This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2022.3206770

# Criticality and Utility-aware Fog Computing System for Remote Health Monitoring

Moirangthem Biken Singh, Navneet Taunk, Naveen Kumar Mall, and Ajay Pratap, *Member, IEEE*

**Abstract**—Growing remote health system allows continuous monitoring of patients' conditions outside medical facilities. However, the real-time smart-healthcare applications having latency limitations, must be solved efficiently. Fog computing is emerging as an efficient solution for such real-time applications. Therefore, Medical Centers (MCs) are becoming more interested in offering IoT-based remote health monitoring services to get profited by deploying fog resources. However, an efficient algorithmic model for allocating limited fog computing resources in a criticality-aware smart-healthcare system while considering the profit of MCs is needed. Thus, we formulate an optimization problem by maximizing system utility, calculate as a linear combination of MC's profit and patients' cost together. We propose a flat-pricing based scheme to measure the profit of MC in health monitoring system. Further, we propose a swapping-based heuristic to maximize the system utility. The proposed heuristic is evaluated on various parameters and shown to be closed to the optimal while considering the criticality of patients and the profit of MC, together. Through extensive simulations, analysis on real-world data and prototype implementation, we find that the proposed heuristic achieves an average utility of $94.5\%$ of the optimal, in polynomial time complexity.

**Index Terms**—IoT, WBAN, Fog Computing, Smart Healthcare, Remote Health Monitoring.

## 1 Introduction

In remote health monitoring system, a patient is equipped with Wireless Body Area Network (WBAN) sensors, capable of collecting health data, and transmitting it to a Local Device (LD) [1]–[3] for further processing. The LD stores and forwards the health data to a Fog Server (FS) for remote health monitoring over 5G networks [4], [5], as shown in Fig. 1. However, rural people have a higher rate of poverty [6] and thus, they cannot afford LDs or IoT sensors on their own. Thus, there is a need for a low cost remote health monitoring system. However, the profit of Medical Center (MC) should also be considered for patients' health monitoring services to encourage the participation of MCs.

Due to criticality, medical data has to be monitored on time without delay involved. For example, health data of patients with chronic illnesses like lung and heart diseases need real-time and continuous assessment; and their monitoring should be prioritized over other diseases. Many critical sensitive diseases' data cannot be computed on low resourced LDs while achieving desired delay constraint. Hence, assistance of FS has emerged to compute patients' health data efficiently while achieving desired delay constraint [7]. Table 1 [3], [8] provides data size and desired delay of different WBAN sensors based on the IEEE Standard 802.15.6-2012. From the table, it can be seen that if all sensors (i.e., ECG, EMG, artificial retina, audio and video) sense data from a patient, and send it to an LD for computation. The desired delays of artificial retina, audio and video do not achieve while doing so[1]. However, FS can reduce latency to a greater extent, allowing real-time remote health monitoring within a desired delay [9], [10].

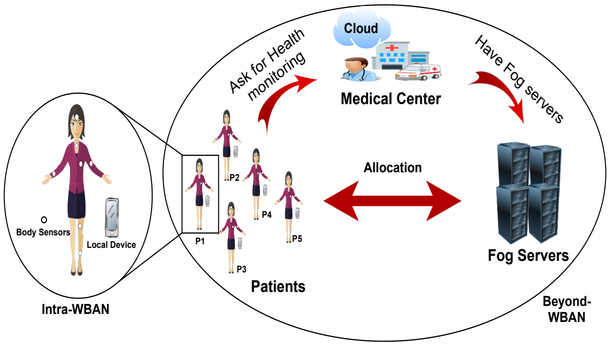


Fig. 1: Remote healthcare architecture.

TABLE 1: Different types of WBAN sensors

| Sensor type | Data size | Desired delay (ms) | Delay at LD (ms) | Delay at FS (ms) |
|---|---|---|---|---|
| ECG | 72 kb | 250 | 208.33 | 4.53 |
| EMG | 80 kb | 250 | 214.52 | 4.68 |
| Artificial retina | 175 kb | 250 | 287.86 | 6.35 |
| Audio | 0.25 Mb | 100 | 345.79 | 7.66 |
| Video | 2.5 Mb | 100 | 2083.33 | 47.17 |

Motivated by the above scenarios, we propose intra-WBAN and beyond-WBAN based system. Intra-WBAN consists of sensors deployed on the patients' bodies, and the LD collects data from them, whereas beyond-WBAN consists of different LDs that send patients' data to FSs. Moreover, we formulate FS assisted beyond-WBAN based remote health monitoring system to minimize the cost of patients while keeping profit of MC into deliberation. Inspired by [11], we aim to use dedicated LDs not only to gather the

- *M. B. Singh, N. Taunk, N. K. Mall and A. Pratap are with the Department of Computer Science and Engineering, Indian Institute of Technology (Banaras Hindu University), Varanasi 221005 India. E-mail: {moirangthembsingh.rs.cse21, navneettaunk.cse18, naveenkumar-mall.cse18, ajay.cse}@iitbhu.ac.in.*

[1] Evaluation parameters are given in Table 4.

patients' health data sent by sensors but also to compute the data locally while achieving desired delay. In nutshell, contributions of this paper are summarized as below:

- Formulate an optimization problem by maximizing the system utility, defined as a linear combination of MC's profit and patients' cost. Moreover, offer a flat-type pricing scheme to measure the profit of MC.
- Propose swapping-based heuristic to maximize the system utility under the constraint of permissible latency for computation of patients' data in polynomial time complexity.
- Through extensive simulations and analysis on real-world data, proposed heuristic is found to achieve an average utility of $94.5\%$ of the optimal solution.

The rest of the paper is organized as follows: Section 2 reviews the relevant works. The system model and the problem formulation are introduced in Section 3. Proposed solution and analysis are given in Section 4. Performance study is presented in Section 5. Finally, Section 6 offers conclusions and future research directions.

## 2 Related Works

This section offers closely related works available in the literature with comparative analysis as given in Table 2.

Primary focused of [12] was to improve haptic communications under three factors- system stability, energy consumption, and network delay. Authors proposed a time-varying swarm algorithm to solve the formulated problem. However, they did not consider profit. Authors of [13] proposed a cost-aware medical cyber-physical system assisted fog computing model. In [14], authors primarily focused on resource allocation to minimize energy consumption and response time through dynamic-cluster algorithm. However, these works [12]–[14] did not consider criticality of patients' data while offloading it to a computing node.

In [15], the authors investigated energy consumption, transmission delay, QoS requirement, power limit and wireless front-haul constraints in fog computing-based Internet of Medical Things (IoMT). However, this model did not consider criticalities of patients and profit of health service provider. Authors of [11] proposed a health monitoring system for IoMT considering criticality, energy and delay constraints. However, this work did not consider profit of MC while offloading the medical data to edge server. The authors of [16] proposed a queue-based transmission of time-sensitive medical data packets in beyond-WBAN using a non-cooperative game-based approach. In [17], authors proposed a criticality-aware dynamic management for medical data transmissions. However, the above works did not consider the profit for delay-sensitive medical data transmission. The authors in [3] proposed a priority-aware time-slot allocation in WBANs. They extended evolutionary game theory to solve the formulated problem. The authors of [18] proposed a Nash bargaining solution for a cooperative game based priority-aware data-rate tuning in WBAN model. However, these works did not consider beyond-WBAN scenario for priority-based data transmission.

**Shortcomings of Existing Approaches**: In most of the existing approaches [3], [12]–[15], [18], only intra-WBAN transmission has been considered under latency and criticality constraints. Some works [11], [16], [17] have considered both intra-WBAN and beyond-WBAN transmission with latency and criticality constraints. However, none of the existing approaches has considered the profit of the MC in their model. Therefore, unlike existing works, we formulate criticality-aware health monitoring system while considering patients' cost and MC's profit, altogether as an optimization problem. Moreover, we propose a novel swapping-based heuristic to solve the formulated problem in polynomial time complexity.

TABLE 2: A relative comparison

| Problem Focus | Criticality | Profit | Beyond-WBAN | Utility Maximization |
|---|---|---|---|---|
| Resource allocation [12] | × | × | × | × |
| Task allocation [13] | × | × | × | × |
| Resource allocation [14] | × | × | × | × |
| QoS requirement [15] | × | × | × | ✓ |
| Data priority, latency [16] | × | × | ✓ | × |
| Criticality aware packet transmission [17] | ✓ | × | ✓ | × |
| Medical criticality [11] | ✓ | × | ✓ | × |
| Time-slot allocation, data priority [3] | ✓ | ✓ | × | × |
| Data-rate tuning [18] | ✓ | × | × | × |
| Criticality and utility -aware resource allocation [Proposed] | ✓ | ✓ | ✓ | ✓ |

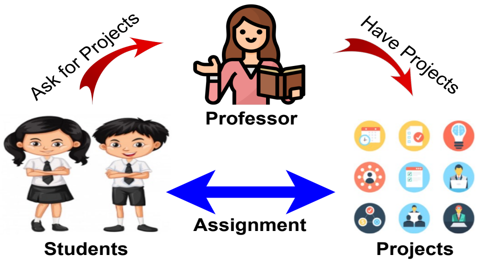


Fig. 2: Student project assignment.

