EAI Endorsed Transactions
on Scalable Information Systems Research Article **EAI.EU**

Efficient load balancing Adaptive BNBKnapsack Algorithm for Edge computing to improve performance of network

Malti Nagle[1,*] and Prakash Kumar[1]

[1] Department of Computer Science and Engineering, Jaypee Institute of Technology, Noida, U.P., India

Abstract

INTRODUCTION: In present days, Automation of everything has become essential. Internet of things (IoT) play an important role among all medical advances of IT. In this paper, feasible solutions are discussed to compare and design better healthcare systems. A thorough investigation and survey of suitable approaches were done to select IoT based systems in hospitals consisting of various high precision sensors.
OBJECTIVES: The challenge healthcare system face is to manage the real time patient's data with high accuracy. Second challenge is at fog devices level to manage the load distribution to all sensors with limited availability of bandwidth.
METHODS: This paper summarizes the selection criterions of suitable load balancing algorithms to reduce energy consumption and computational cost of fog devices and increase the network usage that are supposed to be used in IoT based healthcare systems. According to the survey BNBKnapack algorithm has been selected as best suitable approach to analyze the overall performance of fog devices and results are also verify the same.
RESULTS: Comparative analysis of Overall performance of fog devices has been proposed with using SJF algorithm and Adaptive BNBKnapsack algorithm. It has been observed by analysing system performance, which is found as best among other load balancing algorithm Adaptive BNBKnapsack is successfully reduce the energy consumption by (99.29%), computational cost by (98.34%) and increase the network usage by (99.95%) of system
CONCLUSION: It has been observed by analysing system performance, Adaptive BNBKnapsack Load balancing is successfully able to reduce the computational cost and energy consumption also increase the network usage of the fog network. The performance of the system is found best among other load balancing algorithm.

Keywords: IoT, Cloud Computing, Context aware system, scheduling, load balancing algorithm, EEG sensor, BLE, CloudSim, iFogSim, Stress related health issues, Adaptive BNBKnapsackAlgorithm.Introduction.

Received on 18 June 2023, accepted on 20 August 2023, published on 20 September 2023

doi: 10.4108/ eetsis.3924

1. Introduction

In recent times, world is moving towards automation of everything to overcome the downsides of traditional models. Automation of everything escalating the role of IoT enabled systems, in which network devices integrated with wireless sensors, are being well-liked globally at its extreme. IoT enabled devices collects data from numerous applications, like – advance healthcare system, vehicle system, devices applied for automation of smart city projects and other computing. Along with collection of data, IoT aided infrastructures need decision making and processing of data in real-time.

*Corresponding author. Email: nagle.malti083@gmail.com

There in integration, data require to send over cloud, and processed continuously. As a matter of fact, a massive amount of data collected by end devices and sensors transmitted at cloud to process, it escalate delay, increase energy consumption, and bottlenecks. IoT enabled network transmits data from sensors to cloud by forwarding data to upper level sensors, routers, and gateways, it may escalate power consumption on network. Sensitive cases such as healthcare systems constantly generates heterogeneous data thus, processing of such massive IoT data releases enormous amount of power, and also has high number of delay. Delay in store and data process escalate failure in alarm generation for critical medical issues and patients can critically ill or die [1][2].

Internet of Thing (IoT) is a combination of inter connected network embedded with hardware, sensors, actuators, and communication devices through which devices collect and exchange the data. In the past few years, appearance of IoT considered at certain level of cynicism [3]. IoT envisions the interaction of devices and human being in remarkable way to enhance the standard of living worldwide. It is a revolutionary stroke to transform the wireless network into the communication network where machines can interact with each other via sensors. This is the reason sensors are being installed around the world in massive amount. Therefore, enormous data processed over the cloud. Data forwarded to cloud consistently escalate congestion over the network. Constant confronted of wide range data at cloud is another point of discussion [4][5][6].

Resource management and transmission of real–time data create hurdles for IoT developers. To tackle this challenge, resource allocation policy should skilfully implemented at IoT enabled networks. Another novel contribution are LPWANs in expansion of IoT enabled systems to its apex [7][8][9]. Forthcoming licensed – band standards, named as NB-IoT are primus competitor of unlicensed – band standards, notably fit for healthcare systems [12]. IoT empowers wide range of industries automation, in addition to allow collection of massive data to process and store. Internet of things as a technology fascinating commercial market by its utilities and hailed as the driver of the Fourth industrial revolution [10[11][13]. IoT addressed to develop applications for countless commercial areas like – agriculture, smart city development, health monitoring, and traffic control managements. IoT systems are feasible but be short of dealt with Big Data. IoT has to send data all time over cloud to process, it can cause congestion over network, and along with it resource allocation is also perplexing.

Edge computing brings innovative environment where devices at bottom layer (end sensor's), transmit data to it's an above layer, it sends data to the cloud only if required. This novel approach decreases network congestion, and it helps to reduce the power consumption. Fog computing (FC) paradigm has been proposed to easily manage Big Data, and along with its needed effects enhance the network performance. Cloud services now extend to the edge of the network in Fog Computing (FC). Several challenges addressed by cloud network has been overcome with innovative platform that empowers the performance of resource management policies upon IoT and FC. Distributed computing introduces with new features that enhance cloud services from cloud end to edge of network. There in integration, its infrastructure allows all-in-one leverage of edge and cloud resources [14][15]. It enables management among both cloud and end devices with services like networking, computation, storage services. FC indeed involves all components of any applications running as one in the cloud together with end devices called sensors and actuators or between cloud and gateways/ routers. FC endorses network mobility, resource heterogeneity, interplay with cloud, and distributed data analytics to accomplish essential requirements of application low latency with a wide and dense geographical distribution [2][6][17]. Some of key benefits of FC are its usefulness at both cloud computing and edge, imminent impendency of end devices to edge devices. Cloud resources available with on-demand scalability. There are enormous features allied with FC to assure its success.

Fog computing is network of large nodes but it consumes low energy compared to centralized CC (cloud computing) systems. Architecture of fog computing connects end sensors nodes, edge devices, and cloud in hierarchical arrangement. Layout of fog computing consist sensor nodes placed at geographical location at bottom of the architecture and send collected data to the up-level by gateway/routers [5].

Fog computing is an extension of CC (Cloud computing) which deploys data for storage, computing, resource management, communication from cloud to the endpoint IoT devices. FC architecture is inter connection of physical devices with logical network of useful IoT devices, network elements, software and hardware. Architecture involves connection of geographically located devices and physical network. It is basically hierarchal representation of network where top level is cloud and till lower layer devices are – gateways, protocols, connection pattern, capacities of node links, and sensors/ actuators. In Figure 1 – architecture of fog computing is explained, where it is clearly defined how the connection is established between devices and cloud. Sensors sends data to its upper layer's fog nodes through communication technology. These fog nodes collect data and send to fog service centre to process and analyse, whenever required this fog data send to cloud. End sensors can be – traffic control sensors implemented in car, cctv camera sensors for traffic control, mobile connected sensors, healthcare related sensors etc.

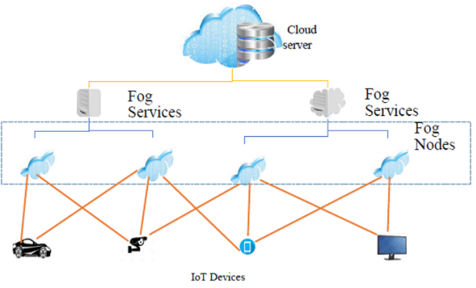

Figure.1 Architecture of Fog Computing

Fog computing architecture has layers of services to expostulate in Figure 2. In architecture, bottom most layer has fog devices located at the geographic level. Sensors at ground level sense data from environment and send to an upper layer (fog device) via gateways/routers to process and analysis. Another device is actuator at bottom layer, which is responsible to alert for changes in surroundings and control all functions at end nodes [18][19][20]. Data collected by end sensors send to fog devices to process. Data collected by sensors further save into IoT data stream in sequence of irreversible values. Fog device is capable to control over the hosting of any application based modules. Gateways are mediator, which connects sensors to the upper layer fog devices over internet. It provides on- demand cloud resources from remotely located data centers.

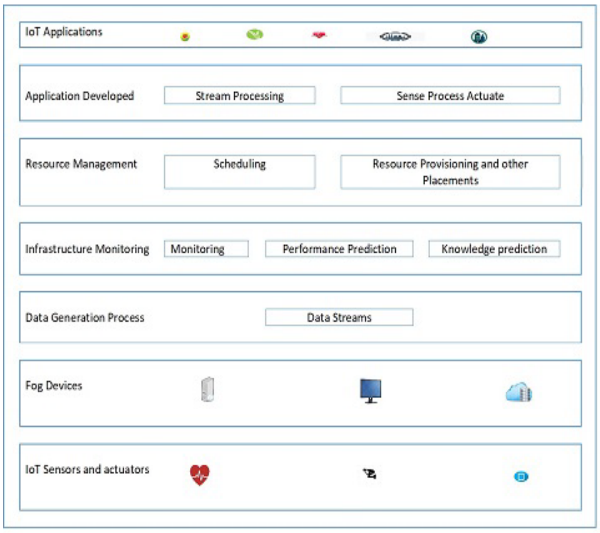

Figure. 2 Fog layer architecture

There are two application based models: data stream processing model and sense-process-model. Fog uses Distributed Data Flow model to deploy the developed applications for simulation. Model of application is set of modules, which comprises processed component of data. Sensors generates data in the form of information grouped together as data stream, which act on behalf of running application and commands generated as results further sent to actuators. The stream processing model process data streams constantly generated from sensors.
Fog computing basically bring into play for minimization of resource allocation and energy consumption. It uses various optimization techniques, load balancing algorithms on fog devices to reduce the above mentioned constraints.
It has been noticed throughout literature survey that, various optimization algorithms implemented to enhance the performance of network addressing parameters, those are essential for fog computing, minimize energy consumption, computational cost, maximize resource utilization, and overall cost of network. Fog network is hierarchal arrangement of devices, nodes and cloud. To utilize the network's full bandwidth and maximize the resource allocation is biggest challenge for any researcher. Resource allocation in fog computing is similar to NP problem in tree structure. Optimization algorithms those are enacts significant role to allocate resources in an appropriate way are knapsack algorithms [1][21][22], min-max algorithm, and GA algorithm. Some of load balancing and scheduling algorithms are also beneficial for resource allocation in fog computing. In this paper, implementation of Adaptive branch and bound knapsack algorithm is illustrated to boost up the performance of fog network. As it was discussed earlier, resource allocation in fog computing is the toughest task, because it has hierarchal arrangement of nodes. Assignment of resource in appropriate manner for resource utilization is NP problem to a certain extent. Optimization techniques being applied for NP problem may perhaps refer, to find suitable algorithm for resource utilization in fog networks. In this paper, core contributions are as follows:

- Algorithm design to solve NP hard problem included in structure of fog devices.
- Purpose of algorithm is to minimize the overall computational cost, and energy consumption of fog architecture.
- System should have implemented in a way that can maximize the network usage.

2. Literature Review

In ultramodern world, wireless network uplifted with innovative features like smart devices, automated machine, and IoT enabled interconnected networks. Researchers encouraged to develop novel techniques to liberate the world

from geographical boundaries to connect devices and communicate with each other, and to send data at remote location. Many researchers designed frameworks to incorporate wireless network with IoT gadgets. Frameworks directly transfer data to cloud to store, process and analyse. It becomes difficult to control over the congestion and latency delay in such frameworks. As in earlier section it has been illustrated that IoT enabled devices are rapidly increasing day by day. It is emitting huge amount of energy in environment, thus researcher encounters difficulty in allocation of VM to IoT devices. Load balancing, latency delay, energy consumption are the major elements to figure out while VMs are allocated to Fog devices [23][24][25].
By and large IoT can be considered as an automation tool for events happens in day-to-day life. Recently IoT has been discovered as powerful tool for healthcare industry. Various scheduling algorithms have been developed at cloud platform to analyse performance of cloud and connected nodes. Author [30] discussed monitoring tools applied to evaluate cloud performance in perspective of both ends users (Providers and cloud users). There are some basic parameters considered by author [26][27][31] to measure performance of cloud – response time, storage usage, CPU utilization at its peak hours. Five dominant cloud providers accumulated to host their websites and evaluate the response time and latency of each and above cloud. [28][29][32] Suggested innovative approach for virtual environment at infrastructure layer by monitoring its response time. Some researcher draw attention to monitor cloud performance based on parameters such as - latency, execution time, and response time to underline its effectiveness at cloud platform. [34] Emphasized on performance factors – response time, latency monitored by automation tool to update current status of cloud to it's all users.

BDCaM Model of also novel framework proposed [1]. This framework is develop to perform analysis and process of data on centralized cloud. The data contains information of various diseases along with the patient medical history from different sources. IOT devices, wearable sensors and many other devices. All collected data further segregated into cluster and stored on cloud. There some papers referred to know more about the healthcare system with IoT.

Study of medical data- it is Framework that monitor and manage IoT device's data. It collects Patient's data via sensors and forward to cloud server. Communication technology used to transfer data is Bluetooth. Physicians diagnose and monitor health status of patients [31].

Paper presented survey report that discuss Nylon coated ECG sensors' fabrication, characteristics and validation. It collects live ECG data and measure electrical properties by electrical probing. Signal reliability is measured and surface characterization is determined through SEM. Fourier transform infrared spectroscopy is also analysed [37].

Model is based theoretical aspects to support, preventive and personalized healthcare system. Various IoT enabled healthcare system highlighted challenges arise during real time smart healthcare [36].

Different issues arises are defined for IoT enabled healthcare system. Structured classes are defined classification of delivery system for healthcare, system types [39].

This paper proposed framework for HMS based on context of medical data. New middleware is placed named as context awareness middleware helps to overcome data management related problem.

Personalized web-based tool to analyse epilepsy. Combination of GA, wavelet transform and SVM are used in proposed system. Effective of system is verified for two EEG data (Long term and Short term). Results shows 90% accuracy and training time accelerate 4.66 times and prediction time is also in real time [35].

Healthcare system with some advance monitoring system consist IoT devices and process and analysis of data carried out at cloud [34].
Fog network for DCNS network and implemented algorithm to reduce latency delay, and energy consumption. Author implemented algorithm to control over the load balancing for fog devices. This algorithm superbly manage load balancing. Scheduling algorithm proposed by author is efficiently allocate VMs to the fog devices [6].
Applied knapsackSOS algorithm to allocate VMs to the Fog devices. The purpose of the algorithm is to minimize computational cost of overall network. The author assumed allocation of VM as metaheuristic NP hard problem. So author proposed the knapsackSOS algorithm because knapsack algorithm found it as best solution of dealt with NP hard problems. The algorithm implemented by author selected MIPS of CPU as profit and bandwidth utilization as a weight required in knapsack algorithm as key parameters. Author achieved best solution for VM allocation in available bandwidth. The overall computational cost is successfully reduced [34].

3. Problem Statement

Healthcare is the most important matter of concern worldwide, majority of population have been influenced with toxic waste emitted by industries. Person of any age group is facing different health issues. Many novel diseases detected by health organizations. To spread awareness around the globe healthcare system moving towards the modern techniques, where traditional healthcare models replaces with modern healthcare system that connected with wireless networks over worldwide. So, the clinical profile of any patient is available over the internet. Using the modern techniques any patient or doctor can access their medical profile from remote locations. The storage of healthcare data is very complicated. One reason, no error is allowed to process and storage of such data. Secondly, it is heterogeneous in nature and very large in size. To process such essential data, network must have to be accurate and fast. In recent days many smart healthcare devices introduced in the market. These devices send data at the central data storage (cloud). Data sent over the cloud further processed and revert the result back to devices. But, puzzle is to manage the network load. It increasing the network congestions, latency delay during data transmission and decreases the overall network performance. The continuous flooding of vast healthcare data at cloud increases the chances of network crashes. This is not accepted for healthcare system, as it can lead the patient to the critical illness or death. Some parameters that affect system performance are:

- **Computational cost** – Healthcare system heterogeneous data is required to analyse and Process every second. Minimization of computational cost due to the complex analysis of health records in cloud computing enabled advance healthcare systems in a real-time challenge.

- **Energy consumption** – The healthcare system generates data in petabytes or more. Cloud requires continuous processing of such critical data in real-time. Also, the number of IoT enabled healthcare devices increasing rapidly. Energy consumption by such devices during data transmission and processing is another matter of serious discussion.

- **Emergency alarm generation** – In any serious medical condition, it is required to generate an emergency alarm to alert the family or doctor of patients' to provide immediate treatment to the patient. There must be an efficient system to decide which critical disease should be kept in priority to generate an alarm for a patient according to their medical history. False alarm ratio may increase the failure of the healthcare system, as it may not be capable to provide emergency medical help to the patients.

- **Access response time** – The access response time of the system should be minimum so that the system can effectively provide the medical information to the patient as well as a doctor.

- **Throughput** – Healthcare data generated in 7 V's of the big data. It needs complex processing that may increase the throughput of the system.

4. Motivation

The fog computing is essential for upcoming technologies. Traditional healthcare system was completely rely on manual observations. New era for modern healthcare system is more based on sensor based data to remotely monitor the health of patients. The modern technology seeks for efficient fog computing to manage and store data. Motivation behind the current research is to find feasible solution to provide efficient load balancing. The research was conducted to find load balancing for fog devices to manage and forward healthcare sensors data over cloud. Key challenges to accommodate maximum sensors in limited bandwidth. The research also focusing to find effective load balancing algorithm to provide feasible solutions.

5. Proposed Work - Optimization techniques

The first step for developing an effective healthcare system is, to design network using elite methodology that can reduce the network load and also send more data at cloud received from sensors within limited bandwidth. This can be a metaheuristic problem. Sometime such problems can be easily handled using algorithm available for NP hard problems. During literature survey it has been studied that many algorithm like Greedy Knapsack and Knapsack with GA applied on the network designed for IoT devices was somehow effective. All algorithm were designed and simulated on iFogSim simulation tool. iFogSim is found as favourable tool to simulate the result for algorithm described above. Minimize the computational cost and energy consumption is the essentials of any IoT enables network, are also focus of current study and point out in problem statement section-2. During study it has been noticed that, to develop well-organized IoT enable system, it is necessary to understand those elements affects network the most. Knapsack approach applied in paper [6] opted the weight as bandwidth and profit as MIPS of the CPU. Here in proposed work, edge of the network where sensors and actuators are connected and root of network where cloud is balancing the load are considered as key variable to reduce the energy consumption, and computational cost of the network in conjunction with upswing the network usage. GKS or the Knapsack with GA both only applied on the edges where sensors are connected and sends data to the cloud. Load balancing by such algorithm only depends on the bandwidth available at edges and minimize them using bandwidth utilization. But in advanced algorithm load balancing at both (root as well as nodes) and job scheduling techniques both applied on network to minimize the energy consumption and maximize the network usage.

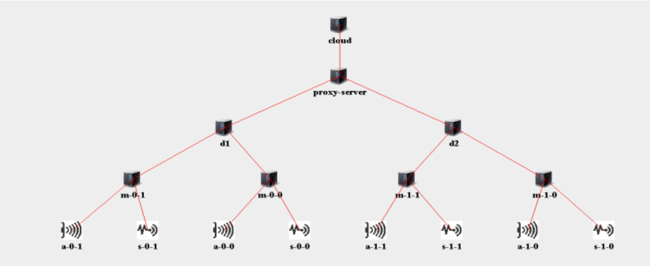

Figure.3 Topology for fog devices on iFogSim

In Figure 3 the network of fog devices, server, gateways and cloud are arranged in hierarchical structure. Architecture has sensors and actuators on leaf node and on the top of sensors, there are a device which receives data from sensors and process at FD or sometimes it sends directly to the cloud. The FD and its' application properties are explained as follows: Task of Fog Device (FD) is to manage data received from sensors using different methods like filter the data, analyse, and stores it at FD. Main properties of FD are: RAM, uplink BW, downlink BW, MIPS (Million instruction per second), rate/MIPS, topology level, power required in busy and ideal mode. Hosts (host1; host2;…..hostn) are assigned to each FD. All host have properties – RAM size, BW size, storage capacity, and number of processing elements for each host. Allocation policy to assign PEs for FD is responsibility of application module. After allocation and execution of PEs to all host, it update MIPS values for each FD. Further total MIPS for all PEs are calculated and updated at FD. Many algorithms are alter timely to manage the FD's overall performance, to reduce the latency delay and network usage; thus, all of them applied to succeed fast execution and best results [37].

5.1 Phases of experimental setup

Fog devices are connected with sensors and actuators. Sensors sends data directly to FD. FD forwards it to cloud for process, storage and management. This escalate the challenges to assign bandwidth to all FD and minimize the network usage because bandwidth is limited for network. Researchers tried to solve this problem using load balancing algorithms (like weight knapsack problem, GA algorithm and other algorithms). In, This paper adaptive BNBKnapsack Algorithm is applied for load balancing. In this section, the experimental setup is described along with adaptive BNBKnapsack Algorithm FCFS and SJF Algorithm. iFogSim simulation environment has been used for outcomes of computational cost, network usage, energy consumption. This section is incorporated Fog Devices setup – illustrated in phases.

5.1.1 Fog Devices initialization

First phase represents the fog device initialization parameter for each device connected in fog network from cloud to fog devices. Table 2 comprised significance of parameters (MIPS rate, size of RAM, bandwidth for uplink and downlink, along with level where devices are installed in fog network, energy consumption in busy and idle state) for each fog layer.

5.1.2 Host Allocation

Phase 2 contains information of host allocation rationalisation. Proposed work entails host allocation with framework –

Operating system - Linux OS
Virtual Memory model - Xen
Bandwidth – 10000 bps
Storage – 1000 MB

Setup mentioned in Table (1) has been established to allocate host to each FD. Allocation of FD is challenging task when Fog network has to assign maximum number VM to FD in limited size of bandwidth. Thus it require efficient algorithm to assign VM to FD. Formation of Fog network in figure 4 represents that it is hierarchical structure and require upper bound and lower bound limit to allocate FDs, therefore it requires the algorithm which necessitates branch and bound alignment of for network. Adaptive BNBKnapsack algorithm is suitable approach to attain challenges of bandwidth allocation. Table represents the overall experimental setup that has been simulated on iFogSim to evaluate the efficiency of porposed BNBKnapSack Algorithm for load balancing.

Table. 1 Required parameters to send data from camera to cloud- MIPS rate, RAM size, Bandwidth of lower level to upper level, CPU power status at busy and idle condition.

Name	MIPS	RAM	Upbandwidth	Lowbandwidth	Level	RMIPS	Busy power	Idle Power
Cloud	44,800	40,000	100	10,000	0	0	1,648	1
Proxy server	2800	4000	10,000	10,000	1	0	107.339	83.4333
Area/department	2800	4000	10,000	10,000	1	0	107.339	83.4333
camera/mobile	500	1000	10,000	10,000	3	0	87.53	82.44

Table. 2 Host Allocation

Storage	BW	Architecture	OS	VM Model	Time one	Cost	Cost/Memory	Soct/Storage
1,000 MB	10,000 Bps	X86	Linux	Xen	10	3	0.05	0.01

Table. 3 Experimental setup for DCNS and EEG game controller – CPU tuple length, updated with modified tuples value required for EEG sensors and actuators.

Source Module	Destination Module	Periodicity (ms)	CPU Tuple length (B)	Tuple updated length (B)
EEG	Client	0	3000	500
Client	Concentration calculator	0	3500	500
Concentration calculator	Connector	100	1000	1000
Concentration calculator	Client	0	14	500
Connector	Client	100	28	1000
Client	Display	0	1000	500

5.2 Implementation of Adaptive BNBKnapsack

The Fog Device follow tree like structure. It contains vaious devices at various level. top most level which can be considered as root of tree is "cloud", at lower levels are other devices are assembles – "Data Center", " Fog Devices", "VM" and at the lowest level – "Sensors" and "actuators". This tree like structure received data from sensors and forward it to the cloud. The overall process requires to store and forward data with limited amount of bandwidth and also generates huge amount of energy. This increases the network usage and also it is difficult to handle the load balancing of network. The traditional approach of load balancing in Fog network is FCFS. But, the disadvantage of this load balancing methods is lack of minimization of the network usage also increases the energy of network.
In this paper, FCFS and other load balancing methods are applied on Fog Devices which consist the sensors like EEG, Diabetes sensor and pulse sensor.

FCFS (First Come First Serve) – It allocate the first sensors to the first VM and second sensor to second VM and this

process repeated until all sensors allocated to VM in series. For the device mention in figure there are some result on algorithm FCFS. VM Allocation perform in sequential manner. Drawback of this is, if first VM is require high amount of bandwidth and time is large to finish the allocation than it has to wait till allocation is finished for first VM and repeat this process till all sensors are allocated with VM. Limitation of bandwidth of network sometimes FCFS is unable to allocate all VMs.
The parameter discussed in “problem statement” is not completely resolved with FCFS [35].

SJF – The SJF is shortest job first algorithm in which it allocates first VM to shortest FD then proceed to FDs.

But in both the cases cloud allocates the VM scheduled in series. But, in IoT enabled system sensors are required to allocate by using metaheuristic techniques. Reason behind the scenario is structure of fog computing is hierarchical, so allocation of VM in fog computing is NP problem. Knapsack in proposed work allocate the VM by considering bandwidth and CPU MIPS rate as parameters of algorithm. But the IoT enabled network availability of sensors and allocate the VM on the basis of combinatorial optimization. It also needs to follow backtracking in case of node at higher level are higher than neighbours nodes at the same level.

Thus, **adaptive BNBknapsack** algorithm gives outshine results as compare to traditional knapsack algorithm. Improved version of knapsack algorithm is adaptive BNBknapsack algorithm which allocate the VM according to the combinatorial measurements of bandwidth along with CPU utilization for each node where sensors are connected and check every node till the upper level after reaching at root node this again sort in ascending order to allocate the shortest VM at priority. SoSKnapsack is define the parameters used to update fog devices. It also describe the allocation of profitofitems and weights for each VM to maximize the Bandwidth utilization.

5.3 Update fog Devices

The Fog Device is first get VM List. The VM list is elected from Adaptive Knapsack Algorithm mentioned in Algorithm 1 and Algorithm 2. The elected VM is further connected to Fog Devices. Selection of best VM is based on the Algorithm 2. The profitofitems in Algorithm 2 is the value of VM CPU utilization and mips rate. The value of profitofitems and weights are assigned in Section Update for Devices in iFogSim. For each VM new fog device is allocated.

Algorithm : represent the steps to update fog devices for each VM till the each node is allocated with VM.

Algorithm Adaptive BNBKnapsack

```
 1: Create fog broker and
  applications.
 2: Add sensors application to fog
 broker.
 3: for i= 0 to number of PEs do.
   4: create fog devices (sensors node
   as I, MIPS, ram, Upbandwidth,
   downbandwidth, busy power, idle
   power).
   5: end for
   6: end for
   7: edges connected to module.
   8: Initialize a module mapping.
   9: Submit applications.
  10: goto update fog application.
  11: calculate profitofitems using CPU
  utilization/bandwidth
  12: calculate weight
  CPUMIPSrateofModule
  13: calculate count of PEs
  14: call adaptiveKnapsackBNB.
  14: (if VM!=0) goto 1
  15: else
  16: allocate VM.
  17: goto 3.
  18: update energy consumption using
  parameters computational cost and
  energy consumption.
  19: if(i>numberofPes).
  20: End for.
  21: stop iFogSim.
  22: output of result evaluation.
```

5.4 update energy consumption

Energy consumption is calculated using total mips allocated, current energy consumption, current time, updated time of utilization and last utilization update time. The energy consumption is updated by every iteration of vm allocation. Every iteration pass the value of all above mentioned parameters and calculate the energy consumption for each host allocated with VM using adaptive BNBKnapsack Algorithm. The following expression is used to calculate the energy consumption.

NEC = CEC + (Timenow –LUUT).host.powermodel.power (LU)SEC(NEC/MIPS)

Where,
NEC = New energy consumption
CEC = Current energy consumption passed from Updated Allocated MIPS
LUUT = Last Updated Utilization Time
LU = Last Utilization

The complete process is based on allocation of utilize maximum bandwidth and allocate the VM.

6. Results and Discussion

The results represent the computational cost, network usage and energy consumption of Fog devices with number of processing element. Number of iteration for same algorithm along with number of processing element has been evaluated and this process has shown that the algorithm is successfully decreasing the computational cost and energy consumption of Fog network and also it is helpful to maximize the network usage of fog network. By different observation on different combination of processing elements and allocation of vm using different load balancing algorithm result has been prepared and tabulated. (Figure 4,5,6) presents the visual of computational cost, network usage and energy consumption of fog computing. It has been clearly observed that Adaptive BNBKnapsack algorithm efficiently reduce the computational cost and energy consumption and also able to improve network usage of overall network.

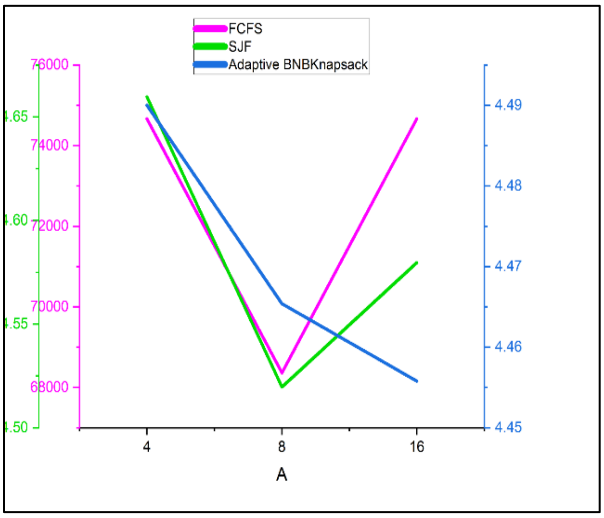

Figure. 4 Energy Consumption of Edge Network

As Figure 4 shows energy consumption is reduced by implementation of Adaptive BNBKnapsack in all conditions, wherein FCFS and SJF has high energy consumption. This can be helpful to apply green computing at cloud which is essential for efficient cloud computing.

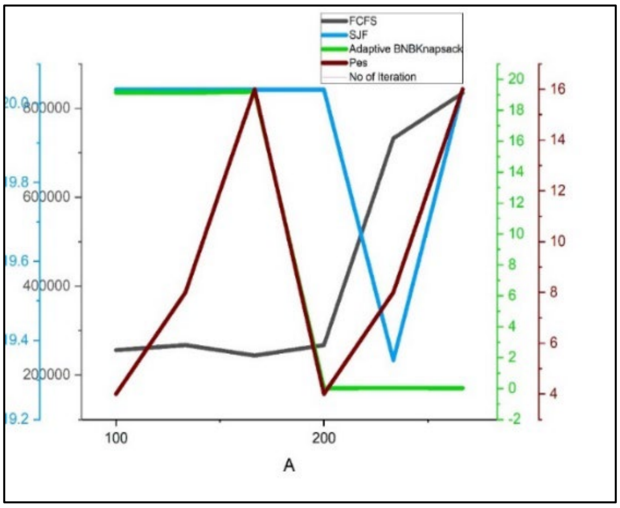

Figure. 5 Computational Cost of Edge Network

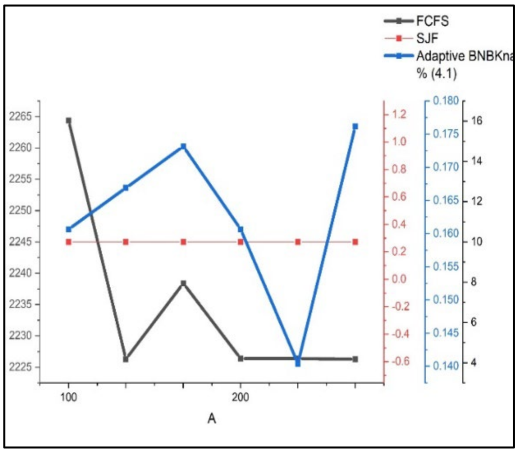

Figure. 6 Network Usage of Edge Network

In Figure 5 is visual representation of overall computation cost of Fog devices. The proposed method is capable to reduce the computational cost.

Figure 6 the proposed algorithm is successfully increase the network usage in comparison to other load balancing algorithms.

Table 4 shows the overall performance and impact of load balancing algorithms at different iterations and variation in number of Tasks (Pes).

Table. 4 Result of load balancing algorithm (FCFS, SJF, BNBKnapsack)

	FCFS	SJF	Adaptive BNBKnapsack	Pes	No of Iteration
Energy Consumption	74665.29822	4.659638151	4.49		
Computational Cost	243654.8384	20.03458365	19.19581865		
Network Usage	2264.4	0.27248	0.17616	4	100
Energy Consumption	68365.29822	4.519638151	4.4654		
Computational Cost	256124.8384	20.03458365	19.14541865		
Network Usage	2238.4	0.27248	0.17316	8	100
Energy Consumption	74665.29822	4.579638151	4.455798989		
Computational Cost	267124.8384	20.03458365	19.13858186		
Network Usage	2226.4	0.27248	0.17316	16	100
Energy Consumption	74665.29822	4.659638151	0.081847284		
Computational Cost	267124.8384	20.03458365	0.023215001		
Network Usage	2226.4	0.27248	0.16068	4	200
Energy Consumption	64265.29822	4.625621	0.145886924		
Computational Cost	732455	19.35	0.041075		
Network Usage	2226.4	0.27248	0.1404	8	200
Energy Consumption	74665.29822	4.659638151	0.060629368		
Computational Cost	834333	20.03458365	0.023655084		
Network Usage	2226.3	0.27248	0.16692	16	200

7. Conclusion

IoT enabled system is kind of NP problems. To optimize the IoT enabled system it is required to apply optimization technique like-knapsack 0/1 algorithm. Knapsack provide better optimized solution for NP problems. IoT is again has challenge to solve combinatorial NP problems, because various factor in combination effects the system. Experiment conducted to optimize by applying Branch and Bound with Knapsack along with scheduling at cloud using job scheduling algorithm to determine the dependency of network usage on the basis of metaheurisitic algorithm to schedule the Vm to different sensors. Results obtained revealed that among load balancing algorithm used selected on the basis of the nature of allocation of fog computing which is basically NP problem. Selection of scheduling algorithm Adaptive BNBKnapsack allocate Vm with NP hard problem. **Consequently, performance of cloud improved by 98.34% in comparison with simple FCFS scheduling algorithm significantly. Energy consumption is also decreased by 99.29% and computational cost by 99.29% and increase the network usage by (99.95%) of system.**

Acknowledgements.

I would like to express my sincere gratitude to the organization for their valuable support and assistance in completion this research work. I would like to thank of Dr. Prakash Kumar for their continuous encouragement and guidance in completion of research work.

References

[1] D. Rahbari, M. Nickray, Low-latency and energy-efficient scheduling in fog-based IoT applications, TJEECS, pp. 1406 – 1427, (2019).

[2] B. Baker, W. Xiang,I. Atinkson, Internet of Things for Smart Healthcare: Technologies, Challenges, and Opportunities, IEEEAccess, Volume 5- 2017, pp.25621 – 25644, (2017).

[3] M. Sain, Y. J. kang, H. J. Lee, Survey on Security in internet of the things: state of the art and challenges, ICACT2017, pp. 699 – 704, Feb (2017).

[4] D. Rahbari, M. Nickray. Scheduling of Fog Networks with Optimized Knapsack by Symbiotic Organisms Search, preceding of the 21st conference of fruct association, (2019).

[5] A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang, and P. Liljeberg, Exploiting smart e-health gateways at the edge of healthcare internet-of-things: a fog computing approach, Future Generation Computer Systems,(2017).

[6] D. Rahbari, M. Nickray. Low-latency and energy-efficient scheduling in fog-based IoT applications, Turkish Journal of Electrical Engineering & Computer Sciences, (2019).

[7] Naina Gupta, Hera Saeed, SanjanaJha, ManishaChahande and Sujata Pandey, Study and implementation of IOT vased smart healthcare System, ICEI, pp 541-546,(2017).

[8] A. Pantelopoulos and N. Bourbakis, A survey on wearable sensor-based systems for health monitoring and prognosis, IEEE Trans. Syst. Man Cybern. C, Appl. Rev., vol. 40, no. 1, pp. 1–12, Jan. (2010).

[9] L. Savu, Cloud Computing: Deployment Models, Delivery Models, Risks and Research Challenges, Computer and Management (CAMAN), International Conference on, pp. 1-4, (2011).

[10] P., Metri, G., Sarote, Privacy Issues and Challenges in Cloud Computing, International Journal of Advanced Engineering Sciences and Technologies (IJAEST), vol. 5, pp.1-6, (2011)

[11] Shweta, PadmawatiKhandnoe, Neelesh Kumar, A Survey of Activity Recognition Process Using Inertial sensors and Smartphone Sensors, ICCCA, pp. 607-612, (2017).

[12] P. K. Schwab, The Fourth Industrial Revolution: What it Means, and How to Respond. Cologny, Switzerland: World Economic Forum, (2016).

[13] Australian Institute of Health and Welfare. Australia's Health. [Online]. Available: http://www.aihw.gov.au/WorkArea/DownloadAsset.aspx-?id=60129548150, (2014).

[14] A. Bahga and V. Madisetti, Cloud Computing: A Hands-on Approach, CreateSpace, (2013).

[15] A. Bahga and V. Madisetti, Internet of Things: A Hands-on Approach, CreateSpace.263, (2014) .

[16] A. Bahga and V. Madisetti, A Cloud-Based Approach for Interoperable Electronic Health Records (EHRs), IEEE J. Biomedical and Health Informatics, vol. 17, no. 5, pp. 894–906, (2013).

[17] J. Mohammed, C.-H. Lung, A. Ocneanu, A. Thakral, C. Jones, and A. Adler, Internet of Things: Remote patient monitoring using Web services and cloud computing, in Proc. IEEE Int. Conf. Internet Things (iThings), IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom), pp. 256–263, Sep. (2014).

[18] M. Díaz, C. Martín, and B. Rubio, State-of-the-art, challenges, and open issues in the integration of Internet of Things and cloud computing, J. Netw. Comput. Appl., vol. 67, pp. 99–117, May (2016).

[19] B. Díaz, M. Martín, and C. Rubio, λ-CoAP: An Internet of Things and cloud computing integration based on the lambda architecture and CoAP, in Proc. Int. Conf. Collaborative Comput., Netw., Appl. Work-sharing, vol. 163. Cham, Switzerland: Springer, pp. 195–206, (2015).

[20] G. Aloi et al., Enabling IoT interoperability through opportunistic smartphone-based mobile gateways, J. Netw. Comput. Appl., vol. 81, pp. 74–84, Mar. (2017).

[21] Rodolfo S. Antunes, Lucas A. Seewald, Vinicius F. Rodrigues, Cristiano A. Da Costa, Luiz Gonzaga Jr., And Rodrigo R. Righi, Andreas Maier, BjörnEskofier, MalteOllenschläger, And FarzadNaderi,RebeccaFahrig, Sebastian Bauer, Sigrun Klein, And GelsonCampanatti, A Survey Of Sensors In Healthcare workflow Monitoring", ACM Computing Surveys, Vol. 51, No. 2, Article 42, April (2018).

[22] BoyiXu, Li Da Xu, Cheng Xie, Jingyuan Hu, and Fenglin Bu, Ubiquitous Data Accessing Method in IoT-Based Information System for Emergency Medical Services, Ieee Transactions On Industrial Informatics, Vol. 10, No. 2, pp. 1578-1586, MAY (2014).

[23] M. Ashouri, F. Lorig, P. Davidsson, R. Spalazzese, Edge computing simulators for IoT system design: An analysis of qualities and metrics, Future internet, vol – 11, pp. 12, (2019).

[24] https://www.chapter247.com/key-benefits-of-cloud-computing-in-healthcare-768x768/

[25] Calheiros RN, Ranjan R, Beloglazov A, De Rose CAF, Buyya R. Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Software: Practice and experience; 41(1): 23-50. doi: 10.1002/spe.995, (2011).

[26] web reference :Edge computing now a days.

[27] JitendraSingh, Study of Response Time in Cloud Computing, IEEB, vol 5, pp 36-43, 2014.

[28] Sagar Sharma, Keke Chen and Amit Sheth, Toward Practical Privacy-Preserving Analytics for IoT and Cloud-Based Healthcare Systems, in healthcare informatics and privacy, IEEE Internet Computing, pp 1089-7801, (2018).

[29] KavitaJaiswal*IIIT et al.*, An IoT-Cloud based smart healthcare monitoring system using container based virtual environment in Edge device**,** Proceeding of ICETIETR, (2018).

[30] Prabal V, Sandeep K. Sood , SheetalKalra**,** Cloud-centric IoT based student healthcare monitoring framework**,** J Ambient Intell Human Compute, vol 9, pp 1293-1309, (2018).

[31] Maulik Parekh, Dr.Saleena B., Designing a Cloud based Framework for HealthCare System and applying Clustering techniques for Region Wise Diagnosis, Elsevier, ScienceDirect, 2nd International Symposium on Big Data and Cloud Computing, Procedia Computer Science vol 50, pp 537 – 542, (2015).

[32] Niharika Kumar, IoTArchtecture and System Design for Healthcare System, ICOSTSN pp 1118-1123, (2017).

[33] Naina Gupta, Hera Saeed, SanjanaJha, ManishaChahande and Sujata Pandey, Study and implementation of IOT vased smart healthcare System, ICEI, pp 541-546, (2017).

[34] A. Jindal, A. Dua, N. Kumar, A. V. Vasilakos, and J. J. P. C. Rodrigues, An efficient fuzzy rule-based big data analytics scheme for providing Healthcare-as-a-Service, presented at IEEE Int. Conf. Commun., Paris, France, May 21–25, pp. 1–6, (2017).

[35] QueenyIp, Daniel C. Malone, Jenny Chong, Robin B. Harris, David M. Labiner, An update on the prevalence and incidence of epilepsy among older adults, Epilepsy Research, pp. 107-112, (2018).

[36] ParthaPratim Ray, A survey of IoT cloud platforms, Future Computing and Informatics Journal, vol 1, pp. 35-46, (2017).

[37] Praful P. Pai, Pradyut K. Sanki, Sudeep K. Sahoo, Arijit De, Sourangshu Bhattacharya, and Swapna Banerjee, Cloud Computing-Based Non-Invasive Glucose Monitoring for Diabetic Care, Ieee Transactions on Circuits And Systems–I: Regular Papers, VOL. 65, NO. 2, pp. 663-676, (2018).

[38] Byungseok Kang, Daecheon Kim, and HyunseungChoo, Internet of Everything: A Large-Scale Autonomic IoT Gateway Byungseok, IEEE Transactions on Multi-Scale Computing Systems, Vol. 3, No. 3, Pp. 206-214, July-September (2017).

[39] H. Khazaei, J. Misic, and V.B. Misic, Performance Analysis of Cloud Computing Centers, Proc. Seventh Int’l ICST Conf. Heterogeneous Networking for Quality, Reliability, Security and Robustness (Qshine), (2010).