S. R. Nayak, M. R. Bhutta and M. Woźniak, "Blockchain technology for secured healthcare data communication among the non-terminal nodes in IoT architecture in 5G network," *Electronics*, vol. 10, no. 12, pp. 1437, 2021.