## 3 System Model and Problem Formulation

We consider a remote health monitoring system, provided by MC to a set of patients $\mathbb{P} = \{1, \ldots, p \ldots, P\}$ in coordination with FSs as shown in Fig. 1. Moreover, the descriptions of important symbols are given in Table 3. The proposed framework is equivalent to student project assignment problems in colleges where students approach a professor for project assignment and the professor assigns different projects to the students (see Fig. 2). However, there are limits on the total number of students a professor can allocate and the number of students assigned to a project. Similarly, patients (LDs) request an MC for health monitoring, and the Cloud Server (CS) employed by the MC allocates patients' data to FSs (see Fig. 1). However, there are limits on the number of patients that a CS can allocate as well as the number of patients whose data can be allocated to an FS.

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2022.3206770

TABLE 3: Symbol description

| Symbol | Description |
|---|---|
| $\mathbb{S}$, $\mathbb{X}$ | Sets of sensors and medical criticalities |
| $x_s$ | Medical criticality of sensor $s$ |
| $\theta^s_{p,t}$ | Health data sensed by $s$ from $p$ at time $t$ |
| $\theta_{l,s}$, $\theta_{u,s}$ | Lower and upper limits of normal value for $s$ |
| $d^s_{p,t}$ | Health severity index for $p$ via $s$ at time $t$ |
| $c^s_{p,t}$ | Criticality index for $p$ and $s$ at time $t$ |
| $\mathbb{P}$, $\mathbb{F}$ | Set of patients and FSs |
| $SINR^f_{p,t}$, $V^f_{p,t}$ | SINR and number of allocated PRBs |
| $\rho^c_{p,t}$ | Patient criticality for patient $p$ at time $t$ |
| $\mathbb{H}_t$ | Set of strategies |
| $u_{p,t}$, $q_{p,t}$ | LD and, FS are chosen for computation |
| $\eta_{p,t}$ | Overall data size for $p$ at time $t$ |
| $\beta_{p,t}$ | CPU cycles for computing $p$'s data at time $t$ |
| $T^{c,l}_{p,t}$ | Computation time for $p$ at time $t$ by LD |
| $T^{tr,f}_{p,t}$, $BR^f_{p,t}$ | Transmission time and rate between $p$ and $f$ |
| $T^{c,f}_{p,t}(\mathbb{H}_t)$ | Computation time for a patient $p$ at FS $f$ |
| $\gamma_p(\mathbb{H}_t)$ | Fraction of FS $f$'s resource utilized by $p$ |
| $\mathbb{L}$ | Set of computation capacity of FSs |
| $\Upsilon$, $\Gamma_f$ | Computation capacity of LD and FS $f$ |
| ${n'}_{p,f,t}$ | Number of patients utilizing FS $f$ |
| $m$, $l$ | Computation charge at FS and LD |
| $\chi_t$ | Revenue earned by MC |
| $\phi_t$, $g$ | Expenses of MC and CPU cycle of FS |
| $\delta$, $k$ | Latency constraint and fixed charge per FS |

The problem setting is divided into two parts: intra-WBAN and beyond-WBAN as described in the following:

### 3.1 Intra-WBAN

Consider a set of sensors $\mathbb{S} = \{1, \ldots, s, \ldots, S\}$ deployed on patient's body. Each sensor collects data with different criticality classes and transmits it to an LD for further analysis. To facilitate this phenomenon, we consider medical criticality for prioritizing different health data in a resource-constrained health monitoring scenario.

Let medical criticalities of health data collected by sensors be a set $\mathbb{X} = \{x_1, \ldots, x_s, \ldots x_S\}$. If data collected by sensor $s$ is more critical than that of sensor $s'$, then $x_s > x_{s'}$. Two sensors' data of the same criticality class can have different medical criticalities, thus $x_s \in [0, \infty)$ [11]. Let $\theta^s_{p,t}$ be the parameter value sensed by sensor $s$ from patient $p$ at time $t$, and $\theta_{l,s}$ and $\theta_{u,s}$ be lower and upper limits of reference range- defined as the range of health parameter values under normal conditions for a healthy person[2]. Then, health severity index of patient $p$'s data collected by sensor $s$ at time $t$ is defined as follows [3], [19]:

$$d^s_{p,t} = \left| \frac{(\theta_{u,s} - \theta^s_{p,t})^2 - (\theta^s_{p,t} - \theta_{l,s})^2}{(|\theta_{u,s}| + |\theta_{l,s}|)^2} \right|. \tag{1}$$

Health severity index defines the deviation of a patient's health data value from its normal reading. Higher health severity index indicates more severe data. For instance, ECG data is more critical than temperature data in healthcare [19]. We further define criticality index, $c^s_{p,t}$ for a patient $p$ and sensor $s$ at time $t$ as the product of health severity index and medical criticality as follows:

$$c^s_{p,t} = x_s d^s_{p,t}. \tag{2}$$

Moreover, LD normalizes the health severity index between 0 and 1 using the min-max normalization technique [20]. Then, $p^{th}$ patient's criticality at time $t$ is defined as the average of criticality indices of sensors, as follows:

$$\rho^c_{p,t} = \frac{1}{S} \sum_{s=1}^{S} c^s_{p,t}. \tag{3}$$

Higher criticality value indicates severe condition of a patient. After receiving data at LD, there is a need to make a decision for its computation either at LD or FS in order to achieve the system's constraints. Moreover, computation capacity of all LDs is considered to be uniform and equal to $\Upsilon$. Let $u_{p,t}$ be a binary variable defined as:

$$u_{p,t} = \begin{cases} 1, & \text{system selects LD for computation of } p\text{'s data;} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Computation time at LD for a patient $p$ is calculated as:

$$T^{c,l}_{p,t} = \frac{\beta_{p,t}}{\Upsilon}, \tag{5}$$

where $\beta_{p,t}$ is the required CPU cycles for computing patient $p$'s health data at time $t$. In addition, we assume that each sensor is allocated a different amount of LD's resources.

In the next section, we describe the transmission[3] and computation of health data at FSs.

### 3.2 Beyond-WBAN

Let a set of FSs $\mathbb{F} = \{1, \ldots, f, \ldots, F\}$ be equipped with a respective computation capacity of a set $\mathbb{L} = \{\Gamma_1, \ldots, \Gamma_f, \ldots, \Gamma_F\}$. However, to determine whether or not the system selects FS for computation of $p$'s data, we define a binary variable $q_{p,t}$ as follows:

$$q_{p,t} = 1 - u_{p,t}. \tag{6}$$

Let $\mathbb{H}_t$ be a $P \times F$ binary matrix that indicates the choice of FS for computing patient $p$'s data at time $t$, as given below:

$$\mathbb{H}_t = \begin{bmatrix} h^1_{1,t} & h^2_{1,t} & . & . & h^F_{1,t} \\ h^1_{2,t} & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ h^1_{P,t} & . & . & . & h^F_{P,t} \end{bmatrix}, \tag{7}$$

$$h^f_{p,t} = \begin{cases} 1, & \text{FS } f \text{ computes patient } p\text{'s data;} \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

Transmission rate between patient $p$ and FS $f$ underlying cellular 5G is computed as follows [4]:

$$BR^f_{p,t} = \Omega V^f_{p,t} \log_2 \left( 1 + SINR^f_{p,t} \right), \tag{9}$$

where $\Omega$, $V^f_{p,t}$ and $SINR^f_{p,t}$ are channel bandwidth, number of allocated PRBs[4] and Signal-to-Interference-plus-Noise

[2] A detailed explanation for calculating $\theta_{l,s}$ and $\theta_{u,s}$, i.e., lower and upper limits of reference range is given in Appendix A of [3].

[3] Transmission of data in intra-WBAN is beyond the scope of this work. However, the existing approach [11], can be utilized for intra-WBAN communication.

[4] PRB is the smallest unit of radio resource that can be assigned to a device [4].

Ratio (SINR)[5] between patient $p$ and FS $f$, respectively. We assume each patient communicates over a distinct channel; thus, interference is not considered[6].

Transmission time between patient $p$ and FS $f$ can be:

$$T^{tr,f}_{p,t} = \frac{\eta_{p,t}}{BR^{f}_{p,t}}, \tag{10}$$

where, $\eta_{p,t}$ is the size of patient $p$'s data at time $t$. Like [22], [23], we assume that each patient uses the same amount of FS's resources. Thus, computation time for a patient $p$ at FS $f$ can be calculated as:

$$T^{c,f}_{p,t}(\mathbb{H}_t) = \frac{\beta_{p,t}}{\gamma_p(\mathbb{H}_t)}, \tag{11}$$

where $\gamma_p(\mathbb{H}_t)$ is the fraction of FS $f$'s resource utilized by the patient $p$, calculated as follows:

$$\gamma_p(\mathbb{H}_t) = \frac{\Gamma_f}{{n'}_{p,f,t}}, \tag{12}$$

where ${n'}_{p,f,t}$ is the number of patients utilizing the FS $f$ for their computation if patient $p$ is also utilizing it without violating the constraints.

One of our objectives is to minimize the cost of the patients, defined as the weighted (i.e., criticality) sum of computation and transmission time as follows:

$$J_t = \sum_{p\in\mathbb{P}} \rho^{c}_{p,t} \left( \sum_{f\in\mathbb{F}} \left( h^{f}_{p,t} T^{tr,f}_{p,t} + T^{c,f}_{p,t}(\mathbb{H}_t) \right) + u_{p,t} T^{c,l}_{p,t} \right). \tag{13}$$

Thus, to minimize the cost, we have to lower down the latency for the patient in the beyond-WBAN scenario.

To determine revenue model of MC, we consider a flat-type pricing scheme [24], [25] as described in the following:

### 3.2.1 Flat-type pricing scheme

Let the MC charges $l$ unit price per time slot for computation at LD. If computation is done at FS, the MC charges $m$ unit price per time slot. Generally, FS charges more than that of LD, i.e., $m > l$. Thus, revenue of the MC is calculated as:

$$\chi_t = \sum_{p\in\mathbb{P}} (u_{p,t} l + q_{p,t} m). \tag{14}$$

Let $k$ be the fixed expenses of each FS per time slot and $g$ be the expenses of MC per CPU cycle for computation on the FS. Then, the expenses of MC can be calculated as follows:

$$\phi_t = kF + g \sum_{p\in\mathbb{P}} q_{p,t}\beta_{p,t}. \tag{15}$$

Now, the profit of the MC can be calculated as follows:

$$\Delta_t = \chi_t - \phi_t = \sum_{p\in\mathbb{P}} (u_{p,t} l + q_{p,t} m) - kF - g \sum_{p\in\mathbb{P}} q_{p,t}\beta_{p,t}. \tag{16}$$

The profit of MC from a patient $p$ depends on whether the patient $p$'s data is allocated to an FS or LD. Let the maximum value of $\beta_{p,t}$ be $\beta^{max}_{p,t}$ ($\beta^{max}_{p,t}$ can be approximated by the MC before deciding the values of $m$ and $l$), then profit of the MC can follow the following constraint:

$$m - l \geq g\beta^{max}_{p,t} + \frac{kF}{P}. \tag{17}$$

The above constraint ensures that if a patient's data is allocated to an FS, then the profit of the MC will be higher than if it is allocated to LD, independent of the CPU cycles needed for computing patients' data, as stated in Lemma 1.

**Lemma 1.** *The profit of the MC either increases or remains constant as more patients' data is allocated to FS instead of LD for their computation.*

*Proof.* Let $P'$ be the number of patients whose data is allocated to FS. Then, the profit of the MC is given by:

$$\Delta_{t,1} = P'm + (P - P')l - kF - g \sum_{p\in\mathbb{P}} q_{p,t}\beta_{p,t}. \tag{18}$$

Take any patient $p'$ utilizing LD and allocate it's data to any FS for computation of health data. So, the new profit in this scenario (assuming allocation of all other patients' data remains the same) is given by:

$$\begin{aligned} \Delta_{t,2} = (P' + 1)m + (P - P' - 1)l - kF \\ - g \sum_{p\in\mathbb{P}} q_{p,t}\beta_{p,t} - g\beta_{p',t}. \end{aligned} \tag{19}$$

Now, $\Delta_{t,2} - \Delta_{t,1}$ is given by:

$$\Delta_{t,2} - \Delta_{t,1} = m - l - g\beta_{p',t}. \tag{20}$$

From Eqs. (17) and (20), we have:

$$\Delta_{t,2} - \Delta_{t,1} \geq 0. \tag{21}$$

From Eq. (21), we conclude that the profit increases or remains constant as we increase the number of patients whose data is allocated to FSs for computation. □

As per Lemma 1, the profit only depends on patients whose data is allocated to FSs. So, if all patients' data is allocated to FSs, the profit does not depend on how their data is allocated to FSs. Thus, the utility depends only on the cost of the patients as defined in Eq. (13).

## 3.3 Problem Formulation

In remote health monitoring system, the patient with a higher criticality should be monitored in less time. Thus, lower computation and transmission time are required for the patients to minimize their costs and ensure proper monitoring. Moreover, MC tries to maximize its profit by providing monitoring services under the condition that no patient faces any delay. However, both objectives cannot be achieved at the same time. Therefore, we consider utility as the linear combination of profit of the MC and the cost of patients, that signifies the importance of profit for the monitoring service as well as the cost of patients, as discussed in the following:

$$U_t = \lambda_1\Delta_t - \lambda_2 J_t, \tag{22}$$

where, $\lambda_1$ and $\lambda_2$ are positive weights assigned to the profit of MC and the cost of patients, respectively, and $\lambda_1 + \lambda_2 = 1$. The weights are taken as inverse units of profit

[5] We assume that the channel exhibits flat fading. However, this can be easily extended to frequency-selective fading channels as well [4].

[6] Interference can be solved by applying methods given in [4], [21].

and latency cost, respectively, so that utility becomes unitless. The weights are dependent on the system requirements and should be considered accordingly. That means, if the system is more profit aware then, $\lambda_1 > \lambda_2$, or if the system is more criticality aware then, $\lambda_1 < \lambda_2$, or if it is equally balancing between these two, then $\lambda_1 = \lambda_2$. Moreover, we consider a constraint on the permissible latency defined as $\delta$, to ensure that no patient has a delay more than $\frac{\delta}{\rho^c_{p,t}}$. As a result, the permissible latency should be lower for higher criticality patients. Therefore, the optimization problem of the proposed model is formulated as follows:

$$\arg\max_{\mathbb{H}_t} \; U_t \tag{23}$$

Subject to the constraints:

$$\sum_{f\in\mathbb{F}} \left( h^f_{p,t} T^{tr,f}_{p,t} + T^{c,f}_{p,t}(\mathbb{H}_t) \right) + u_{p,t} T^{c,l}_{p,t} \leq \frac{\delta}{\rho^c_{p,t}}, \forall p \in \mathbb{P}, \tag{24}$$

$$l \leq l_{max}, m \leq m_{max}, \tag{25}$$

$$m - l \geq g\beta^{max}_{p,t} + \frac{kF}{P}, \tag{26}$$

$$\sum_{f\in\mathbb{F}} h^f_{p,t} = q_{p,t}, \forall p \in \mathbb{P}, \tag{27}$$

$$\sum_{p\in\mathbb{P}} h^f_{p,t} \leq F^{max}_f, \forall f \in \mathbb{F}, \tag{28}$$

$$\sum_{p\in\mathbb{P}} \sum_{f\in\mathbb{F}} h^f_{p,t} \leq C_{max}, \tag{29}$$

Eq. (24) refers to the latency constraint. Eq. (25) puts constraint on service charges. Eq. (26) refers to constraint defined in Eq. (17). Eq. (27) ensures that every patient's data is allocated to at most one FS. Eqs. (28) and (29) define the maximum number of patients whose data can be allocated to an FS and the maximum number of patients whose requests can be handled by the CS, respectively.

The formulated problem in Eqs. (23)-(29) is a Binary Integer Programming problem in $\mathbb{H}_t$ decision variables, that is generally NP-hard to solve as its feasibility problem is strongly NP-complete [26]. Due to high conditionality and complexity of the formulated problem, this work proposes a sub-optimal solution for the maximization problem based on swapping-based heuristic in the following section.

# 4 Proposed Solution

To avoid high computation charges at FS, each patient would like to compute the health data at LD. However, they may not be able to satisfy the latency constraint given in Eq. (24) while doing so (see Table 1). Thus, a sub-problem here is to allocate these patients' data to FSs. Let $\mathbb{P}^v_t$ be the set of patients that violate the latency constraint at time $t$ if their data is computed at the LD. Formally,

$$\mathbb{P}^v_t = \{p \in \mathbb{P} : \rho^c_{p,t} T^{c,l}_{p,t} > \delta\}. \tag{30}$$

The MC allocates patients from the set $\mathbb{P}^v_t$ to FSs. Next objective is to allocate a subset of the remaining patients such that it maximizes the utility under the system constraints. Due to limited resources, it is not possible to allocate all of the patients' data to FSs. Doing so may result in violation of the latency constraint given in Eq. (24) and significant increase in the cost of patients, resulting in lower utility. Let $n^{max}_{p,f,t}$ be the maximum number of patients (see Theorem 1) that can utilize the FS $f$ for their computation if patient $p$ utilizes it without violating the constraint given in Eq. (24).

**Theorem 1.** *Maximum number of patients that can utilize the FS $f$ for their computation, if patient $p$ utilizes FS $f$, is given by:*

$$n^{max}_{p,f,t} = \left\lfloor \left( \frac{\Gamma_f}{\beta_{p,t}} \right) \left( \frac{\delta}{\rho^c_{p,t}} - \frac{\eta_{p,t}}{\Omega V^f_{p,t} \log_2 \left( 1 + SINR^f_{p,t} \right)} \right) \right\rfloor. \tag{31}$$

*Proof.* According to Eq. (24), if a patient $p$ utilizes FS $f$, then,

$$T^{tr,f}_{p,t} + T^{c,f}_{p,t}(\mathbb{H}_t) \leq \frac{\delta}{\rho^c_{p,t}}. \tag{32}$$

From Eqs. (10) and (11), we get

$$\frac{\eta_{p,t}}{BR^f_{p,t}} + \frac{\beta_{p,t}}{\gamma_p(\mathbb{H}_t)} \leq \frac{\delta}{\rho^c_{p,t}}. \tag{33}$$

After solving the inequality and putting the value of $BR^f_{p,t}$,

$$n'_{p,f,t} \leq \left( \frac{\Gamma_f}{\beta_{p,t}} \right) \left( \frac{\delta}{\rho^c_{p,t}} - \frac{\eta_{p,t}}{\Omega V^f_{p,t} \log_2 \left( 1 + SINR^f_{p,t} \right)} \right). \tag{34}$$

Thus, the maximum value of $n'_{p,f,t}$ is the greatest integer value of the right-hand side expression. Hence, proved. □

We employ *Brakerski-Gentry-Vaikuntanathan* (BGV) [27] homomorphic encryption scheme in the proposed health monitoring system between LD and FS to maintain the privacy and security of the health data. Here, LD encrypts patient $p$'s data before transmitting it to FS. Our proposed model employs encryption parameters as suggested in [28].

To solve the formulated problem, we propose Utility Maximization Patient Monitoring (UMPM) algorithm. We consider that the MC has CS that executes the UMPM algorithm to allocate patients' data to FSs. Fig. 3 shows data flow between patients, LDs, FSs, and MC. Labels 2, 3, 4 and 7 in Fig. 3 are handled by control signals (e.g., beacons [29]). However, labels 1, 5 and 6 are handled by data signals. UMPM algorithm begins with an initial allocation of patients' data to FSs and then iteratively re-positioning the patients' data by swapping their allocated FSs to achieve higher utility as shown in Fig. 4. The elaboration of each sub-algorithm is described in the following:

## 4.1 UMPM Algorithm

We define $U_{diff,t}$[7] as a difference between utility before and after a patient $p$'s data is allocated to FS $f$, keeping the

[7] The profit of $p$ changes and it affects the cost with a value of the difference of $p$'s local computation time and its transmission and computation time at FS $f$. Moreover, computation time of other patients whose data is allocated to $f$ also changes.

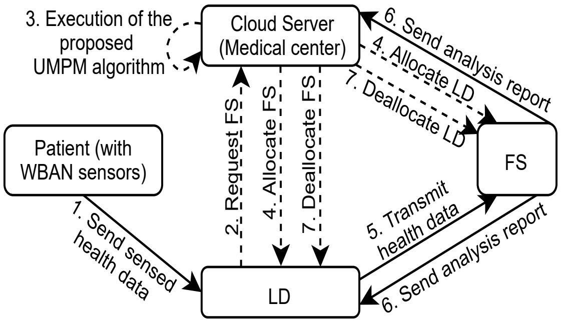


Fig. 3: Data flow diagram of the proposed architecture.

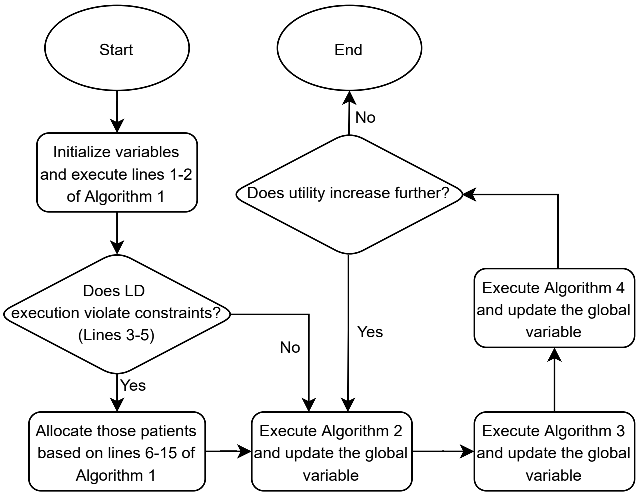

Start
End
No
Initialize variables and execute lines 1-2 of Algorithm 1
Does utility increase further?
Does LD execution violate constraints? (Lines 3-5)
Yes
No
Execute Algorithm 4 and update the global variable
Yes
Allocate those patients based on lines 6-15 of Algorithm 1
Execute Algorithm 2 and update the global variable
Execute Algorithm 3 and update the global variable

Fig. 4: Flow chart of UMPM algorithm.

**Algorithm 1:** UMPM Algorithm

```
   Input: $\mathbb{L}, \Upsilon, g, m, l, \delta, \mathbb{H}_t = \{0\}, \Omega$;
   $\forall p \in \mathbb{P}$: $\rho^c_{p,t}$, $\beta_{p,t}$, $\eta_{p,t}$; $\forall p \in \mathbb{P}, \forall f \in \mathbb{F}$: $SINR^f_{p,t}$;
   $\forall f \in \mathbb{F} : \mathbb{P}^f_t, n^f_t = 0$; $U^{max}_{diff,t} = -\infty$
   Output: Allocation Strategy ($\mathbb{H}_t$).
1  Calculate $n^{max}_{p,f,t}$, $\forall p \in \mathbb{P}$ and $\forall f \in \mathbb{F}$ using Theorem 1;
2  Calculate local computation time using Eq. (5);
3  for $p \leftarrow 1$ to $P$ do
4      if $T^{c,l}_{p,t} > \frac{\delta}{\rho^c_{p,t}}$ then
5          insert $p$ into $\mathbb{P}^v_t$;
6  Sort patients in set $\mathbb{P}^v_t$ in decreasing criticality order;
7  for $p \in \mathbb{P}^v_t$ do
8      for $f \in F$ do
9          if $n^f_t \geq \min_{p' \in \mathbb{P}^f_t \bigcup \{p\}} n^{max}_{p',f,t}$ then
10             continue;
11         Calculate $U_{diff,t}$ as per Eq. (35);
12         if $U_{diff,t} > U^{max}_{diff,t}$ and Eq. (29) satisfied then
13             $U^{max}_{diff,t} \leftarrow U_{diff,t}$;
14             $temp_p \leftarrow p$, $temp_f \leftarrow f$;
15     Allocate $temp_p$ to $temp_f$, update variable ; //  $h^f_{p,t}$
16 Run Algorithm 2;
17 Run Algorithm 3;
18 $\mathbb{P}^{rem} \leftarrow \mathbb{P} - \mathbb{P}^v_t$;
19 Run Algorithm 4;
20 repeat
21     Run Algorithm 2;
22     Run Algorithm 3;
23     Update $\mathbb{P}^{rem}$;
24     Run Algorithm 4;
25 until Utility does not increase;
```

allocation of other patients' data as it is as follows:

$$U_{diff,t} = \lambda_1(m - l - g\beta_{p,t}) - \lambda_2 \sum_{p' \in \mathbb{P}^f_t} \frac{\rho^c_{p',t}\beta_{p',t}}{\Gamma_f} + \lambda_2 \left(\rho^c_{p,t}\left(\frac{\beta_{p,t}}{\Upsilon} - \left(\frac{\beta_{p,t}(n^f_t + 1)}{\Gamma_f} + \frac{\eta_{p,t}}{BR^f_{p,t}}\right)\right)\right). \tag{35}$$

Set of patients whose data get allocated to FS $f$ is given by:

$$\mathbb{P}^f_t = \{p \in \mathbb{P} : h^f_{p,t} = 1\}. \tag{36}$$

Moreover, the number of patients whose data is allocated to FS $f$ can be estimated as follows:

$$n^f_t = \sum_{p \in \mathbb{P}} h^f_{p,t}. \tag{37}$$

The UMPM algorithm begins by calculating the constraint parameter $n^{max}_{p,f,t}$ (Eq. (31)). It selects the patients that violate latency constraint if their data is computed on LD and sorts them in the order of their decreasing criticalities (lines 3-6). As a result, the algorithm prioritizes the patients with higher criticalities over the lower criticality patients. Then, re-allocation is done using Algorithms 2 and 3. Then, the algorithm constructs the set of patients whose data is not yet allocated to any FS (line 18). It calls Algorithm 4 to allocate more patients' data to FSs. Then, Algorithms 2 and 3 re-position the allocation of patients' data to FSs. The execution order of the Algorithms 2 and 3 does not affect the outcome of UMPM algorithm (see Fig. 12a). Further, Algorithm 4 allocates more patients' data to FSs by improvising the utility. This process repeats until there is no possibility of increment in the utility (Fig. 4). The reason for having Algorithms 2 and 3 is to obtain better utility by swapping the allocation between patients and FSs as described in Subsections 4.2 and 4.3, respectively.

### 4.2 Two Way Swap based Algorithm

Utility difference due to two way swap, $J^{tr}_{diff,t}$[8] is given as:

$$J^{tr}_{diff,t} = \frac{\rho^c_{p,t}\eta_{p,t}}{BR^f_{p,t}} - \frac{\rho^c_{p,t}\eta_{p,t}}{BR^{f'}_{p,t}} + \frac{\rho^c_{p',t}\eta_{p',t}}{BR^{f'}_{p',t}} - \frac{\rho^c_{p',t}\eta_{p',t}}{BR^f_{p',t}} + \frac{\rho^c_{p,t}\beta_{p,t}n^f_t}{\Gamma_f} - \frac{\rho^c_{p,t}\beta_{p,t}n^{f'}_t}{\Gamma_{f'}} + \frac{\rho^c_{p',t}\beta_{p',t}n^{f'}_t}{\Gamma_{f'}} - \frac{\rho^c_{p',t}\beta_{p',t}n^f_t}{\Gamma_f}. \tag{38}$$

At each iteration, the algorithm picks a pair of patients whose data is allocated to different FSs (lines 2-7). Then checks, whether the number of patients' data allocated to

[8] When two patients' data allocated to different FSs are swapped, the change in utility is caused by the difference of their transmission and computation latencies, as considered in Eq. (38).

those two FSs is greater than the maximum number of patients that can be allocated to the two FSs (line 8). If it exceeds the maximum number of patients, then the algorithm picks another pair of patients and checks the condition in line 8. Else, it checks whether swapping the allocation of the two patients can increase utility or not (line 10). If utility can be increased, patients are swapped, and updates are taken place in the corresponding values (lines 11-12). This process repeats until no such pair of patients exist (lines 1-13).

**Algorithm 2:** Two Way Swap

```
   Input: Globally accessible $\mathbb{H}_t$ , Information of all
          patients (as per Algorithm 1) and FSs.
   Output: $\mathbb{H}_t$
1  repeat
2      for $f \leftarrow 1$ to $F$ do
3          for every $p \in \mathbb{P}_t^f$ do
4              for $f' \leftarrow 1$ to $F$ do
5                  if $f' == f$ then
6                      continue;
7                  for every $p' \in \mathbb{P}_t^{f'}$ do
8                      if $n_t^{f'} > n_{p,f',t}^{max}$ or $n_t^f > n_{p',f,t}^{max}$ then
9                          continue;
10                     if $J_{diff,t}^{tr} > 0$ then
11                         Swap $p$ and $p'$;
12                         Update corresponding values;
13 until No swap increases utility;
```

## 4.3 One Way Swap based Algorithm

Utility difference due to one way swap, $J_{diff,t}$[9] is given as:

$$J_{diff,t} = \frac{\rho_{p,t}^c \eta_{p,t}}{BR_{p,t}^f} - \frac{\rho_{p,t}^c \eta_{p,t}}{BR_{p,t}^{f'}} + \frac{\rho_{p,t}^c \beta_{p,t}(n_t^f - n_t^{f'} - 1)}{\Gamma_f} + \sum_{p' \in \mathbb{P}_t^f \setminus \{p\}} \frac{\rho_{p',t}^c \beta_{p',t}}{\Gamma_f} - \sum_{p' \in \mathbb{P}_t^{f'}} \frac{\rho_{p',t}^c \beta_{p',t}}{\Gamma_{f'}}. \tag{39}$$

At each iteration, algorithm picks a patient whose data is allocated to an FS (lines 2-3). It then selects another FS and calculates utility difference if the patient's data is allocated to that FS (lines 4-7). Then, it checks if allocating the patient's data to that FS increases utility (line 8). If utility is increased by satisfying the constraints, the patient's data is allocated to that FS and updates are taken place in the corresponding values (lines 9-10). This process repeats until no such patient exists (lines 1-11).

## 4.4 Patient-FS Allocation Algorithm

The algorithm selects a subset of patients and allocates their data to FSs that satisfy the constraints and maximize the utility. Algorithm 4 terminates when there is no improvement in the utility compared to utility obtained in previous iteration. The following Lemma 2 provides the utility correlation across different iterations of Algorithm 4.

**Algorithm 3:** One Way Swap

```
   Input: Globally accessible $\mathbb{H}_t$, Information of all
          patients (as in Algorithm 1) and FS.
   Output: $\mathbb{H}_t$
1  repeat
2      for $f \leftarrow 1$ to $F$ do
3          for every $p \in \mathbb{P}_t^f$ do
4              for $f' \leftarrow 1$ to $F$ do
5                  if $f' == f$ then
6                      continue;
7                  Compute $J_{diff,t}$ according to Eq. (39);
8                  if $J_{diff,t} > 0$ and
                   $n_t^{f'} + 1 \leq \min_{p' \in \mathbb{P}_t^{f'} \bigcup \{p\}} (n_{p',f',t}^{max})$ then
9                      Add $p$ to $\mathbb{P}_t^{f'}$ and remove $p$ from $\mathbb{P}_t^f$;
10                     Update the values correspondingly;
11 until No swap increases utility;
```

**Lemma 2.** *Let $U_{diff,t,i}^{max}$ be the $U_{diff,t}^{max}$ calculated by the algorithm at $i^{th}$ iteration at time $t$, then $U_{diff,t,i}^{max} \geq U_{diff,t,i+1}^{max}$. In other words, the maximum utility difference decreases with each iteration of Algorithm 4.*

*Proof.* Let $p$ and $p'$ be the patients whose data is allocated to FS $f$ at $i^{th}$ iteration and FS $f'$ at $(i+1)^{th}$ iteration, respectively. Then, consider the following two cases:

*Case 1: $f = f'$*

$$U_{diff,t,i}^{max} = \lambda_1(m - l - g\beta_{p,t}) - \lambda_2 \sum_{p'' \in \mathbb{P}_t^f} \frac{\rho_{p'',t}^c \beta_{p'',t}}{\Gamma_f} + \lambda_2 \left( \rho_{p,t}^c \left( \frac{\beta_{p,t}}{\Upsilon} - \left( \frac{\beta_{p,t}(n_t^f + 1)}{\Gamma_f} + \frac{\eta_{p,t}}{BR_{p,t}^f} \right) \right) \right). \tag{40}$$

$$U_{diff,t,i+1}^{max} = \lambda_1(m - l - g\beta_{p',t}) - \lambda_2 \sum_{p'' \in \mathbb{P}_t^f \bigcup p} \frac{\rho_{p'',t}^c \beta_{p'',t}}{\Gamma_f} + \lambda_2 \left( \rho_{p',t}^c \left( \frac{\beta_{p',t}}{\Upsilon} - \left( \frac{\beta_{p',t}(n_t^f + 2)}{\Gamma_f} + \frac{\eta_{p',t}}{BR_{p',t}^f} \right) \right) \right), \tag{41}$$

where $n_t^f$ is the number of patients utilizing $f$ before $i^{th}$ iteration at time $t$ and similarly $\mathbb{P}_t^f$ is the set of such patients. On subtracting Eq. (40) from Eq. (41), it is clear that,

$$U_{diff,t,i+1}^{max} \leq U_{diff,t,i}^{max}. \tag{42}$$

*Case 2: $f \neq f'$*

In this case, as both the FSs are different, if $U_{diff,t,i+1}^{max}$ would have been greater than $U_{diff,t,i}^{max}$, then the algorithm would have picked $p'$ at the $i^{th}$ iteration only, but that is not the case. Hence, $U_{diff,t,i+1}^{max} \leq U_{diff,t,i}^{max}$.

From both cases, $U_{diff,t,i+1}^{max} \leq U_{diff,t,i}^{max}$. Hence, proved. □

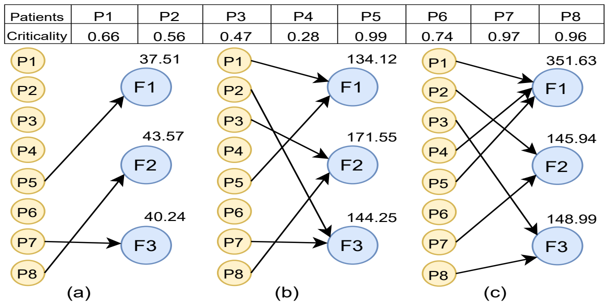
[9] When a patient $p$'s data is reallocated to $f'$ from $f$, the profit does not change. The change in the cost is calculated as the difference between the transmission times when $p$'s data allocated to $f$ and $f'$. The computation time of all patients' data allocated to $f$ and $f'$ changes.

**Algorithm 4:** Patient-FS Allocation

```
Input: Globally accessible $\mathbb{H}_t$, Set of Patients, $\mathbb{P}^{rem}$,
       $flag = 0$, $temp_p$, $temp_f$, $U^{max}_{diff,t} = 0$
Output: Allocation Strategy ($\mathbb{H}_t$).
repeat $|\mathbb{P}^{rem}|$ times
    for every $p \in \mathbb{P}^{rem}$ do
        if $q_{p,t}$ == 1 then
            continue;
        for $f \leftarrow 1$ to $F$ do
            if $n^f_t \geq \min_{p' \in \mathbb{P}^f_t \bigcup \{p\}} n^{max}_{p',f,t}$ then
                continue;
            Calculate $U_{diff,t}$ as per Eq. (35);
            if $U_{diff,t} > U^{max}_{diff,t}$ then
                $flag \leftarrow 1$;
                $U^{max}_{diff,t} \leftarrow U_{diff,t}$;
                $temp_p \leftarrow p$, $temp_f \leftarrow f$;
    if $flag == 0$ then
        break
    Assign patient $temp_p$ to FS $temp_f$;
    Update $n^{temp_f}_t$, $\mathbb{P}^{rem}$ and $\mathbb{H}_t$;
```

### 4.5 Illustration of UMPM Algorithm

Let there be 8 patients and 3 FSs in the system. We consider a constant data size of 2 MB and needed CPU cycles as 600 for each patient. Computation capacity of FSs, F1, F2, and F3 are considered as 22 GHz, 18 GHz, and 20 GHz, respectively. In Fig. 5, yellow and blue colors indicate patients and FSs, respectively. The patients' criticalities are shown at the top of the figure. These values are calculated based on simulation Table 4 and Eqs. (1)-(3). Moreover, the costs of patients' data allocations are labeled on respective FSs in Fig. 5 after the execution of UMPM algorithm.



| Patients | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 |
|---|---|---|---|---|---|---|---|---|
| Criticality | 0.66 | 0.56 | 0.47 | 0.28 | 0.99 | 0.74 | 0.97 | 0.96 |

Fig. 5: (a) Initial allocation (lines 1-17 of Algorithm 1). (b) Allocation after execution of Algo. 4 (lines 18-19 of Algo. 1). (c) Final allocation (lines 20-25 of Algo. 1).

Fig. 5(a) shows an initial allocation of patients' data to FSs (lines 1-17 of Algorithm 1), in which data of patients P5, P7, and P8 is allocated to F1, F3, and F2, respectively. Algorithm 4 allocates remaining patients' data to FSs for further improvising the system utility. Thus, we get an outcome as shown in Fig. 5(b) after the execution of Algorithm 4 (lines 18-19 of Algorithm 1). Then, after reiterating Algorithms 3, 2, and 4 (lines 20-25 of Algorithm 1), Fig. 5(c) is obtained as final allocation with a maximized overall system utility.

### 4.6 Analysis of Proposed Heuristic

This section discusses convergence and time complexity of UMPM algorithm as follows.

**Lemma 3.** *The proposed UMPM algorithm converges.*

*Proof.* Convergence of UMPM relies on convergence of three sub-algorithms. Algorithms 3 and 2 swap patients only if utility increases. Else, their execution terminates. As the total possible combinations between patients and FSs are finite, utility will also be a finite value. Thus, both the swap algorithms converge. Algorithm 4 converges since the number of iterations is finite, i.e., $P$. Further, UMPM repeatedly executes two way swap, one way swap, and Patient-FS allocation algorithms. Every iteration converges, and the algorithm goes to next iteration only when utility increases. Therefore, UMPM algorithm converges. □

**Theorem 2.** *Time complexity of UMPM algorithm is $O(P^2F)$.*

*Proof.* Time complexity of UMPM depends on complexity of three sub-algorithms it calls. Time complexity of Algorithm 1 from lines 1-15 is $O(PF)$. Algorithm 2 considers $P^2$ pairs of patients and repeats until it converges. Thus, the number of iterations is bounded by a finite value. Hence, time complexity of Algorithm 2 is $O(P^2)$. Similarly, time complexity of Algorithm 3 is $O(PF)$ as it considers $PF$ number of possible swaps. In Algorithm 4, the number of iterations is bounded by number of patients, i.e., $P$. In each iteration, $PF$ pairs are considered. Thus, time complexity of Algorithm 4 is $O(P^2F)$. Hence, time complexity of proposed UMPM algorithm is $O(P^2, PF, P^2F)$, i.e., $O(P^2F)$. □

Simulation setup and obtained results are discussed in the following section:

## 5 Performance Study

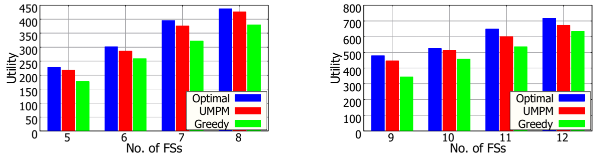
Patient's data size and required CPU cycles for computing patient's data are randomly considered between [1, 3] MB and [100, 1000] Megacycles, respectively [11]. The value of $\delta$ is taken as 250 ms. The patients' criticalities are considered between [0, 1], as shown in Table 4. Moreover, we have considered 5 physiological sensors' data such as body temperature, heart rate, blood pressure, respiration rate and blood oxygen saturation for evaluation purposes. We used HElib open-source library [30] for implementing homomorphic encryption in our proposed model. Simulation experiments are performed using Windows 10 Laptop with Intel(R) i7-10750H @ 2.60 GHz processor and 16 GB memory.

We could not compare the proposed model with other existing works because none of the existing works have considered patients' criticality, profit of MC and other constraints altogether as per best of our knowledge (see Table 2). Gurobi optimization tool is used to get the optimal solution [33]. Moreover, we offered *Greedy* scheme to compare with the proposed model. In greedy scheme, patients that violate latency constraint are sorted in decreasing order of their criticality. Then, patients' data is allocated to FSs one by one. The remaining patients are sorted in decreasing

This article has been accepted for publication in IEEE Transactions on Services Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSC.2022.3206770

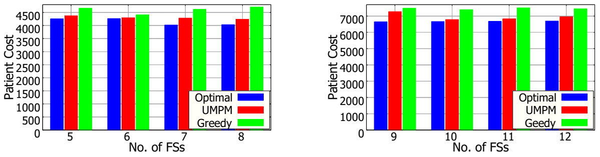
TABLE 4: Simulation parameters

| Parameter | Value |
|---|---|
| $P$, $F$ | [20-1000], [2-200] |
| $\eta_{p,t}$ [11] | [1, 3] MB |
| $\beta_{p,t}$ [11] | [100, 1000] Megacycles |
| Blood pressure $[\theta_{l,s}, \theta_{u,s}]$ [31] | 91 mmHg, 169 mmHg |
| Body temperature $(\theta_{l,s}, \theta_{u,s})$ [31] | 34.1 $°C$, 37.9 $°C$ |
| Heart rate $(\theta_{l,s}, \theta_{u,s})$ [31] | 51 bpm, 139 bpm |
| Respiration rate $(\theta_{l,s}, \theta_{u,s})$ [31] | 11, 29 breath/min |
| Oxygen saturation $(\theta_{l,s}, \theta_{u,s})$ [32] | 95 %, 100 % |
| $\delta$, $\rho^c_{p,t}$, $\Upsilon$ | 250 ms, [0, 1], 2.4 GHz |
| $l$, $m$, $g$, $k$ | 100, 200, 0.1, 0 units |
| $\Omega$, $\Gamma_f$ [11] | [5-15] MHz, [18-22.4] GHz |
| $SINR^f_{p,t}$, $V^f_{p,t}$ | [13-20] dB, [5-15] |
| $F^{max}_f$, $C_{max}$ | [3-15], [200-1400] |
| $\lambda_1$, $\lambda_2$ | 0.5, 0.5 |



(a) No. of patients = 40 (b) No. of patients = 60

Fig. 6: Utility comparison among different schemes.



(a) No. of patients = 40 (b) No. of patients = 60

Fig. 7: Patients' cost comparison among different schemes.

order of criticality, and the allocation process repeats until no improvement takes place in the utility.

**System Utility Analysis:** Fig. 6 considers two cases where the numbers of patients are 40 and 60, and the number of FSs varies from 5 to 12. We observe from the result that the utility increases as the number of FSs increases. UMPM algorithm performs better than the greedy scheme. The utility obtained by UMPM algorithm is 94.5% of the optimal value compared to 77% that of greedy scheme on an average. The reason is that the greedy scheme allocates patients' data in a particular order. Although the greedy scheme considers patients' criticality, it ignores data size and CPU cycles of health data. UMPM algorithm considers all the above factors to reach a sub-optimal utility within polynomial time complexity. Moreover, the optimal solution performs better than the UMPM algorithm because it considers all possible combinations of allocations and selects the best out of them while maximizing the utility.

**Patient Cost Analysis:** Fig. 7 compares the patients' cost of the UMPM algorithm in various scenarios. We observe that UMPM algorithm generally results in lower patients' cost than that of greedy scheme because UMPM considers different parameters, and it allocates and re-allocates patients' data to maximize the utility by minimizing patients' cost. However, greedy scheme does not re-allocate patients' data, resulting in higher patients' cost and lower utility. We also observe that patients' cost obtained by UMPM algorithm is higher than that of the optimal because the optimal solution considers all possible combinations of allocations and selects the best out of them to maximize the utility.

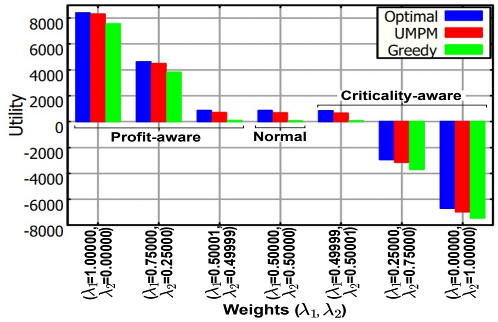


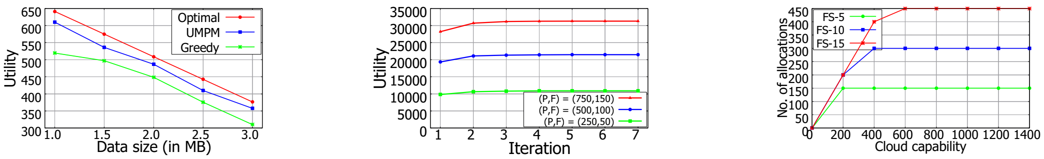
Fig. 8: Utility based on $\lambda_1$ and $\lambda_2$.

**Impact of $\lambda_1$ and $\lambda_2$ on Utility:** Fig. 8 compares the utility of optimal, UMPM, and greedy schemes, taking precision level of 5 decimal points of lambda values. Normal system balances profit and cost factors equally, i.e., $\lambda_1$=$\lambda_2$=0.5. Criticality-aware system concerns more about cost factor than profit, i.e., $\lambda_1 < \lambda_2$. Experimental evaluation on precision level of 5 decimal points of lambda values gives highest utility when $\lambda_1$=0.49999 and $\lambda_2$=0.50001. In fact, the obtained profit and loss factors are the highest and the lowest respectively, when $\lambda_1$=0.49999 and $\lambda_2$=0.50001 on precision level of 5 decimal points; resulting in a higher utility. Therefore, criticality-aware system reaches its optimal value when $\lambda_2$ is closed to 0.5 and $\lambda_1 < \lambda_2$. Moreover, profit-aware system concerns more about the profit factor than the cost, i.e., $\lambda_1 > \lambda_2$. Profit-aware system reaches its optimal value when $\lambda_1$=1 and $\lambda_2$=0. The reason is that with increase in $\lambda_1$ and decrease in $\lambda_2$, profit and loss factors become higher and lower, respectively, resulting in a better utility value. Furthermore, UMPM algorithm performs better than greedy in all cases and is closer to optimal value. Therefore, UMPM can be applied efficiently in profit-aware system.

**Impact of Data Sizes on Utility:** Fig. 9a shows the impact of data sizes on system utility. We observe that the system utility decreases as the data size increases. The utility is higher when the patient's data size is small (i.e., 1.0 MB). However, the utility is lower when the patient's data size is large (i.e., 3.0 MB). It is because the required CPU cycles to compute patients' data rise, and transmission time and FS's computation time increase as the data size grows. As a result, the patients' cost increases, resulting in lower utility.

**Convergence Analysis:** Fig. 9b depicts convergence of the UMPM algorithm for three different cases. We observe from the result that the algorithm converges in a few iterations in all three cases. We also observe that the utility increases as the number of patients and FSs increases, as does the number of iterations because the number of possible combinations between patients and FSs increases. However, after certain iterations, the increase in utility is very small and converges in finite iterations for all three cases. Thus, UMPM algorithm converges in finite time.

**Patients' Data Allocation Analysis:** Fig. 9c shows the number of allocations based on the cloud capability ranging



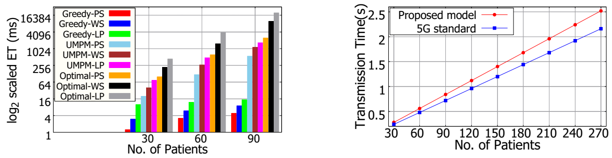
(a) Utility v/s Data sizes (b) Convergence analysis (c) Patients allocated

Fig. 9: Utility v/s data sizes, convergence and allocation analysis.

from 200 to 1400 patients' requests at a time. We considered 30 FSs, and the number of patients that an FS can handle ranges from 5 to 15. We observe from result that number of allocated patients' data increases when CS's capability rises, but after reaching total capacity of FSs, it becomes constant.

**Execution and Transmission Time Analysis:** Fig. 10a compares the Execution Time (ET) of optimal, UMPM and greedy schemes in various scenarios using Laptop (LP), Workstation (WS), and the Param Shivay (PS) super computer [34]. For optimal, we consider the ET required by the Gurobi optimization tool to run the proposed heuristic. To improve the readability, the y-axis in Fig. 10a is given after applying the $log_2$ scale to ET. We observe that the greedy scheme completes its execution in less time than that of the optimal and the UMPM approaches in all three machines. However, the utility obtained by the greedy scheme is much lower than that of the UMPM algorithm (see Fig. 6). Moreover, the optimal solution takes more time than that of the UMPM algorithm since it considers all possible combinations of allocations and selects the best out of them while maximizing the utility. We also observe that the ET of the UMPM algorithm is different for different machines. This result shows that the ET depends on machine's configuration, i.e., the configuration of the CS available at MC.

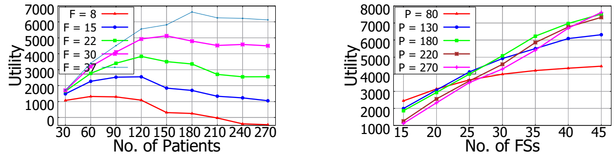
Fig. 10b shows the transmission time of the proposed model and the 5G standard. We considered 1 MB data size for each patient and 1 FS to simulate the transmission time. We observe that the transmission time of the proposed model is little higher than that of the 5G standard. The reason for the slight variation is that the proposed model achieves slight less data rate (Eq. (9)) between patients and FS than that of the standard method (1 Gbps [35]) due to limited consideration of available PRBs in the system [29].



(a) Execution time (b) Transmission time

Fig. 10: Execution and transmission time analysis.

**System Utility using Real World Data:** Fig. 11 compares the system utility of UMPM algorithm using a real dataset on different settings. We use the Statlog (Heart) dataset [36] which contains 13 attributes and has total of 270 data samples. In our work, we considered three attributes (i.e., blood pressure, ECG and heart rate) as the data collected by sensors from patients. Other simulation parameters are considered the same as given in Table 4.

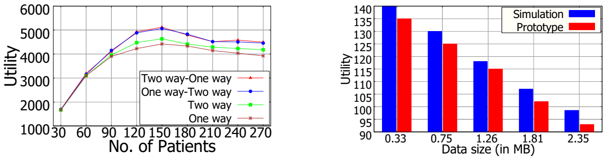


(a) Utility v/s No. of FSs (b) Utility v/s No. of patients

Fig. 11: Utility comparison of UMPM on various parameters.

Fig. 11a considers five cases where the numbers of FSs are 8, 15, 22, 30, and 37, and the number of patients varies from 30 to 270. We observe that the utility increases when the number of patients increases, but the utility decreases after a certain number of patients. It is because, up to a certain number of patients, the total computation required for the patients is less than the total fog computation power. However, if we increase the number of patients furthermore, the required computation increases but the total fog computation power remains the same. This leads to higher computation time, resulting in higher costs and lower utility. We also observe that the utility is higher for the higher number of FSs. It is because the computation time decreases since total computation power increases as the number of FSs increases, resulting in lower costs.

Fig. 11b considers five cases where numbers of patients are 80, 130, 180, 220, and 270, and the number of FSs varies from 15 to 45. We observe from result that the utility increases when the number of FSs increases because the total fog computation power rises, but the required CPU cycles of patients' data remain the same. Thus, FSs can compute patients' data in less computation time. This leads to lower costs, resulting in higher utility. We also observe that the utility is lower for the higher number of patients. The reason is that computation time increases since the number of patients increases, but total computation power remains the same, resulting in higher costs.



(a) Execution order analysis (b) Utility using prototype

Fig. 12: Execution order and utility analysis.

**Execution Order Analysis:** Fig. 12a analyses the execution order of swap Algorithms 2 and 3. We use the Statlog (Heart) dataset as discussed above. From the result, we observe that the utility obtained by performing two way

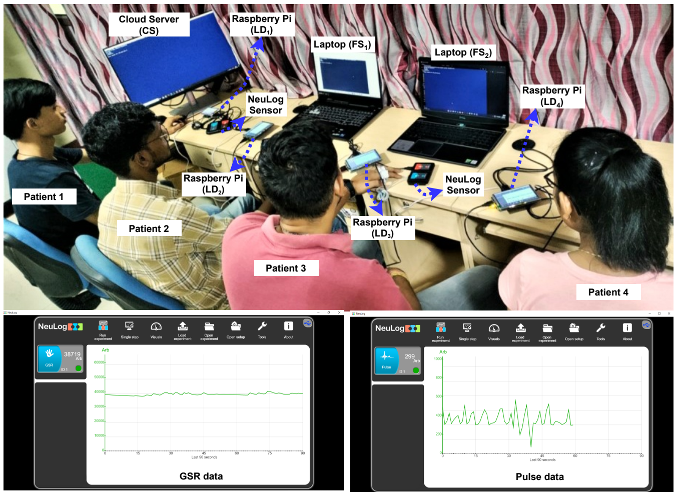


Fig. 13: Prototype setup.

swap followed by one way swap algorithm is almost the same as that of performing one way swap followed by two way swap algorithm. Thus, UMPM algorithm can perform two way and one way swap algorithms in any order. As seen from the result, execution of one way swap or two way swap algorithms one after another increases the overall system utility. We also observe that the utility obtained by performing two way swap algorithm is better than that of performing only one way swap algorithm.

**Utility and Encryption Analysis:** We use a WS as CS, 2 LPs as FSs, 4 NeuLog sensors, and 4 Raspberry Pi as LDs, as shown in Fig. 13. The WS's specification is Core-i7-10700 CPU @ 4.10 GHz processor and 32 GB memory, and that of LPs is same as discussed above. We use Raspberry Pi 4 Model B with 1.5 GHz 64-bit quad-core, ARM Cortex-A72 CPU, and 8GB memory. NeuLog sensor is connected with Raspberry Pi using a USB. Raspberry Pi, LP and WS are connected with a 4G mobile hotspot. NeuLog sensor collects data from the patient and sends it to Raspberry Pi, and Raspberry Pi sends a request to WS for remote health monitoring. The WS allocates patients' data to the LP by executing the UMPM algorithm. Then, Raspberry Pi transmits the health data to the LP for further processing.

UMPM has been evaluated on data sizes varying from 0.33 MB to 2.35 MB (Fig. 12b). The result denotes the utility obtained in simulation is higher than that of the prototype model. It is because the prototype model used 4G mobile hotspot for transmitting data between Raspberry Pi and LP, unlike the 5G communication used in simulation. Moreover, Raspberry Pi, LPs, and WS use in the prototype model are not dedicated same configured devices as considered in simulation set-up. This leads to higher transmission and computation time, resulting in higher costs and lower utility.

Table 5 shows the impact of BGV in terms of data size and encryption/decryption time at the LD. We observed from experiment that encryption and decryption overhead time is very low, as shown in Table 5a. As a result, the BGV encryption scheme can be employed in practice. We also observe an increase in original data size after employing the BGV encryption scheme, as can also be seen in Table 5b.

TABLE 5: BGV encryption analysis

| Operation | Time (ms) |
|---|---|
| Encryption | 22.99785 |
| Decryption | 1.00088 |

(a) Time

| | Data size |
|---|---|
| Before encryption | 84 bytes |
| After encryption | 192 bytes |

(b) Data size

## 6 Conclusion and Future Work

This paper proposed a beyond-WBAN based fog assisted remote health monitoring system. Formulated an optimization problem based on the MC's profit and patients' cost. Further, proposed UMPM algorithm to maximize overall system utility. Through extensive simulations on real-world data and prototype model, the paper concludes that UMPM algorithm achieves an average utility of 94.5% of the optimal value in polynomial time complexity.

This work considers CS to execute UMPM algorithm for allocating patients' data to FSs. However, future work will consider compute heavy tasks such as data analysis and trend analysis at the cloud level in coordination with FS to minimize total computation delay. Moreover, the proposed utility maximization model can be extended to incorporate the sub-channel allocation problem considering interference into account. The role of doctors can be included in the proposed system by employing appropriate pricing model and energy minimization for WBAN-enabled system along with latency and patients' criticality altogether in the future.

**Acknowledgments:** We would like to thank the Associate Editor and anonymous reviewers for insightful comments that helped us improve the quality of the manuscript significantly. This work was supported by the Science and Engineering Research Board (SERB), Government of India (GoI) under Grant SRG/2020/000318. The support and the resources provided by PARAM Shivay Facility under the National Supercomputing Mission, GoI at IIT (BHU), Varanasi are gratefully acknowledged.

## References

[1] F. Wu, C. Qiu, T. Wu, and M. R. Yuce, "Edge-Based Hybrid System Implementation for Long-Range Safety and Healthcare IoT Applications," *IEEE IoT Jour.*, pp. 1–1, 2021.

[2] R. M Abd El-Aziz *et al.*, "An Effective Data Science Technique for IoT-Assisted Healthcare Monitoring System with a Rapid Adoption of Cloud Computing," *Comput. Intell. Neurosci.*, vol. 2022, 2022.

[3] S. Misra and S. Sarkar, "Priority-based time-slot allocation in wireless body area networks during medical emergency situations: An evolutionary game-theoretic perspective," *IEEE Jour. Biomed. Health Inform.*, vol. 19, no. 2, pp. 541–548, 2014.

[4] A. Pratap and S. K. Das, "Stable Matching based Resource Allocation for Service Provider's Revenue Maximization in 5G Networks," *IEEE Trans. Mob. Comput.*, pp. 1–1, mar 5555.

[5] P. K. Bishoyi and S. Misra, "Enabling Green Mobile Edge Computing for 5G-Based Healthcare Applications," *IEEE Trans. on Green Comm. and Net.*, 2021.

[6] P. Deshingkar, *Migration, remote rural areas and chronic poverty in India*. ODI, 2010.

[7] A. A. Mutlag *et al.*, "Enabling technologies for fog computing in healthcare IoT systems," *Future Gener. Comput. Syst.*, vol. 90, pp. 62–78, 2019.

[8] M. Shu, D. Yuan, C. Zhang, Y. Wang, and C. Chen, "A MAC protocol for medical monitoring applications of wireless body area networks," *Sensors*, vol. 15, no. 6, pp. 12906–12931, 2015.

[9] S. Yi, C. Li, and Q. Li, "A survey of fog computing: concepts, applications and issues," in *Proceedings of the 2015 workshop on mobile big data*, pp. 37–42, 2015.

[10] M. Kamruzzaman *et al.*, "Blockchain and Fog Computing in IoT-Driven Healthcare Services for Smart Cities," *Jour. of Healthcare Engineering*, vol. 2022, 2022.

[11] Z. Ning *et al.*, "Mobile edge computing enabled 5G health monitoring for Internet of medical things: A decentralized game theoretic approach," *IEEE Jour. Sel. Areas Commun.*, 2020.

[12] L. Feng *et al.*, "Optimal haptic communications over nanonetworks for E-health systems," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 3016–3027, 2019.

[13] L. Gu *et al.*, "Cost efficient resource management in fog computing supported medical cyber-physical system," *IEEE Trans. Emerg. Topics Comput.*, vol. 5, no. 1, pp. 108–119, 2015.

[14] H. K. Apat, K. Bhaisare, B. Sahoo, and P. Maiti, "Energy Efficient Resource Management in Fog Computing Supported Medical Cyber-Physical System," in *2020 ICCSEA*, pp. 1–6, IEEE, 2020.

[15] Y. Qiu *et al.*, "Computation Offloading and Wireless Resource Management for Healthcare Monitoring in Fog-Computing based Internet of Medical Things," *IEEE IoT Jour.*, 2021.

[16] C. Yi and J. Cai, "Transmission management of delay-sensitive medical packets in beyond wireless body area networks: A queueing game approach," *IEEE Trans. Mob. Comput.*, vol. 17, no. 9, pp. 2209–2222, 2018.

[17] C. Yi and J. Cai, "A priority-aware truthful mechanism for supporting multi-class delay-sensitive medical packet transmissions in e-health networks," *IEEE Trans. Mob. Comput.*, vol. 16, no. 9, pp. 2422–2435, 2016.

[18] S. Misra *et al.*, "A cooperative bargaining solution for priority-based data-rate tuning in a wireless body area network," *IEEE Trans. Wirel. Commun.*, vol. 14, no. 5, pp. 2769–2777, 2015.

[19] P. K. Bishoyi and S. Misra, "Enabling collaborative data uploading in body-to-body networks," *IEEE Commun. Lett.*, vol. 25, no. 2, pp. 538–541, 2020.

[20] S. Patro and K. K. Sahu, "Normalization: A preprocessing stage," *arXiv preprint arXiv:1503.06462*, 2015.

[21] B. Di *et al.*, "Sub-channel assignment, power allocation, and user scheduling for non-orthogonal multiple access networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7686–7698, 2016.

[22] A. Pratap *et al.*, "On maximizing task throughput in IoT-enabled 5G networks under latency and bandwidth constraints," in *2019 IEEE SMARTCOMP*, pp. 217–224, IEEE, 2019.

[23] A. Pratap, R. Gupta, V. S. S. Nadendla, and S. K. Das, "Bandwidth-constrained task throughput maximization in IoT-enabled 5G networks," *Pervasive Mob. Comput.*, vol. 69, p. 101281, 2020.

[24] T. Tanwar, U. D. Kumar, and N. Mustafee, "Optimal package pricing in healthcare services," *Jour. Oper. Res. Soc.*, vol. 71, no. 11, pp. 1860–1872, 2020.

[25] S. Rajagopalan and C. Tong, "Payment models to coordinate healthcare providers: Extension to risk-averse providers," *Oper. Res. Lett.*, vol. 49, no. 3, pp. 326–332, 2021.

[26] H. R. Lewis, "Computers and intractability. A guide to the theory of NP-completeness," 1983.

[27] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "Fully Homomorphic Encryption without Bootstrapping." Cryptology ePrint Archive, Report 2011/277, 2011. https://ia.cr/2011/277.

[28] S. Halevi and V. Shoup, "Algorithms in helib," in *Annual Cryptology Conference*, pp. 554–571, Springer, 2014.

[29] A. Pratap and S. K. Das, "Stable Matching based Resource Allocation for Service Provider's Revenue Maximization in 5G Networks," *IEEE Trans. Mob. Comput.*, 2021.

[30] S. Halevi, "HElib: An Implementation of homomorphic encryption." https://github.com/shaih/HElib. Accessed: 2022.

[31] J. J. Kang, T. H. Luan, and H. Larkin, "Inference system of body sensors for health and internet of things networks," in *Proceedings of the 14th MoMM'16*, pp. 94–98, 2016.

[32] Charles Patrick Davis, "Safe, Normal, Low Blood Oxygen Levels: Pulse Oximeter Chart." https://www.onhealth.com/content/1/normal_low_blood_oxygen_pulse_oximeter_levels, 2022.

[33] Gurobi Optimization, LLC, "Gurobi Optimizer Reference Manual." https://www.gurobi.com, 2022.

[34] IIT (BHU), "Param Shivay Supercomputing Center, IIT (BHU), Varanasi." https://www.iitbhu.ac.in/cf/scc, 2022.

[35] N. Wang, E. Hossain, and V. K. Bhargava, "Backhauling 5G small cells: A radio resource management perspective," *IEEE Wireless Commun.*, vol. 22, no. 5, pp. 41–49, 2015.

[36] D. Dua and C. Graff, "UCI machine learning repository." http://archive.ics.uci.edu/ml, 2017.

**Moirangthem Biken Singh** completed the B.Tech degree in Computer Science and Engineering from the National Institute of Technology Manipur, India, in 2018 and the M.Tech degree from National Institute of Technology Kurukshetra, India, in 2021. He is currently pursuing Ph.D. degree in Computer Science and Engineering, IIT (BHU), Varanasi, India. His current research interest include AI/ML and FL in Smart Healthcare.

**Navneet Taunk** is an undergraduate student with the Department of Computer Science and Engineering, IIT (BHU), Varanasi, India. His research interests include Design of Algorithms, Internet of Things (IoT) and Mathematical Modelling.

**Naveen Kumar Mall** is an undergraduate student with the Department of Computer Science and Engineering at the Indian Institute of Technology (BHU), Varanasi, India. His research interests include Deep learning, Machine learning and IoT.

**Ajay Pratap** is an Assistant Professor with the Department of Computer Science and Engineering, IIT (BHU), Varanasi, India. His research interests include Cyber-Physical Systems, IoT-enabled Smart Environments, Statistical Learning, and AI/ML for Next-G Cellular Networks. His papers appeared in several international journals and conferences including IEEE Transactions on Mobile Computing, IEEE Transactions on Parallel and Distributed Systems, IEEE Transactions on Vehicular Technology, etc. He has received several awards including the Best Paper Award and NSF travel grant for IEEE SmartComp conference in the USA.