An optimal task scheduling towards minimized cost and response time in Fog computing Infrastructure

Hemant Kumar Apat
Compt Science and Engg
NIT,Rourkela
Rourkela, India
hemant.rimt@gmail.com

Bibhudatta sahoo
Compt Science and Engg
NIT,Rourkela
Rourkela,India
bibhudatta.sahoo@gmail.com

Kunal Bhaisare
Compt Science and Engg
NIT,Rourkela
Rourkela,India
kunal.bhaisare.res@gmail.com

Prasenjit Maiti
Compt Science and Engg
NIT,Rourkela
Rourkela,India
pmaiti1287@gmail.com

***Abstract*—Due to the transformation and expansion of Internet of Things(IoT), a large number of services are deployed on the edge of the network to provide the services to the end users rather than from the cloud data center since processing the data at the edge can reduce the response time and bandwidth cost while fulfilling the Quality of services(QoS). The fog-cloud computing environment offers promising solution to provision the available resources for IoT based application.Undoubtely Fog computing compliment of cloud computing helps to provide efficient solution to deal with diverse IoT application. However to provide efficient solution in such environment is challenge in different IoT based application such as health care applications, intelligent transportation system and smart cities. Task scheduling and Resource allocation are the NP-hard issues in distributed computing. Each Application consists of several modules that requires resources to execute. However, providing an optimal task scheduling policy in such a heterogeneous system is a *NP* class problem and has been proposed by different methods like Greedy, meta-heuristic and all nature inspired algorithm for solving an *NP*-complete problem. The task scheduling problem is a key challenge in the distributed computing system. In this paper, we are trying to map the independent task into the fog layer and our algorithm gives good result if we place the services in fog layer rather than cloud data center. The system resources available may be CPU, RAM, etc by assigning some priority to the task based on its deadline. Also, we have assumed that once a task assigned to a particular node will not leave that until its execution complete. In this paper, we also proposed a three-layer architecture for efficient task scheduling for application such as health care in smart homes.**

***Index Terms*—Fog computing, Task Scheduling, Internet of Things(IoT),Resource Optimization, Response Time, Cost.**

I. Introduction

Over the past couple of years, it has been considered that cloud computing is the best option to offload the computation and storage capacity to the dedicated data center[2,3]. As the data center is deployed in a predefined longitude and latitude the connection to the cloud data center depends on the location of the device, so some times this may lead to connection interrupt and faced high latency to process the real time application. The other problem in cloud computing is low bandwidth available to the application; these drawbacks enforced a new computing paradigm to get rid of these issues. The expansion and consumption of Iot devices growing exponentially which affect in our daily lives like adoption of smart health care , dependability on IoT devices for traffic control and production in agriculture.The centralized nature of cloud does not satisfies the decentralized nature of IoT. Now a days, IoTs data are mostly divergent which directly sent to cloud for further processing and then returned back to variant user or distributed IoT devices often located very near to the data sources. The major loopholes of cloud computing fashion tends to high latency even if the data source is very close to the data center and ultimately affected the QoS(Quality of services).

Fog computing, now a days has added a new dimensions to the traditional computing such as Grid computing, Parallel computing, Cloud computing etc. in terms of computation, data storage, and service provisioning[1][2][3]. However the rapid growth of IoT device which are connected to the internet challenges the traditional computing framework like cloud computing. As reported by IDC(International data corporation) we have reached the tip point of the technology where no of connected devices are more than the number of people in the world and simultaneously we have also reached the tipping point of data centers where the Iot(Internet of things) and subsequently demand of high connectivity speed and processing power has propel the needs of these capabilities close to the user forcing solution to the Edge[4]. Nowadays, the Internet of things(IoT) is playing a convincing role in our daily life. IoT devices used in almost every smart application with different domains such as smart city, health care, smart grid,video surveillance,etc. The popularity and growth of these applications, the number of IoT devices(e.g., Sensors, camera, actuators smart meters) has been increasing significantly[1,2]. Hence it has been expected that a large amount of data is generated by these devices over the period which requires real-time processing. The drawbacks of these IoT devices are limited computational power, storage, and battery life. To cope up with the current situation they have to rely on some other computing resources which offer flexible, scalable and large computational storage. Fog computing is generally distributed computing environment, which is still in its infancy stage. Despite the system has many benfits to IoT based application and has numerous challenges still existing which require more research and attention. Fog computing paradigm, that maps the devices through some hierarchy levels with various degree of computational and storage capabilities[5].

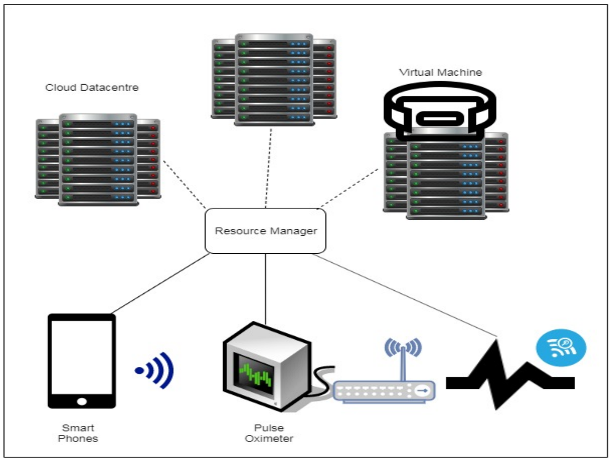

Fig. 1. Cloud-based IoT system architecture

II. Related Work

The task placement in fog computing is new research area that has recently attracted considerable attention. In this section some of the relevant work done by different authors are discussed.

In [14], Bitam et al. proposed a task scheduling method by using nature inspired algorithm Bee Life Algorithm (BLA). In this paper Authors proposed and efficient method to solve the multi-objectve optimization problem and find the non dominated solution to improve the objective function. The result obtained was quite good, but due to non-collaboration with cloud data center the obtained result cannot be more appropriate.

In [15], Topcuoglu et al.In this paper authors worked on most versatile algorithm called HEFT (Heterogeneous Earliest Finish Time) and CPOP (Critical Path On a Processor) for scheduling dependent tasks. Authors represent the task model as Directed Acyclic Graph(DAG) and calculate the upward rank of each node of the graph and the highest rank assigned first to the processor available . On the other hand, the CPOP algorithm find upward and downward rank values for prioritizing task on heterogeneous computing systems to reduce the makespan as well as cost, schedule length ratio, speedup, frequency of best results, and average scheduling time .

In [16] Dastjerdi et al. In this paper authors presented a reference architecture [9] for fog with IoT application.

In[18]Hong et al. proposed efficient heuristic deployment algorithm to solve dynamic module placement problem. Authors also implement an optimal algorithm for comparisons. They conducted experiments with a real testbed to evaluate the algorithms and fog computing platform.

Giang et al. [19]In this paper Authors examined the applicability of deployment of IoT application in fog computing environment, In this paper authors present a Distributed Data flow (DDF) programming model for the IoT that utilizes computing infrastructures across the Fog and the Cloud.

Quang Tran Minh et,al.,[20] In this paper authors proposed an optimize service placement plan to place the services on fog computing layer in context of IoT. A multi-tier fog computing architecture has been proposed by the authors that supports the IoT service. This paper mostly focuses on reducing latency, energy consumption, and load balancing in comparison with the conventional cloud computing model.

Sadoon Azizi et al.[23] In this paper Authors proposed service placement policy for fog computing by calculating the most delay sensitive application service. The proposed policy has been implemented in ifogsim Simulator[30] .The result shows the significant improvement in terms of service latency and execution cost compared to cloud computing .

Skarlat et al.[24]In this paper authors proposed a conceptual framework for service provision in fog computing which is based on the concept of fog colony which has been considered in many related works[27,28,29]. Authors formulate the service placement problem as a constrained optimization problem and considered that all the application are heterogeneous and independent.In this paper Authors present a three tier architecture for service placement in context of IoT data in fog computing environment.

III. THE SYSTEM MODEL AND PROBLEM FORMULATION

The architecture proposed here for fog computing consists of three layers: IoT layers, fog computing layer, and cloud computing layer as shown in Fig.1.The IoT layer consists of intelligent terminal devices refereed as sensor devices or IoT devices responsible for collecting geographically distributes IoT data and transfered to upper layer(fog computing) for further processing. The fog computing layer consists of different fog node that have better computing and storing capability.Fog nodes may be switches, routers and virtual Network function(VNFs) running a server using Network Virtualization (NFV) technologies. The fog computing layer received IoT data and task generated by the IoT devices and allocate computing resources as per the requirements of the task without violating Service level agreement(SLA).The fog computing layer also consists of some more powerful devices referred to as fog controller which control all the node within a cluster of nodes.As the cloud computing layer consists of high performance servers memory intensive task may go to the cloud layer for processing and executing . According to the requirement of the task and capacity of the fog resources we divide the fog computing layer into multiple fog cluster and each cluster is managed by a corresponding fog control manager which is just more powerful in terms of computation and memory controls all the fog nodes in the cluster. This fog control manager may act as gateway for the fog cluster that is, the available fog nodes cn easily communicate with cloud data center if they need .The min function of fog control manager is to manage all the fog nodes and allocate appropriate fog resources to the task sent the IoT devices and monitoring the IoT devices .

Let us assume the network topology diagram of fog nodes denoted by $FT = (N, L)$ where $N = \{N_1, N_2, N_3, ...N_n)$ represents the set of fog nodes and N_i represent the ith fog

TABLE I
PERFORMANCE METRICES IN FOG

Author	Metrics	Method
Olena Skarlat et al[25]	QoS and Resource provision in fog computing	optimization with genetic algorithm
Redowan Mahmud,et.al [31]	structural security and service related issues	taxonomy
Quang Tran Minh,et.al [20]	latency, energy consumption, and network load	optimization technique
Zhenyu Wen,et.al[32]	Security and system reliability	parallel genetic algorithm
Xianling Meng,et.al[33]	energy efficiency	Dynamic programing(Divide and conquer)
Adila Mebrek,et.al[34]	energy consumption, and Quality of service	optimization technique using Evolutionary Algorithm
H. Gupta,,et.al[29]	ifogsim,	DDF model
Bitam, et.al[14]	delay and energy consumption	Energy-Aware Offloading Clustering Approach (EAOCA)
Dastjerdi et al.[16]	End to end latency and Network Overhead	cloud with edge computing
Sarkar et al. [35]	Theoretical model of fog computing	cloud with edge computing

nodes.Also assume that that N_{P_i}= Processing rate of each fog node and L represents the link between the nodes

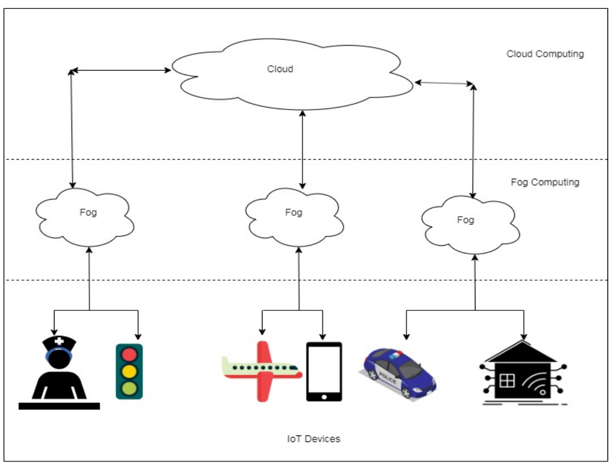

Fig. 2. Fog Computing architecture in context of IoT

(1)An IoT application consists of well defined tasks generated by an IoT devices by sensing some events occured which may be medical relted data, traffic related or temperature of vehicle etc, required to process and execute and get the result back to clients in distributed environments.

A. Cloud layer

This layer is the topmost layer of our architecture consists of cloud data center which is physically a long distance from the IoT device.Processing and executing IoT data at this layer is applicable for non emergent application which has latency is negligible factor.

IV. TASK SCHEDULING PROBLEM IN FOG COMPUTING WITH IOT BASED APPLICATION

A. General Scheduling problem

A Scheduling problem is the mapping of one set of item to another set of item that can be defined mathematically as given a set S of n tasks, a partial order $\prec$ on S,a weighting function W, a number of processors k and a time limit t, does there exist a total function f from S to $\{0, 1,..., \mathrm{t} - 1\}$such that

(i) if $\mathrm{T} \prec \mathrm{T}^{'}$, then $f(T) + W(T) \leq f(T')$

(ii) for each J in S, $f(T) + W(T) \leq \mathrm{t}$, and

(iii) for each i $0 \leq i \leq t$ there are at most k values of T for which $f(T) \leq i < f(T) + W(T)$?

from[34] it has been proved that all scheduling problem known as *NP* complete problem and it has received much more attention recently by the research communities and some papers discussing bout NP class problem are given as[34].Informally A problem is said to be in *NP* class if it is acceptable by non deterministic turing machine in polynomial time.

Task scheduling in distributed system is the process of assigning non-preemptive task to the available heterogeneous resources and produce an acceptable sequence so that total make span is minimum without violating any constraints.Task scheduling in fog computing environment also assigning the tasks in fog landscape(fog resources) available such that all the application get executed by satisfying the deadline of the application. As the application consists of dependent and independent task their task model is different for processing the application into the available processor In this paper we have assumed that all the task arrived is independent and arrival pattern of tasks follows the Poisson distribution and for buffering the task individual priority queue is maintained.

B. Case 1- Dependent task model- Input of tasks

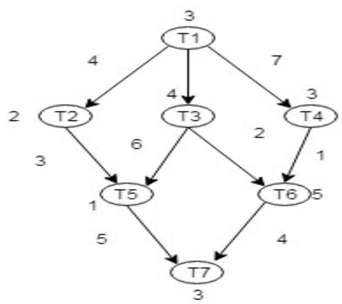

Fig. 3. Input DAG Model of Dependent task

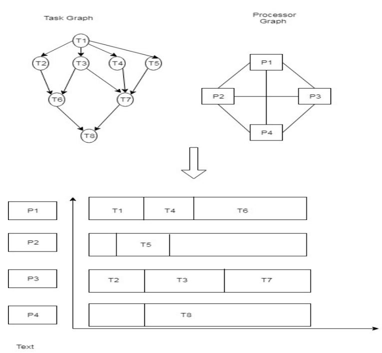

Fig. 4. Output of task scheduling problem

C. Case 2-Independent Task model

A task is denoted five tuple $\{I_s, M, I, O, D_t\}$ where
I_s=Number of Instruction
M=Memory requirement
I=Input files
O=Output files
D_t=Deadline time
A set of input tasks $T = \{T_1, T_2, T_3, ...T_n\}$
A set of processing Node consists of cloud and fog node
$P = \{P_1, P_2, P_3, ...P_n\}$

The cost of executing a task in the cloud or fog node can be computed by the equation given below:-
$Cost(T_k^i) = x_m(T_k^i) + x_p(T_k^i) + x_b(T_k^i)$.....(1)
Processing cost may be defined as
$x_p(T_k^i)$=x$_1 * ExTime(T_k^i)$.....(2)
Memory usage cost may be dined as
$x_m(T_k^i)$=x$_2 * Mem(T_k^i)$.......(3)
Bandwidth cost may be defined as
$x_b(T_k^i)$=x$_1 * Bw(T_k^i)$
Hence the total cost of executing a task in the fog node or cloud node may be defined as

$$TotalCost = \sum_{T_k^i \in NodeTasks} Cost(T_k^i)......(4)$$

V. Problem Statement

Consider a fog cluster with Fog control Manager and m fog nodes. we use the notation F for the fog node manager and the set R^f=$\{f^1, f^2, ...f^m\}$ fo the fog nodes that are inside the fog cluster. we consider each fog cluster interact with neighbor fog cluster through the fog control manager which has the lowest delay with F, and we use the notation N to identify its control node.So the total set of computational devices in the system model can be represented by D= $\{F, R^f, N, C\}$

Each of the device in the set D has different resources, such s CPU and RAM.The capacity of CPU and RAM of the control node is represented by C_F^{cpu} and C_F^{ram}.The latency of the communication link between a particular fog node f^j and the fog control node is d^j. Similarly we can assume d^N and d^C as the delay between fog control node F and cloud respectively.

Consider A=$\{A_1, A_2, ...A_n\}$ represent IoT application generated by the IoT devices for processing and executing to the fog layer.For all $A_i \in A$ consists a set of tasks $A_i = \{T_{i,1}, T_{i,2}, ...T_{i,n}\}$, where $T_{i,j}$ represents the j-th task of application i. Each task $T_{i,j}$ has different resource requirements.There are two types of computational resources CPU and RAM available. For each task $T_{i,j}$ Its CPU and RM requirements can be defined as $T_{i,j}^{cpu}$ and $T_{i,j}^{ram}$ respectively.We assumed our model work on the principle Sense-Process-Actuate.Each task belongs to one types of sensing, processing or actuating. We also assume a specific deadline D_{A_i} that is defined by the users of the Application.

We considering here service placement problem in fog landscape which can be stated as the process of mapping a set of applications$A = \{A_1, A_2, A_3, ..., A_n\}$ where each A_i has a deadline D_{A_i} that consists of n number of dependent tasks into the set of computational devices in the system which includes D= $\{F, R^f, N, C\}$

The response time of application A_i is obtained by adding the deployment time W_{A_i} and the make span M_{A_i}, which can be obtained by following equation:-

$$R_{A_i} = W_{A_i} + M_{A_i}$$

VI. PROBLEM FORMULATION

we are considering here different decision variable that we use to describe the task placement policy in the fog cloud environment. Tasks, services, and module have same meaning through out this paper and they can be used interchangeably. Consider the variable $x_{i,j}^{dev}$ indicates the task $T_{i,j}$ on which dev computing device is located, where $dev \in D$. For example, if $x_{i,j}^{f^k} = 1$ then the task $T_{i,j}$ is located on the fog cell f^k, $1 \leq k \leq n$.In mathematical forms:

$$x_{i,j}^{f^k} = \left\{ \begin{array}{l} 1 \\ 0 \end{array} \right. .....(1)$$

$$x_{i,j}^{F} = \left\{ \begin{array}{l} 1, \\ 0 \end{array} \right. .....(2)$$

$$x_{i,j}^{N} = \left\{ \begin{array}{l} 1, \\ 0 \end{array} \right. .....(3)$$

$$x_{i,j}^{C} = \left\{ \begin{array}{l} 1, \\ 0 \end{array} \right. .....(4)$$

A. Optimization Formula

We here trying to improve the response time to the users and reduce costs, by providing maximum services provided by the fog layer could be utilized well. To this end our goal is to solve the problem of task placement in a fog cloud computing environment such a way that the number of assigned services to the fog nodes be maximized. So our problem can be formulated as follows:

$$Maximise \sum_{i=1}^{n} \sum_{j=1}^{|A_i|} (\sum_{k=1}^{m} x_{i,j}^{f^k} + x_{i,j}^{F} + x_{i,j}^{N}) \text{..........}(5)$$

Subject to the following constraints:

$$\sum_{k=1}^{m} x_{i,j}^{f^k} + x_{i,j}^{F} + x_{i,j}^{N} + x_{i,j}^{C} = 1, \forall T_{i,j} \text{.....}(6)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{|A_i|} x_{i,j}^{D} S_{i,j}^{R} \leq C_{D}^{R}, \forall k \in \{1...m\}, D = \{R^f, F, N, C\} \text{...}(7)$$

$$R = \{cpu, ram\} \text{....}(8)$$

$$x_{i,j}^{f^k}, x_{i,j}^{F}, x_{i,j}^{N}, x_{i,j}^{C} \in \{0, 1\} \text{.........}(9)$$

$$R_{A_i} \leq D_{A_i} \forall A_i \in A \text{................}(10)$$

VII. PROPOSED ALGORITHM

Algorithm 1 Task Assignment Algorithm

1.Sort Application list A in descending order of preference by computing

$$\lambda_i = 1/D_{A_k} - W_k^t$$

2.Sort the Available Computational device in the list X in order of minimum length to the clients
3.For each Application A_i in list A do
4. Place A_i on F.
5. For each task in A_i do say task $T_{i,j}$
6. While task $T_{i,j}$ has not been assigned do
if $T_{i,j}^{cpu} \leq C_{placedev}^{cpu} \&\& T_{i,j}^{ram} \leq C_{placedev}^{ram}$ then
place $T_{i,j}$ on placedev:

$$C_{placedev}^{cpu} = C_{placedev}^{cpu} - T_{i,j}^{cpu};$$
$$C_{placedev}^{ram} = C_{placedev}^{ram} - T_{i,j}^{ram};$$

break;
else
placedev = next device in X

VIII. Conclusion

The primary objective of our proposed framework is to reduce response time. A comparison was conducted between the response times of our proposed work with cloud. Response time refers to the time that elapsed among task arrivals until complete execution of the tasks First we proposed a three layer architecture that supports wide variety of IoT tasks. The task generated by the IoT devices are stochastically and we calculate the most delay sensitive task . In the first phase we optimize the response time of an application where the independent task of an application are placed into the fog layer, second we also try to minimize the network service utilization which tends to cost reduction in fog computing system. Furthermore, our algorithm is much more effective utilizing fog landscape resources. In future we will improve our thinking and apply more especially evolutionary algorithm. In addition, we will try to expand some more parameters like bandwidth, transmission cost user perspective and energy consumption.

IX. Simulation and Results

We have implemented our algorithm on the iFogSim simulator[30] and compared its performance with cloud computing strategies. In cloud computing strategies all application services run on the centralized data center. In this strategy the sense-process actuate principle works and sensor sends the received data to the cloud and after processing it comes back to client via actuators.The result given below represent the average response time, network utilization and energy consumption in fog computing system which is substantially lower than cloud computing system.

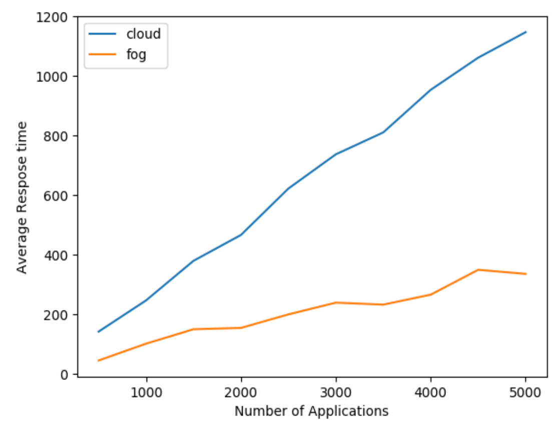

Fig. 5. Average Response time

References

[1] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A Break in the Clouds: Towards a Cloud Definition," SIGCOMM Comput Commun Rev, vol. 39, no. 1, pp. 50–55, Dec. 2008.

[2] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud Computing: State-of the-art and Research Challenges," J. Internet Serv. Appl., vol. 1, no.1, pp. 7–18, May 2010.

[3] N. Chen, Y. Chen, S. Song, CT. