International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2021/209983 Volume 8, Issue 5, September – October (2021)

RESEARCH ARTICLE

An Energy Aware Data Scheduling Approach in Cloud Using GK-ANFIS

Sampath Kumar Y R
Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bengaluru, Karnataka, India
yrsam008@gmail.com

Champa H N
Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bengaluru, Karnataka, India
champahn@yahoo.co.in

Received: 20 July 2021 / Revised: 08 September 2021 / Accepted: 13 September 2021 / Published: 27 October 2021

Abstract – HealthCare (HC) applications are vital and also time-sensitive. Due to the Internet of Things (IoT) technology's capability to enhance the quality and efficiency of treatments, multiple HC applications were implemented through it to augment the patients' health. IoT technology comprises of scheduling methodologies, which makes it intricate to self-configure and self-adapt to respond with respect to the environmental changes. Prevailing scheduling techniques don't consider allocating tasks via sleep modes that consecutively bring about additional power consumption in addition to long time delays. Here, an energy-efficient as well as activity aware management framework called Gaussian Kernel-based Adaptive Neuro-Fuzzy Inference System (GK-ANFIS) is proposed for IoT devices on the cloud. The proposed work follows data filtering, Features Extraction (FE), Features Selection (FS), along with scheduling of IoT data. The proposed work allows the distribution of HC data of the patients to the proper Cloud Server (CS) of hospital admin through the implementation of GK-ANFIS centered scheduling along with allocation approach. The proposed method is implemented and its performance is analyzed. The outcomes rendered exhibit that the proposed techniques execute better when weighed against other existing algorithms.

Index Terms – Adaptive Neuro-Fuzzy Inference System (ANFIS), Cloud Computing, Internet of Things (IoT), Scheduling.

1. INTRODUCTION

The interconnection of physical domains and electronics has gained remarkable progress, which results in growth of IoT applications [1]. The IoT lets individuals and things to be interconnected by utilizing any path/network as well as any service [2]. Numerous sensor devices say pressure, gyro, temperature, together with humidity sensors are present in the IoT, which encompass a vital part [3]. The environmental data that are sensed and collected via these sensors are analyzed to launch appropriate actions, reduce social and economic losses, and it also minimizes the intricacy (hardware along with man-power) [4]. IoT technology is being utilized in numerous applications, say Smart agriculture, cities, transportation, grids and HC, and also in novel inventory system [5].

IoT-HC is one amongst the main application domains. The IoT can be implemented in HC utilizing these three ways: a) tracking of objects together with individuals (medical team, workers, as well as patients), b) automatic data acquisition and sensing, and c) transfering gathered data to the cloud [6]. In HC applications, to capture patient's health data, say heart rate, blood pressure and so on, small wearable devices are deployed [7]. HC-IoT mainly relies on Cloud Computing (CC) services, which could render disparate sorts of services on-demand, say computation resources, networks, storage capabilities, together with higher processing services that can fulfil IoT needs [8][9].

Currently, for processing an enormous quantity of data that is gathered by the IoT devices, they are consigned to the centralized cloud. This is done prior to getting back the outcome as of the cloud to data consumers that are always located adjoining to the original data sources [10]. One amongst the key challenges on the operation of complex cloud data centers is the Energy Consumption (EC), which occur on account of the huge quantity of data and the escalating total users [11]. In order to minimize the EC, CS providers utilize scheduling along with Resource Allocation (RA) algorithms that schedule the arriving data and also manage the resources. The data scheduling together with RA is an imperative issue on account of the fairness in scheduling and RA that makes certain the Quality of Services (QoS) standards and aids in reducing the production expense via capably exploiting the accessible resources [12][13].

RESEARCH ARTICLE

The cloud system utilizes the virtualization technique, which enables several Virtual Machines (VM) placements on an individual physical machine. Additionally, it reinforces live relocation of VM, which enhances resource usage and also aids in reducing energy utilization [14]. The effective scheduling of data to cloud requires VM instances of the required resources [15]. Recently, numerous scheduling algorithms, say machine learning (ML) and deep learning (DL) algorithms, Adaptive Neuro-Fuzzy Inference Systems (ANFIS) algorithm, disparate intelligent heuristic in addition to metaheuristic algorithms were introduced [16][17][18]. Since HC-IoT is a crowd-sensing domain and it collects a huge quantity of raw data, enhancement is still required in those algorithms for handling and scheduling IoT data. Therefore, the conventional scheduling algorithms must be enhanced to effectually schedule IoT requests for fulfilling the users' expectations [19]. This paper proposes an enhanced scheduling algorithm aimed at energy-efficient scheduling.

1.1. Problem Statement

To implement an effective scheduling technique on cloud in order to optimize various system resources like memory utilization, energy consumption, and CPU usage.

1.2. Motivation

Energy is a critical factor in the crowd sensing domain, where it is difficult to maintain the sensing infrastructure. Processing and storing of data is expensive and directly associated to the computational resource requirements. Handling unwanted data requires more energy and processing time. Existing Schedulers utilize the classification and optimization algorithm. Thus scheduling time is comparatively large and there is a need for optimised scheduling strategy which minimizes various computational resources.

1.3. Objectives

The proposed work focusses to attain the following goals.

- It provides a data filtering technique for pre-processing the IoT data.
- A new MK-means scheme is introduced to improve the scheduling accuracy and reduce the scheduling time.
- A GK-ANFIS centred IoT data scheduler in the cloud environments aims at optimizing the performance of HC applications.
- Testing and evaluation of the proposed and existing techniques' performance considering a few performance metrics like f-measure, recall, precision, accuracy, etc.

1.4. Organization

The paper is systematized in a subsequent way: Section 2 renders the associated papers. Section 3 elucidates, in brief, the proposed process. Section 4 exhibits the experimental setup and performance measure of the proposed technique, and finally, conclusions have been rendered in section 5.

2. RELATED WORKS

Randa M. Abdelmoneem *et al.* [20] presented the mobility-aware heuristic-centered scheduling and allocation Technique (MobMBAR) aimed at the IoT HC system in the CC platform. Based on the patients' movements and their sensed data's temporal/spatial residual, MobMBAR balanced the task execution's distribution in a dynamic manner. MobMBAR's goal was to reduce the scheduled time by employing the task features like the task's critical level and the maximal response time at the ranking and reallocation phases. The experimental outcomes demonstrated that MobMBAR possesses a distinctively lesser make-span analogized to the Health-Edge, HEFT, and Cloud-Only platforms. Nevertheless, this technique needs to improve the fault tolerance, scalability, and also mobility.

Abishi Chowdhury *et al*. [21] presented the technique which intended to overcome the issue of scheduling of bulk IoT requests. The sufficient resources for the requests were allocated via the CC and data mining methods whilst guaranteeing a good quality of service to the IoT users. A good monitoring system had been utilized for gathering the data related to service requests; then befitting information extraction procedure was employed to acquire valuable patterns of the collected data. Later, this information was utilized to schedule IoT requests to suitable cloud servers. The outcomes exhibited large improvement in the system's performance based on few major QoS parameters. The major demerit was it does not apt for much intricate IoT applications that are prevalent in the cloud environment.

Weiwei Lin *et al.* [22] suggested two IoT-aware multiple-resources task scheduling technique aimed at various cloud environments, like core resources load balancing along with time balancing. The technique aimed for attaining a good outcome of IoT task responses time, Service-Levels Agreement (SLA), as well as load balance; meanwhile, it also aimed at minimizing the EC to the maximal extent. They were developed for allotting a single task to a correctly elected Virtual Machine (VM). The task positioned in the pre-processed queue had been sequentially allotted every time. Depending on the recently invented ideas termed the relative load or relative time cost, the VM selection rule had been executed. Outcomes exhibited that the proffered techniques efficiently improved the SLA and load balance, whilst the time balancing was better at conserving energy and time.

Xiaojin Ma *et al.* [23] proffered the cost-aware genetic optimization (DCAGO) technique to schedule the IoT data in the cloud environment. To reduce the cost beneath the deadline constraint, the technique concentrated on the cloud's

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2021/209983 Volume 8, Issue 5, September – October (2021)

RESEARCH ARTICLE

vital features, like acquisition delay, VMs' performance variation, on-demand acquisition, and heterogeneous dynamics. Moreover, the presented technique employed the heuristic procedures aimed at allotting tasks for suitable VM. The experimental outcomes determined that the DCAGO technique attained less execution costs under diverse constraints. However, the VM failure and task reassignment problems prevalent in the DCAGO couldn't be resolved; these factors considerably affected the DCAGO's reliability and scalability.

Yiping Wen *et al.* [24] developed the energy as well as cost-aware technique aimed at the instance-intensive IoT workflows' scheduling utilizing batch processing in the clouds that was termed ECIB and to improve the energy efficiency and decrement the execution cost whilst fulfilling the various necessities. The ECIB technique was established for the pondered issue that comprised the prediction centered stratagem which aimed at guiding the resource management. The batch processing stratagem combined few activity instances of the identical type and this schemes aimed at adding the correct amount of resources, consequently the EC is optimized. Experimental examinations were executed for verifying the ECIB's efficiency and the outcomes revealed that the ECIB was not appropriate in a multi-cloud environment.

Mukhtar M.E. Mahmoud *et al.* [25] integrated Cloud with IoT termed as the Cloud of Things (CoT) which aimed at fulfilling the IoT necessities. The IoT's capabilities (for instance: sensing) had been provisioned as services on the CoT. Fog Computing introduced in CoT aimed at assisting the IoT services and also its applications. An energy-aware allocation stratagem was established and aimed at positioning the application modules on the Fog devices. Later, by utilizing the iFogSim simulator, the CoT stratagem's performance was examined in analogy with the default allocation and also the Cloud-only policies. The suggested solution was found to be highly energy-efficient, saving about 2.72% energy analogized to the Cloud and about 1.6% energy analogized to the Fog platform.

Chunlin Li *et al.* [26] established a two-level scheduling optimization technique in the edge cloud environment to schedule the IoT data. The 1st-level scheduling utilized the artificial fish swarm-centered job scheduling protocol; most of the data was scheduled onto the edge data centres. If the edge data centre doesn't possess sufficient resources to execute, the data might be scheduled onto the centralized cloud data centre. In the 2nd-level scheduling, the edge cloud scheduling was employed for decrementing the completion time, whilst the centralized cloud-task scheduling was implemented to decrement the total expense. The outcomes exhibited that the proposed technique attained minimal latency and less time. However, it offered less user satisfaction whilst the maximal execution cost had been minimized. Table 1 summarizes and compares various existing works related to IoT data scheduling.

The survey above exhibits the different protocols aimed at the scheduling of IoT HC data which are prevalent in the cloud. Most schemes largely concentrate on optimization techniques to perform IoT data scheduling. Nevertheless, they didn't employ the clustering techniques aimed at the scheduling procedure in the cloud. Thus it led to least network life-time, maximum EC, and the efficient resource utilization just for scheduling outcomes in maximum processing time. Consequently, for overcoming such demerits and for promoting energy-aware IoT data scheduling into the cloud, this work put forward new techniques to cluster and schedule data.

Authors	**Research Concepts**	**Benefits**	**Disadvantages**
Charith Perera *et al.*[2]	Context-aware Mobile Sensor Data Engine(C-MOSDEN)	Location and activity aware platform is capable of capturing sensors' data, on-demand.	Context-aware functionalities generate a small overhead. It needs to improve data privacy and capabilities of data analytics.
Randa M. Abdelmoneem *et al.* [20]	Mobility-aware heuristic-centered scheduling and allocation Technique (MobMBAR).	Execution of tasks were distributed dynamically	The technique fails to improve the fault tolerance, scalability, and also mobility.
Abishi Chowdhury *et al.* [21]	Technique to overcome the scheduling of bulk IoT requests.	It guarantees a good quality of service to the IoT users.	It is not suitable for intricate IoT applications prevalent in the cloud environment.

RESEARCH ARTICLE

Weiwei Lin *et al.* [22]	IoT-aware multiple-resources task scheduling technique aimed at various cloud environments.	Proffered techniques efficiently improved the (Service level agreement) SLA and load balance.	It needs to improve IoT task responses time and time balancing.
Xiaojin Ma *et al.* [23]	Cost-aware genetic optimization (DCAGO) method to schedule the IoT data in the cloud.	It requires less execution cost.	VM failure and task reassignment problems couldn't be resolved.
Yiping Wen *et al.*[24]	Developed the energy as well as cost-aware technique (ECIB).	The scheduling technique utilized batch processing in the clouds.	The method was not suitable for a multi-cloud environment.
Mukhtar M.E. Mahmoud *et al.* [25]	Introduced the concept of Cloud of Things (CoT) aimed at fulfilling the IoT requirements.	The method is energy-efficient.	CoT needs fog computing to be implemented in the cloud.
Chunlin Li *et al.* [26]	Two-level scheduling technique in the edge cloud environment to schedule the IoT data.	Proposed technique attained minimal latency and time.	Cost of computation needs to be improved.

Table 1 Comparison of Various Existing Works on Data Scheduling

3. PROPOSED METHODOLOGY

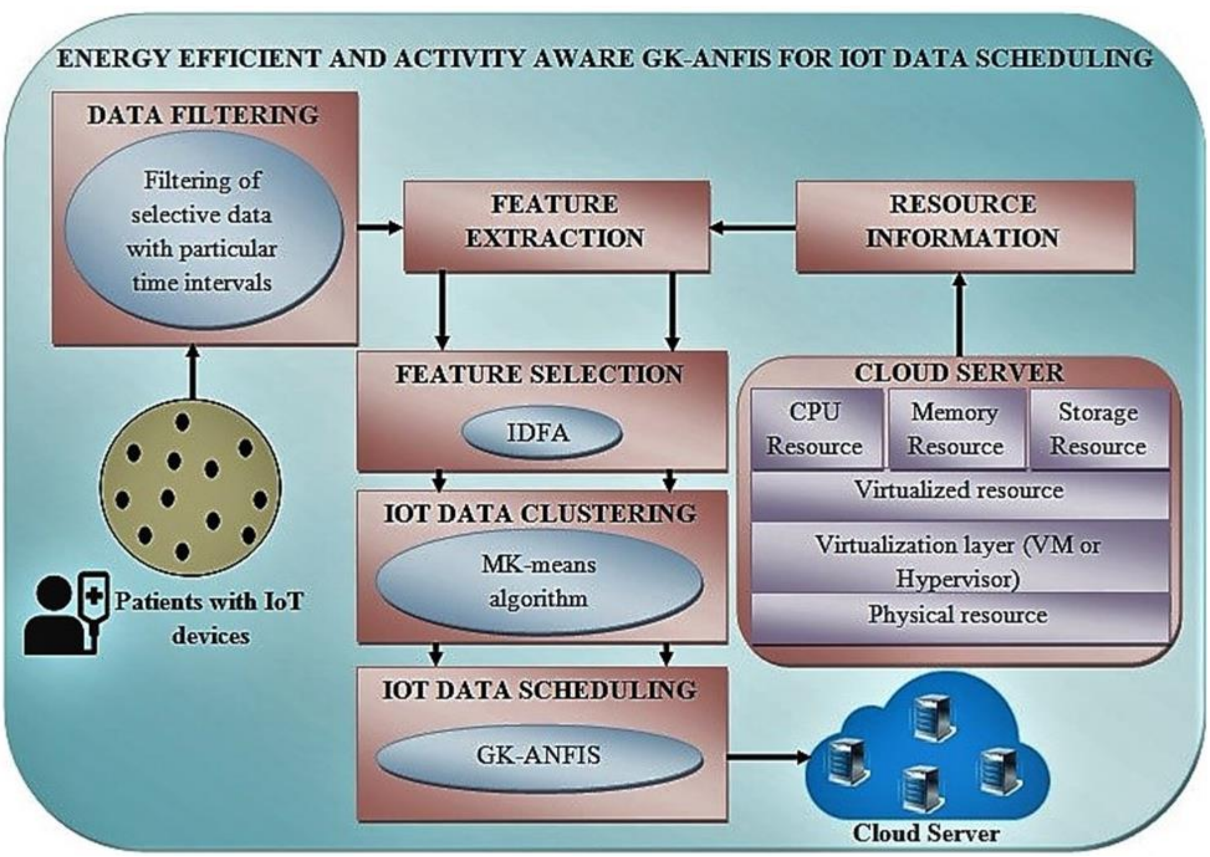

Figure 1 Proposed Architecture

RESEARCH ARTICLE

This paper introduces a GK-ANFIS algorithm and proposes a scalable as well as effectual data analytics framework intended for on-demand based mobile crowd-sensing. The proposed work comprises of five phases: data filtering, FE, FS, clustering, and scheduling. The sensor devices are attached to disparate patients, from which the IoT sensor values are taken. The data filtering approach filters the sensor values by filtering out the data of the patients at a specific time interval. The system only senses the IoT data in three scenarios, say environmental monitoring, rehabilitation, and health and well-being in terms of the time feature in the dataset. Subsequent to data filtering, the FE phase is done. Here, the feature is extracted as of the CS together with the IoT sensor values. Subsequently, the IDFA algorithm selects the necessary features from extracted features. Next, the MKM algorithm clusters the features. Lastly, subsequent to clustering the IoT data, they are scheduled utilizing the GK-ANFIS algorithm. The proposed method's architecture is exhibited in Figure 1.

3.1. Data Filtering

Data filtering is the initial step. Here, depending upon the time interval, selective IoT data is filtered from the patients' data and this filtration is done whilst the patient is performing some important tasks, namely walking, jogging, exercise, etc. The data reading extracted from their internal organs during that time is different from that of which is captured at normal time. Thus, it is essential to filter the patients' data whilst they perform selective tasks. Thus, at first, the data filtering process with a corresponding time interval is executed in several scenarios (Scenario 1 - environmental monitoring, Scenario 2 - rehabilitation, Scenario 3 -health along with well-being). Therefore, in these scenarios, the system only senses the patients' IoT data within a specific time interval.

3.2. Feature Extraction

It is crucial to extract the features from the sensed IoT data after the end of data filtering. Disparate IoT devices attached to patients within a specific time interval, sense enormous amount of data. Thus, data collision might occur and the energy would be wasted throughout that time. Utilizing the schedulers, the filtered IoT data from different patients are scheduled to the proper cloud server to resolve this issue. A huge quantity of patient's data is handled by hospitals and the patients might be suffering from high BP, heart related issues and diabetic disorders. Every patient's feature might differ and the separate cloud server for every category of patient is managed by the hospitals. Thus, allocating the patient's IoT data onto the hospital's correct cloud server is essential, thus the essential features are extracted as of both the IoT together with the cloud server. The extraction of features or resource information namely memory and storage resource, CPU resources, number of requests, Processing speed and cycle, bandwidth, disk space, load on VM, along with CPU usage of the cloud server are considered. Additionally, certain features are extracted from patient's attached IoT sensor device which include timestamp, activity name, time interval, patient id and details, protocol name, etc.

3.3. Feature Selection

After FE by utilizing the IDFA, the vital features are selected from those extracted features. The DFA algorithm is inspired from the dragonflies' (DFs') dynamic as well as static swarming behaviors in nature. The two essential phases of exploration, exploitation and also optimization are prepared by designing the DFs social interaction in navigating, looking for food, and also evading enemies whilst swarming in a dynamic or else static manner. This algorithm's demerit is it gets trapped easily within a few local optimum solutions due to premature convergence. Aimed at upgrading the updating process's accuracy in the existent DFA, the Genetic Algorithm phases like Mutation and Cross-over, are hybridized utilizing the DFA before the updation procedure. The proposed method employs the technique in which every individual present in the existent iteration will undergo cross-over and also mutation procedures if the DF has at least one DF in the neighborhood velocity. Consequently, this improved DFA version is called IDFA that pursues three principles.

i) Separation that signifies the individuals' static collision avoidance against other individuals present in the neighbourhood.

ii) Alignment, which signifies the individuals' velocity matching to that of the other individuals present in the neighbourhood.

iii) Cohesion that specifies the individuals' tendency to the centre of the neighbourhood's mass.

Step 1: Any swarm's core aim is survival; therefore, every individual must be attracted to food resources and distracted from the enemies. Pondering the two behaviours, five core factors prevail in the individuals' position updation in swarms. The factors are: separation, attraction, alignment, cohesion, and also distraction. Each factor is arithmetically designed as: the separation is enumerated as:

$$S_i = -\sum_{j=1}^{N} F - F_j \quad (1)$$

Equation (1) represents the separation factor where F signifies the present individual's position, F_j implies the j th adjacent individual's position, and N symbolizes the number of neighbouring individuals.

Step 2: Alignment is formulated as shown in Equation (2).

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2021/209983 Volume 8, Issue 5, September – October (2021)

RESEARCH ARTICLE

$$A_i = -\frac{\sum_{j=1}^{N} V_j}{N} \quad (2)$$

Here in equation (2), V_j signifies the j^{th} adjacent individual's velocity.

Step 3: The cohesion is equated as indicated in Equation (3).

$$C_i = -\frac{\sum_{j=1}^{N} F_j}{N} - F \quad (3)$$

Step 4: Attraction to a food resource is enumerated as shown in Equation (4).

$$R_i = F^+ - F \quad (4)$$

Here, F signifies the present individual's position; F^+ signifies the food source's position.

Step 5: Distraction from the enemy is enumerated as formulated in Equation (5).

$$E_i = F^- + F \quad (5)$$

Here, F signifies the present individual's position; F^- implies the enemy's position.

Step 6: Examine the novel solutions by employing the cross-over and also mutation procedure, whilst the DF comprises at least one DF in its neighbourhood, which makes the optimization much efficient. The cross-over kind implemented here is termed the two-point cross-over. It is executed by employing the crossover points as shown in Equation (6) and Equation (7).

$$c_1 = \frac{F_i^{t+1}}{3} \quad (6)$$

$$c_2 = c_1 + \frac{\left|F_i^{t+1}\right|}{2} \quad (7)$$

Here c_1 and c_2 symbolizes the two points which are chosen as the cross-over points.

Step 7: The mutation has been done by the swapping of genes from individual chromosome with the newer ones. The genes swapped comprises of the arbitrarily generated ones which do not possess any repetition within the chromosome. Here, the chromosomes represent the parameters' compilation that determines the solution.

Step 8: For updating the artificial DFs' position in the search space and aiming at simulating their movements, two vectors are basically considered: step ΔF and position F. The step vector presents the direction of the DFs' movements and it is determined as indicated in Equation (8) (note that the artificial DFs' position updating design is determined in one dimension; however, the introduced methodology can be prolonged to high dimensions):

$$\Delta F_{t+1} = \left(sS_i + aA_i + cC_i = rR_i + eE_i\right) + \omega \Delta F_t \quad (8)$$

Here, in equation (8), the $s, a, and\ c$ values signify the separation, alignment, and cohesion coefficients, correspondingly; $r, e, \omega\ and\ t$ values imply the food factor, enemy factor, inertia coefficient, and iteration number, correspondingly.

Whereas if there is no neighbouring individual, then the DFs' behaviour is presumed to be a random walk (Levy-flight) around the search place for incrementing the randomness, exploration and stochastic behaviour. The DF's position is updated as represented in Equation (9).

$$F_{t+1} = F_t + Levy(d) * F_t \quad (9)$$

Here, t signifies the existent iteration; d implies the position vectors' dimension. Algorithm 1 represents the pseudocode for proposed IDFA.

Input: Extracted Features of IoT and Cloud Server
Output: Optimized Features
Begin
Initialize the population of dragonflies as $F_i = \left(f_i^1, f_i^d, ... f_i^N\right)$
Initialize the step vectors
While the stopping criteria is not met
Compute the fitness function for all dragonflies
Initialize the weights of S, A, C, R and E randomly for all dragonflies
Update the position and velocity of dragonflies by computing S_i, A_i, C_i, R_i and E_i as indicated from Equation(1)-(5)
Compute the distance of the neighbourhood
If the dragonfly has at least one corresponding dragonfly in the neighbourhood

RESEARCH ARTICLE

Update Velocity and Position by performing crossover and mutation as shown in Equation (8)
Else
Update velocity and position vector by performing Levy Walk as shown in Equation (9)
End if
Check and rectify the new positions based on the boundaries of variables
End while
End

Algorithm 1 IDFA Based Feature Selection

3.4. Clustering

After FS, the IoT data from various patients are clustered utilizing the MK-means technique. The clustering is executed for grouping the similar IoT data of patients. For instance, IoT devices sense the various patients' data. The patients might be suffering from heart related issues, abnormal BP, diabetes, etc. The clustering of the patients' IoT data is executed to identify and clarify the same type. The clustering is executed depending on the features selected. Based on the features, the data clustering is executed. The proposed technique utilizes the MK-means technique for clustering. K-means clustering is one of the simplest and well-known unsupervised machine learning technique. This technique detects the k-number of centroids; then, allots every data-point to the closest cluster, whilst maintaining the number of centroids as minimal as possible. But the algorithm failed to accurately cluster the sensor nodes (SN) at the 1st iteration, thus, the existing algorithms require more number of iterations for achieving the finest clusters. To improvise the clustering performance, the proposed method utilized Minkowski distance in the normal k-means clustering technique. Consequently, the proposed algorithm which aids in sensor data's cluster formation on the WSN is termed as the MK-Means technique. The MK-means' technique is explained as:

Step 1: Initialize the n number of sensor data sensed from the diverse IoT devices as $S_i = \{S_1, S_2, ...S_n\}$ and c-number of cluster centres as $K_j = \{K_1, K_2, ...K_c\}$.

Step 2: Elect c cluster centres arbitrarily from the initialized K_i.

Step 3: Allot the SN to the cluster centre whose distance from the cluster centre is the minimal of all cluster centres according to the Minkowski distance function as depicted in Equation (10).

$$M_D = \left(\sum_{i=1}^{n} \sum_{j=1}^{c} \left| S_i - K_j \right|^{\alpha} \right)^{1/\alpha} \tag{10}$$

Here, α implies the positive real-number.

Step 4: Recompute the novel cluster centre by taking the mean of every sensor data allotted to that centroid's cluster, therefore reducing the total intra-cluster's variance in association with the preceding step.

Step 5: Recalculate the distance between every sensor data and the newly attained cluster centres.

Step 6: The algorithm iterates as of 3rd step till certain criteria are satisfied (e.g. the summation of distances between the SNs and their respective centroid is decreased. A maximal number of iterations are attained; no modifications in the centroids' values or else no data-points alter clusters).

3.5. Scheduling

Here, the proposed work's final step is proffered. Utilizing the GK-ANFIS algorithm, the clustered IoT sensor data is scheduled. The optimized features of cloud server and the clustered IoT data are fed to the GK-ANFIS. The clustered data are scheduled to the corresponding cloud server centered on the IoT data and cloud server's features. GK-ANFIS algorithm is employed to schedule clustered data to corresponding cloud server. ANFIS is a kind of artificial neural network, which is centred upon Takagi–Sugeno fuzzy inferences system. It holds the potential for capturing the advantages of both systems in a single frame as it combines the neural networks along with fuzzy logic principles. This system corresponds to a group of fuzzy IF-THEN rules which possess learning ability aimed at estimating nonlinear functions. ANFIS's architecture encompasses five layers. The input value is taken by the first layer and it establishes the Membership Functions (MFs) belonging to them and it is usually known as the fuzzification layer. For creating the rules' Firing Strengths (FSs), 2nd layer is employed and it is termed as "rule layer". For normalizing the calculated FS by dividing every value aimed at the total FS, 3rd layer is employed. The consequent parameter set and the normalized values are taken as input by the 4th layer. The value returned by this layer are defuzzificated ones. For attaining the final output, these values are inputted to the end layer. Bell MF is employed by the typical ANFIS strategy. But, for ameliorating the performance of the rule generation process, the proposed method utilized the Gaussian kernel (GK) Membership Function (MF). Thus the ANFIS is termed as GK-ANFIS. The below Equations (11) and (12) specifies two basic rules (R_1 & R_2) of ANFIS that are aimed at two inputs D_1 along with D_2.

Rule 1: If D_1 is P_1 as well as D_2 is Q_1,

$$R_1 = r_1 D_1 + s_1 D_2 + t_1 \tag{11}$$

RESEARCH ARTICLE

Rule 2: If D_1 is P_2 and D_2 is Q_2,

$$R_2 = r_2 D_1 + s_2 D_2 + t_2 \qquad (12)$$

Here, P_1, Q_1, P_2 along with Q_2 specifies the fuzzy sets. D_1 along with D_2 values specifies the disparate patients' clustered IoT data obtained from MK-means algorithm. r_1, s_1, t_1, r_2, s_2 & t_2 values indicate the set of parameters. The layers of ANFIS are explained as,

Layer-1: Here, each individual node is considered as an adaptive node possessing a node function as shown in equation (13)

$$O_{1,i} = \chi_{S_i}(D_1) \qquad (13)$$

Here, D_i signifies the input to the node i. Every node conforms to a function parameter. The resultant of every node is regarded as a member-ship value rendered by the MF's input. The proposed system employed the Gaussians kernel MF and it is articulated in the below equation(14).

$$\chi_{S_i} = \exp\left(-\frac{\|r_i - s_i\|^2}{2t_i^2}\right) \qquad (14)$$

Where $i = 1,2$

Here, the MF parameters which could alter the MF's shape are r_i, s_i and t_i and $\exp(.)$ signifies the particular value's exponential function. The parameters are adverted to as the assumed parameters.

Layer 2: Here, every node is a Fixed Node (FN) marked as ANFIS, whose output is the incoming signals' product and it is shown in equation (15).

$$G_i = \chi_{S_i}(D_1) \times \chi_{S_i}(D_2), i = 1,2 \qquad (15)$$

The layer's output $O_{2,i}$ signifies the rule's FS.

Layer 3: Here, each node is an FN marked N. The ratio of firing strength (FS) of i th rule to the addition of every rules' FS is computed by the i^{th} node as indicated in equation (16).

$$\overline{G}_i = \frac{G_i}{G_1 + G_2}, i = 1,2 \qquad (16)$$

Here G_i signifies the i^{th} rule's FS, G_1 along with G_2 signifies the initial along with second rule's FS. Each layer's output is defined as normalized FSs.

Layer 4: Here, every node is basically an adaptive node possessing a node function which is represented as shown in Equation (17).

$$O_{4,i} = \overline{G}_i . R_i = \overline{G}_i.(r_i D_1 + s_i D_2 + t_i) \qquad (17)$$

$\overline{G}_i$ denotes the normalized FS of 3rd layer whereas r_i, s_i, and t_i signifies parameter set of this node.

Layer 5: Here, a single node is a FN marked ANFIS that computes the whole output by adding every incoming signal. With the two data values (D_1 & D_2) as the input, the algorithm is designed. Utilizing the GK-ANFIS model, the data values as of every cluster are classified similarly as articulated in Equation (18).

$$O_{5,i} = \sum_{i=1}^{n} \overline{G}_i R_i \qquad (18)$$

4. RESULT AND DISCUSSION

For scheduling the user data on the cloud with optimal energy consumption, an effectual methodology say GK-ANFIS is proposed. The proposed method's implementation is done in JAVA by utilizing the cloudsim simulator. The MHEALTH [27] dataset is employed for proposed GK-ANFIS's execution. The MHEALTH (Mobile HEALTH) dataset encompasses body motion along with vital signs-recording of 10 volunteers with disparate profiles whilst executing various physical activities. The dataset consists of body motion and vital signs recordings for 10 volunteers of diverse profile while performing 12 physical activities (Table.2). The brackets indicate the number of repetitions (Nx) or the duration of the exercises (min).

Activity set	Description
L1	Standing still (1 min)
L2	Sitting and relaxing (1 min)
L3	Lying down (1 min)
L4	Walking (1 min)
L5	Climbing stairs (1 min)
L6	Waist bends forward (20x)
L7	Elevation of arms (20x)
L8	Knees bending (20x)
L9	Cycling (1 min)
L10	Jogging (1 min)
L11	Running (1 min)
L12	Jumping front & back (20x)

Table 2 Activity Dataset

RESEARCH ARTICLE

The various sensors are located on the subject's right wrist, chest and left ankle for the sake of measuring the acceleration, the rate of turn, along with magnetic field orientation. The sensor which is located on the chest gives ECG measurements which could be possibly utilized for basic monitoring of the heart. Here, considering various metrics, the proposed GK-ANFIS's outcomes are analogized with the prevailing C-MOSDEN. The intention of the initial three sequences of experiments is to comprehend the context-aware functionality's effect on memory, CPU, along with energy consumption. The proposed work adopts the strategy of selective sensing in scheduling the IoT data to a cloud server. Thus the performance of cloud server is improved when compared with existing techniques.

The main purpose is to comprehend how much additional computational resources are needed by the proposed GK-ANFIS whilst context-aware capabilities are in the act. GK-ANFIS is configured and aimed at collecting sensor data and pushing it to the cloud in 1s intervals in these experiments and the experiments are executed for 30 min. The proposed techniques' outcomes are analogized with the prevailing technique's outcomes. Table.3 depicts the values of CPU consumption which are obtained after simulation. Considering CPU consumption values which are illustrated in Table. 3, Figure.2 is plotted for the comparison of the CPU consumption of existing and proposed techniques. In figure 2, 3 and 4 the WOCA+AS, WOCA+AIS, WCA+AS along with WCA+AIS, signifies without context-awareness including accelerometer sensor, without context-awareness including all inbuilt sensors, with context awareness including accelerometer sensor and with context-awareness including all inbuilt sensors. By varying the time-interval between 1 to 30 min, CPU, memory, along with energy consumption of the proposed along with prevailing techniques are plotted.

Time(min)	Existing C-MOSDEN WOCA+AS	Proposed GK-ANFIS WOCA+AS	Existing C-MOSDEN WOCA+AIS	Proposed GK-ANFIS WOCA+AS	Existing C-MOSDEN WCA+AIS	Proposed GK-ANFIS WCA+AS
1	13	11	31	29	43	39
2	15	13	33	30	47	43
3	17	14	35	32	46	42
4	21	18	32	30	46	42
5	21	17	32	30	41	38
6	17	15	34	31	45	42
7	18	15	35	31	44	42
8	19	14	36	32	46	42
9	18	14	31	29	47	43
10	17	13	39	37	46	42
11	18	14	38	36	46	42
12	19	15	37	34	47	43
13	20	15	36	34	48	44
14	17	13	36	34	44	41
15	18	16	31	28	46	42
16	18	16	35	31	45	43
17	17	14	35	31	44	41
18	16	14	33	30	44	39
19	18	13	34	30	42	39
20	22	20	34	30	45	42
21	22	20	36	32	46	44
22	17	15	37	35	47	43
23	18	15	35	31	42	39
24	17	14	36	32	44	41
25	16	13	32	29	44	41
26	18	15	42	38	45	42
27	19	14	31	29	43	39
28	20	15	32	28	41	38
29	17	15	33	28	45	41
30	16	14	34	31	43	39

Table 3 CPU Consumption

RESEARCH ARTICLE

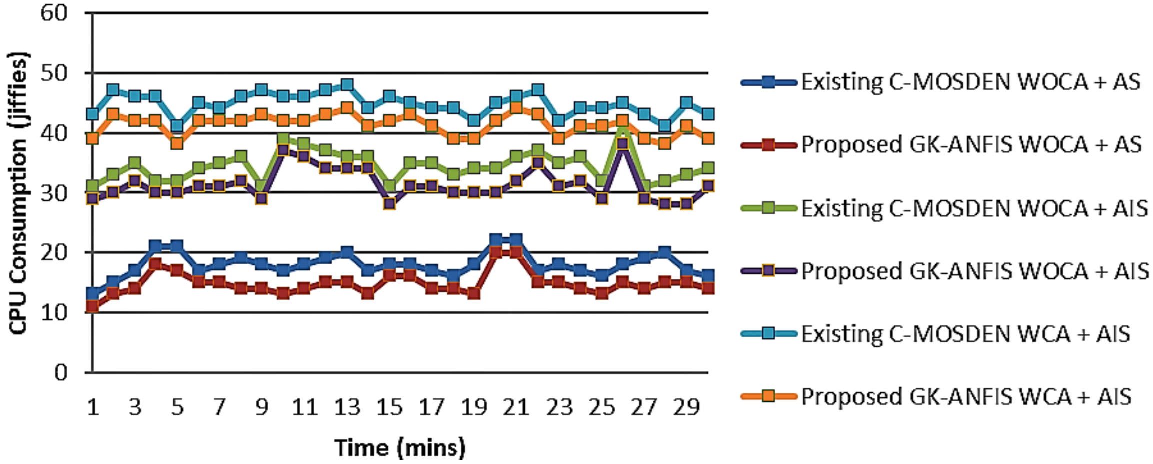

Figure 2 CPU Consumption

Time (min)	Existing C-MOSDEN WOCA+AS	Proposed GK-ANFIS WOCA+AS	Existing C-MOSDEN WOCA+AIS	Proposed GK-ANFIS WOCA+AS	Existing C-MOSDEN WCA+AIS	Proposed GK-ANFIS WCA+AS
1	13	11	31	29	43	40
2	15	11	33	30	47	43
3	17	13	35	32	42	40
4	21	18	32	30	46	44
5	21	18	32	30	41	38
6	17	15	34	32	45	42
7	18	16	35	31	48	46
8	19	15	36	31	46	44
9	18	15	33	30	47	43
10	17	14	39	37	46	42
11	18	15	38	36	41	38
12	19	16	37	35	47	44
13	20	17	36	34	48	44
14	17	14	36	34	49	45
15	18	16	39	35	46	42
16	18	15	35	32	45	43
17	17	15	35	32	47	43
18	16	13	33	30	44	41
19	18	15	34	31	42	39
20	22	19	32	28	45	42
21	22	19	36	31	42	40
22	17	14	37	33	47	44
23	18	15	35	31	42	40
24	17	14	36	32	41	38
25	16	12	35	31	44	40
26	18	13	33	30	45	41
27	19	15	31	28	45	41
28	20	17	32	29	43	39
29	17	13	33	29	44	39
30	16	13	34	30	41	38

Table 4 Energy Consumption

RESEARCH ARTICLE

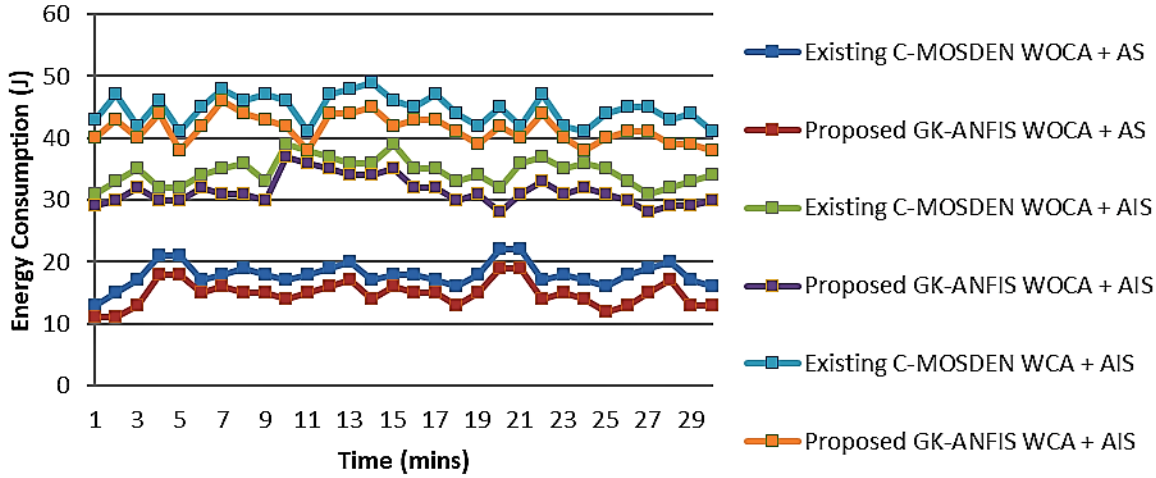

Figure 3 Energy Consumption

Time(min)	Existing C-MOSDEN WOCA+AS	Proposed GK-ANFIS WOCA+AS	Existing C-MOSDEN WOCA+AIS	Proposed GK-ANFIS WOCA+AS	Existing C-MOSDEN WCA+AIS	Proposed GK-ANFIS WCA+AS
1	62	59	76	72	109	106
2	60	58	77	73	108	106
3	63	61	78	75	107	103
4	68	63	78	75	116	112
5	66	61	85	81	120	117
6	64	61	82	80	118	115
7	68	63	80	78	117	114
8	61	57	78	76	116	113
9	74	71	77	75	111	108
10	72	70	85	81	110	107
11	65	61	79	76	108	106
12	53	50	83	80	110	107
13	61	59	80	78	98	96
14	74	71	89	87	97	93
15	71	68	85	81	93	89
16	68	65	77	75	92	87
17	53	50	80	78	97	94
18	61	58	75	73	100	96
19	74	71	78	75	102	98
20	71	68	85	82	99	97
21	61	59	89	86	95	93
22	53	48	86	83	101	97
23	61	58	79	76	102	98
24	74	71	79	76	106	103
25	72	68	85	82	107	104
26	63	60	80	78	109	105
27	68	66	84	82	108	105
28	61	58	82	80	105	102
29	74	70	80	77	100	87
30	71	69	85	83	99	96

Table 5 Memory Consumption

RESEARCH ARTICLE

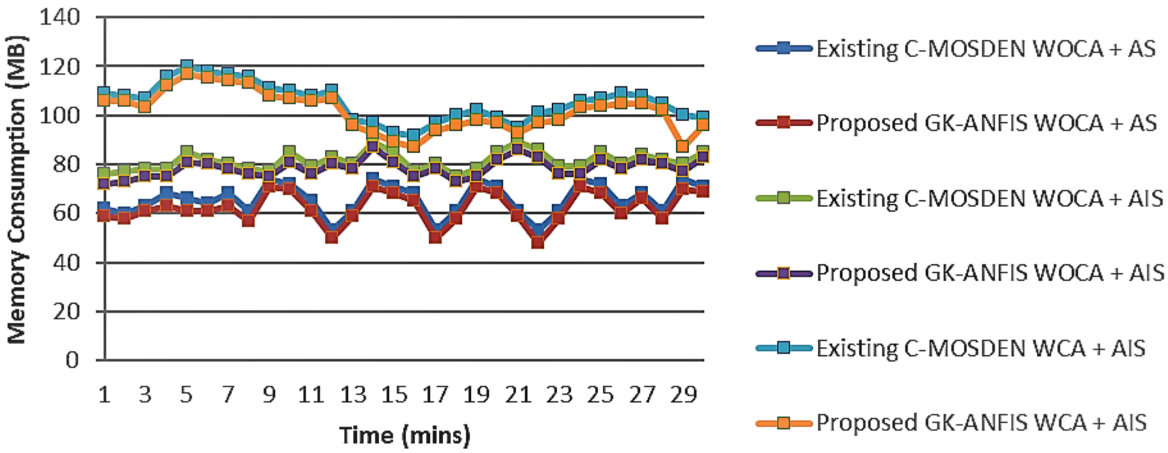

Figure 4 Memory Consumption

Considering energy consumption values obtained in Table.4, Figure 3 is plotted to compare the energy consumption of existing and proposed techniques. By varying the time-interval, energy consumption of the proposed method along with prevailing techniques are plotted and compared. It is evident that the proposed method consumes less energy when compared with C-MOSDEN.

Considering memory consumption results obtained in Table.5, Figure.4 is plotted to compare the memory consumption of existing and proposed techniques. By varying the time-interval between 1 to 30 minutes, memory consumption of the proposed as well as existing techniques are plotted.

In all the cases, the proposed method's memory, energy, along with CPU consumption is lower when analogized with the prevailing C-MOSDEN algorithm. It indicates that the context-aware set of functions or operations create few overheads centered on the outcomes with respect to CPU, memory, along with energy consumption. The additional overhead created by context-aware functionalities is not substantial and it is stated by examining the outcomes of memory consumption. The capabilities of GK-ANFIS technique are also tested in the real world. The memory, CPU, energy consumptions, along with network usage of the techniques is measured. The cost savings are pondered as the common measurements rather than plotting all the measures and Figure.5 exhibits the techniques' cost-saving graph. Table.6 represents the cost savings of proposed and existing techniques.

	CPU	Memory	Energy	Network Usage
Existing C-MOSDEN	20	10	38	51
Proposed GK-ANFIS	17	8	34	48

Table 6 Cost Savings

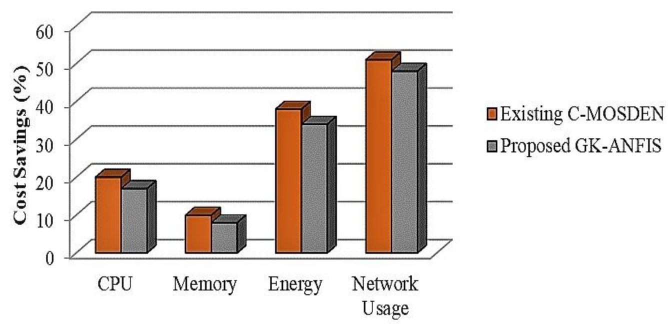

Figure 5 Cost Savings of the Proposed GK-ANFIS and Existing Methodology

Regarding CPU, energy, memory, along with network usage, Figure.5 exhibits that the GK-ANFIS's outcomes are superior to the prevailing C-MOSDEN's outcomes. The GK-ANFIS's cost-saving regarding CPU, memory, energy, and network usage is 17%, 8%, 34%, and 48%. Those values are lower when analogized with the prevailing C-MOSDEN technique. Based on the outcomes, the context-aware capabilities are competent in saving costs and it is expressed through the four parameters. Those parameters are CPU, memory, Energy and network usage. Selective sensing which could be attained by the proposed GK-ANFIS through data filtering reduces both energy and network usage. On account of wireless communication radios' less usage, the proposed method's energy consumption is reduced.

For evaluating three application scenarios like environmental monitoring, rehabilitation, and health and well-being, the experiments are executed. For executing this, every available sensor is configured and aimed at collecting the data. Using both with and without context-aware potentialities being activated, the experiment is performed. The metrics namely memory, CPU, storage, energy along with the network consumption are measured. Table. 7 represents values

RESEARCH ARTICLE

containing average CPU consumption and those values are plotted as indicated in Figure 6.

	Scenario 1	Scenario 2	Scenario 3
Existing C-MOSDEN WOC	47	47	47
Proposed GK-ANFIS WOC	46	45	44
Existing C-MOSDEN WC	41	39	37
Proposed GK-ANFIS WC	40	37	36
Existing C-MOSDEN CS	10	11	16
Proposed GK-ANFIS CS	9	8	14

Table 7 Average CPU Consumption

Table. 8 represents average memory consumption values and those values are plotted as indicated in Figure.7.

	Scenario 1	Scenario 2	Scenario 3
Existing C-MOSDEN WOC	110	110	110
Proposed GK-ANFIS WOC	104	103	105
Existing C-MOSDEN WC	108	107	109
Proposed GK-ANFIS WC	98	97	96
Existing C-MOSDEN CS	9	13	5
Proposed GK-ANFIS CS	7	11	4

Table 8 Average Memory Consumption

	Scenario 1	Scenario 2	Scenario 3
Existing C-MOSDEN WOC	52	48	46
Proposed GK-ANFIS WOC	48	46	44
Existing C-MOSDEN WC	45	43	42
Proposed GK-ANFIS WC	43	40	40
Existing C-MOSDEN CS	10	9	5
Proposed GK-ANFIS CS	9	8	4

Table 9 Average Energy Consumption

Table. 9 represents values containing average energy consumption and those values are plotted as indicated in Figure.8.

Table. 10 represents values containing total storage consumption of proposed and existing methods and those values are compared in Figure.9.

	Scenario 1	Scenario 2	Scenario 3
Existing C-MOSDEN WOC	2.5	2.4	2.8
Proposed GK-ANFIS WOC	2.2	2.1	2.5
Existing C-MOSDEN WC	2.1	1.7	1.9
Proposed GK-ANFIS WC	1.8	1.5	1.7
Existing C-MOSDEN CS	0.6	1	1
Proposed GK-ANFIS CS	0.4	0.8	0.7

Table 10 Total Storage Consumption

	Scenario 1	Scenario 2	Scenario 3
Existing C-MOSDEN WOC	2548	2000	2908
Proposed GK-ANFIS WOC	2375	1867	2765
Existing C-MOSDEN WC	2000	1088	1758
Proposed GK-ANFIS WC	1867	986	1574
Existing C-MOSDEN CS	783	1578	1500
Proposed GK-ANFIS CS	584	1374	1268

Table 11 Network Energy Consumption

Table. 11 represents values containing network consumption and those values are plotted as indicated in Figure.10. By creating the predefined data set that simulated the relevant user behaviors' including location changes and activity changes, in these scenarios, the techniques' implementation is done.

Considering three disparate scenarios, Figures 6, 7, 8, 9 and 10 exhibit the performance of proposed methods along with prevailing techniques' outcomes. The memory, energy, CPU, total storage and network usage consumption of the GK-ANFIS along with prevailing techniques is plotted. In all three scenarios, the GK-ANFIS attains better outcomes when analogized with the prevailing technique. The network communication, CPU, memory, total storage, along with

RESEARCH ARTICLE

energy consumption are optimal in case of utilizing GK-ANFIS technique. Context-aware potentialities could reduce costs at disparate levels based on the sensing objectives, scenarios, conditions, along with characteristics of outcomes. Depending upon the outcomes, it is inferred that, for saving the costs, any sort of context-aware functionality which would lessen the uninterested data collection along with transmission could be helpful.

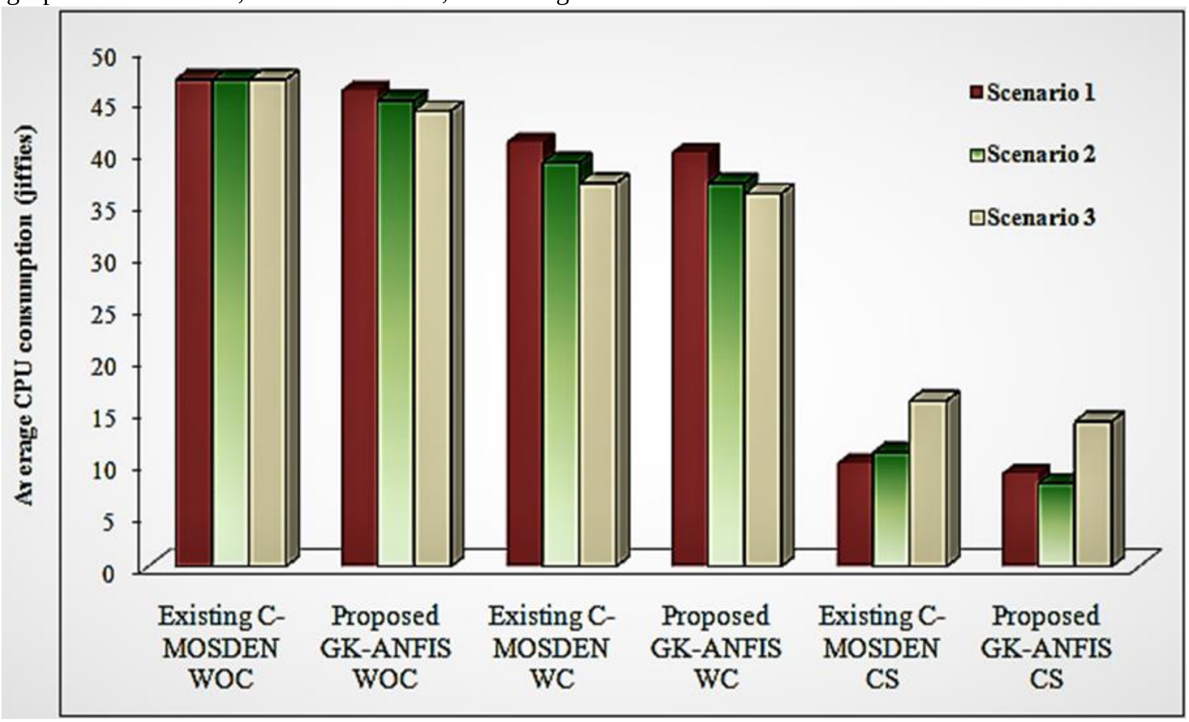

Figure 6 Average CPU Consumption for GK-ANFIS and Existing Methodology

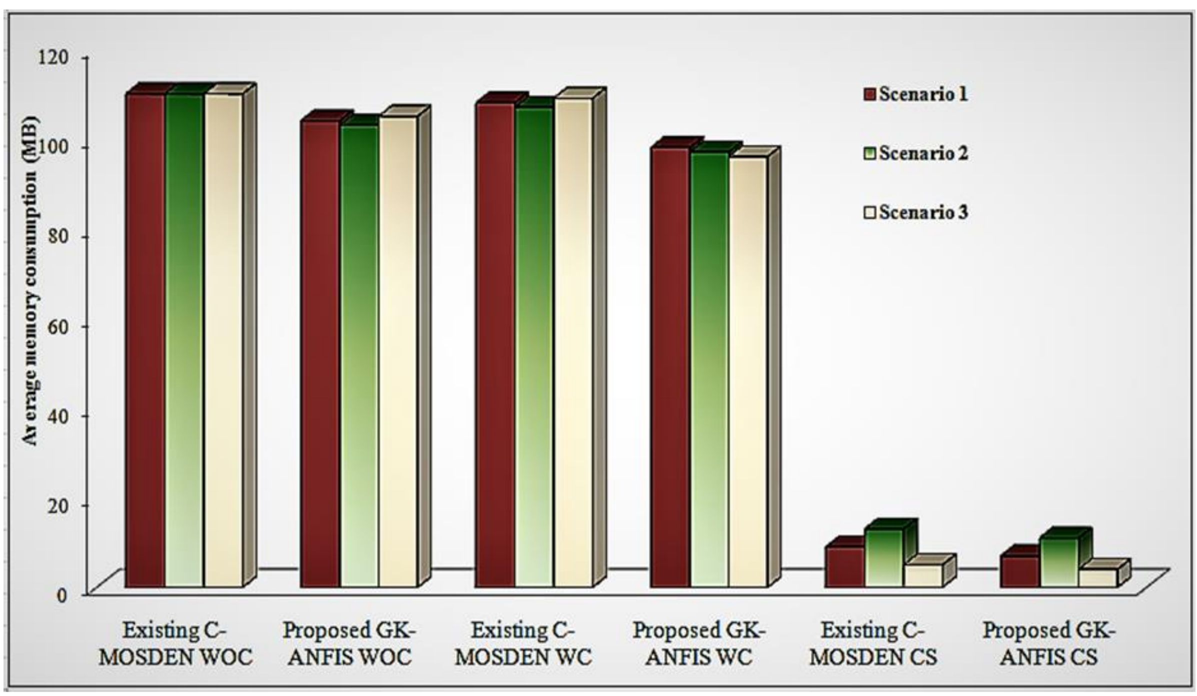

Figure 7 Average Memory Consumption for GK-ANFIS and Existing C-MOSDEN

RESEARCH ARTICLE

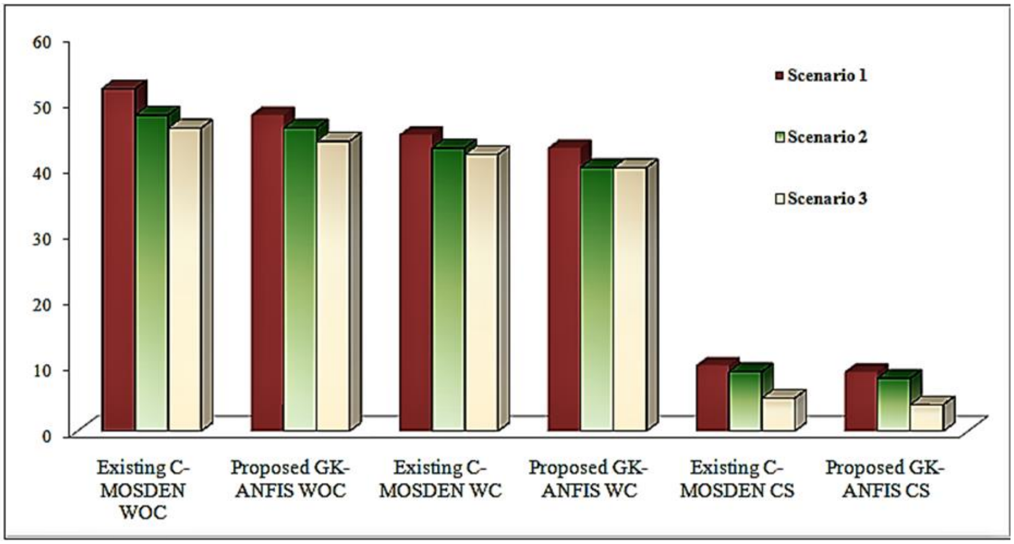

Figure 8 Average Energy Consumption of GK-ANFIS and Existing Methodology

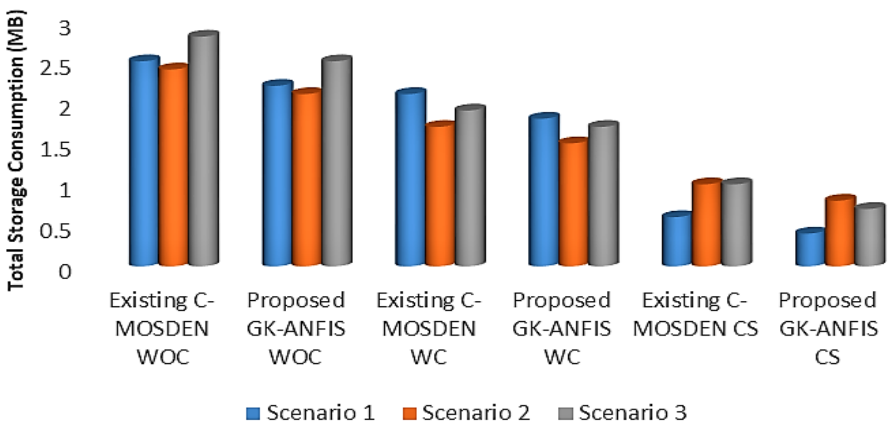

Figure 9 Total Storage Consumption for GK-ANFIS and Existing Methodology

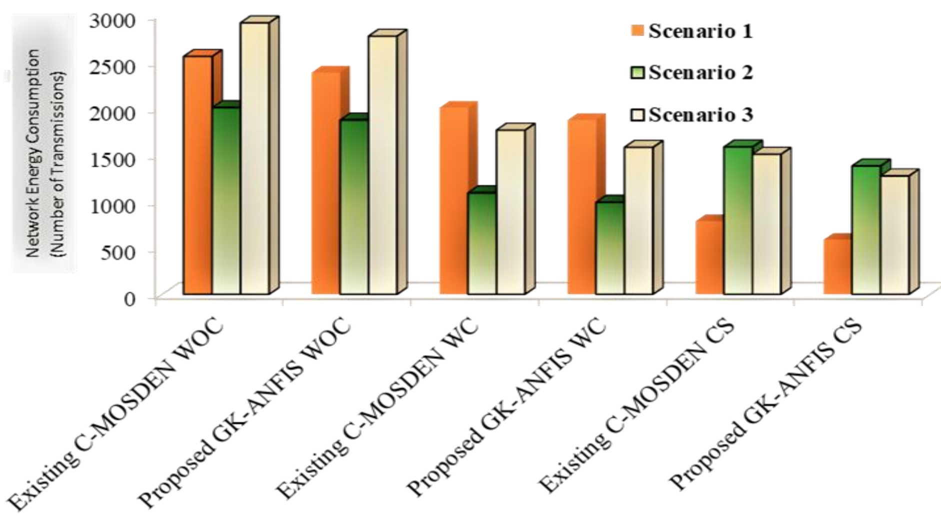

Figure 10 Network Energy Consumption of GK-ANFIS and C-MOSDEN

RESEARCH ARTICLE

5. CONCLUSION

An energy and activity aware data scheduling and data analytics scheme termed GK-ANFIS aims to assist IoT-HC architecture in the CC paradigm. GK-ANFIS executes dynamically balanced HC data distribution between the IoT and cloud devices. The proposed platform gathers just the data which are relevant, thus decrements the data storage and also processing time. A comparative research is performed against existent technique to examine the proposed work utilizing diverse simulation experiments. The techniques are executed utilizing three diverse real-world use-case scenarios. The metrics like memory consumption, energy consumption, CPU consumption, storage consumption, network communication consumption of the proposed and existent techniques is plotted and then the methodology's performance is assessed. Analogizing with the prevalent C-MOSDEN, the proposed GK-ANFIS attains the least cost savings regarding the metrics like storage, network, energy, memory, and CPU consumption in both the use-case as well as real-time scenarios. The technique effectively decreases the storage requirements, network communication requirements, and EC via the data filtration of IoT-data at certain time-intervals. The methodology can be utilized in the upcoming future and it is aimed at enriching the GK-ANFIS by utilizing the privacy-preserving data analytics abilities.

REFERENCES

[1] Sanjeevi Pandiyan, T. Samraj Lawrence, V. Sathiyamoorthi, Manikandan Ramasamy, Qian Xia, and Ya Guo, "A performance-aware dynamic scheduling algorithm for cloud-based IoT applications", Computer Communications, vol. 160, no. 1, pp. 512-520, 2020.

[2] Charith Perera, Dumidu S. Talagala, Chi Harold Liu, and Julio C. Estrella, "Energy-efficient location and activity-aware on-demand mobile distributed sensing platform for sensing as a service in IoT clouds", IEEE Transactions on Computational Social Systems, vol. 2, no. 4, pp. 171-181, 2015.

[3] Mahalakshmi, J., and Venkata Krishna P, "An efficient priority based resource management framework for IoT enabled applications in the cloud", Evolutionary Intelligence, vol. 14, no. 2, pp. 863-869, 2021.

[4] Kalaivanan Karunanithy, and Bhanumathi Velusamy, "Cluster-Tree based Energy Efficient Data Gathering Protocol for Industrial Automation using WSNs and IoT", Journal of Industrial Information Integration, vol. 19, no. 1, pp. 100156, 2020.

[5] Mahammad Shareef Mekala, and Perumal Viswanathan, "Energy-efficient virtual machine selection based on resource ranking and utilization factor approach in cloud computing for IoT", Computers & Electrical Engineering, vol. 73, no. 1, pp. 227-244, 2019.

[6] Tahereh Saheb, and Leila Izadi, "Paradigm of IoT big data analytics in the healthcare industry: A review of scientific literature and mapping of research trends", Telematics and Informatics, vol. 41, no. 1, pp. 70-85, 2019.

[7] Thar Baker, Muhammad Asim, Hissam Tawfik, Bandar Aldawsari, and Rajkumar Buyya, "An energy-aware service composition algorithm for multiple cloud-based IoT applications", Journal of Network and Computer Applications, vol. 89, no. 1, pp. 96-108, 2017.

[8] Tahani Aladwani, "Scheduling IoT Healthcare Tasks in Fog Computing Based on their Importance", Procedia Computer Science, vol. 163, no. 1, pp. 560-569, 2019.

[9] Leila Ismail, and Huned Materwala, "Energy-aware vm placement and task scheduling in cloud-iot computing: Classification and performance evaluation", IEEE Internet of Things Journal, vol. 5, no. 6, pp. 5166-5176, 2018.

[10] Huaiying Sun, Huiqun Yu, Guisheng Fan, and Liqiong Chen, "Energy and time efficient task offloading and resource allocation on the generic IoT-fog-cloud architecture", Peer-to-Peer Networking and Applications, vol. 13, no. 2, pp. 548-563, 2020.

[11] Ding Ding, Xiaocong Fan, Yihuan Zhao, Kaixuan Kang, Qian Yin, and Jing Zeng, "Q-learning based dynamic task scheduling for energy-efficient cloud computing", Future Generation Computer Systems, vol. 108, no. 1, pp. 361-371, 2020.

[12] Zahra Ghanbari, Nima Jafari Navimipour, Mehdi Hosseinzadeh, and Aso Darwesh, "Resource allocation mechanisms and approaches on the Internet of Things", Cluster Computing, vol. 22, no. 4, pp. 1253-1282, 2019.

[13] Praveenchandar, J., Tamilarasi, A. Dynamic resource allocation with optimized task scheduling and improved power management in cloud computing. Journal of Ambient Intelligence and Humanised Computing, vol. 12, no. 3, pp. 4147–4159, 2021.

[14] Sampa Sahoo, Bibhudatta Sahoo, and Ashok Kumar Turuk, "An Energy-Efficient Scheduling Framework for Cloud Using Learning Automata", In 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE, pp. 1-5, 2018.

[15] Mohammed Joda Usman, Abdul Samad Ismail, Abdulsalam Yau Gital, Ahmed Aliyu, and Tahir Abubakar, "Energy-Efficient Resource Allocation Technique Using Flower Pollination Algorithm for Cloud Datacenters", In International Conference of Reliable Information and Communication Technology, Springer, pp. 15-29, 2018.

[16] Amudha, S., and M. Murali, "Deep learning based energy efficient novel scheduling algorithms for body-fog-cloud in smart hospital", Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 7, pp. 7441-7460, 2020.

[17] Xiang Wu, Huanhuan Wang, Dashun Wei, and Minyu Shi, "ANFIS with natural language processing and gray relational analysis based cloud computing framework for real time energy efficient resource allocation", Computer Communications, vol. 150, no. 1, pp. 122-130, 2020.

[18] Hamid Reza Boveiri, Raouf Khayami, Mohamed Elhoseny, and M. Gunasekaran, "An efficient Swarm-Intelligence approach for task scheduling in cloud-based internet of things applications", Journal of Ambient Intelligence and Humanized Computing, vol. 10, no. 9, pp. 3469-3479, 2019.

[19] Husnu Narman S, Md Shohrab Hossain, Mohammed Atiquzzaman, and Haiying Shen,"Scheduling internet of things applications in cloud computing", Annals of Telecommunications, vol. 72, no. 1-2, pp. 79-93, 2017.

[20] Randa Abdelmoneem M, Abderrahim Benslimane, and Eman Shaaban, "Mobility-Aware Task Scheduling in Cloud-Fog IoT-Based Healthcare Architectures", Computer Networks, Vol. 179, no. 1, pp. 107348, 2020.

[21] Abishi Chowdhury, and Shital Raut, "Scheduling Correlated IoT Application Requests Within IoT Eco-System: An Incremental Cloud Oriented Approach", Wireless Personal Communications, vol. 108, no. 2, pp. 1275-1310, 2019.

[22] Weiwei Lin, Gaofeng Peng, Xinran Bian, Siyao Xu, Victor Chang, and Yin Li, "Scheduling algorithms for heterogeneous cloud environment: main resource load balancing algorithm and time balancing algorithm", Journal of Grid Computing, vol. 17, no. 4, pp. 699-726, 2019.

[23] Xiaojin Ma, Honghao Gao, Huahu Xu, and Minjie Bian, "An IoT-based task scheduling optimization scheme considering the deadline and cost-aware scientific workflow for cloud computing", EURASIP Journal on Wireless Communications and Networking, no. 1, pp. 249, 2019, https://doi.org/10.1186/s13638-019-1557-3.

[24] Yiping Wen, Zhibin Wang, Yu Zhang, Jianxun Liu, Buqing Cao, and Jinjun Chen, "Energy and cost aware scheduling with batch processing for instance-intensive IoT workflows in clouds", Future Generation Computer Systems, vol. 101, no. 1, pp. 39-50, 2019.

RESEARCH ARTICLE

[25] Mukhtar Mahmoud, ME, Joel JPC Rodrigues, Kashif Saleem, Jalal Al-Muhtadi, Neeraj Kumar, and Valery Korotaev, "Towards energy-aware fog-enabled cloud of things for healthcare", Computers & Electrical Engineering, vol. 67, pp. 58-69, 2018.
[26] Chunlin Li, Chengyi Wang, and Youlong Luo, "An efficient scheduling optimization strategy for improving consistency maintenance in edge cloud environment", The Journal of Supercomputing, vol. 76, no. 9, pp. 6941-6968, 2020.
[27] Banos, Oresti, Rafael Garcia, Juan A. Holgado-Terriza, Miguel Damas, Hector Pomares, Ignacio Rojas, Alejandro Saez, and Claudia Villalonga. "mHealthDroid: a novel framework for agile development of mobile health applications." In International workshop on ambient assisted living, Springer, pp. 91-98, 2014.

Authors

Sampath Kumar Y R received his M.Tech Degree from the Department of Computer Science and Engineering at BNMIT, VTU, Belagavi, Karnataka, India. He is a research scholar at the University Visvesvaraya College of Engineering, Bengaluru, India. He is pursuing his Ph.D degree in Computer Science and Engineering from Bangalore University. His current research interest includes Internet of Things and Cloud Computing.

Dr. Champa H N received her Ph.D degree from the Department of Computer Science at University of Mysore, Karnataka, India. She is currently a professor in the Department of Computer Science and Engineering at University Visvesvaraya College of Engineering, Bengaluru, India. Her research interest includes Data mining, Image Processing and Machine Learning.

How to cite this article:

Sampath Kumar Y R, Champa H N, "An Energy Aware Data Scheduling Approach in Cloud Using GK-ANFIS", International Journal of Computer Networks and Applications (IJCNA), 8(5), PP: 490-506, 2021, DOI: 10.22247/ijcna/2021/209983.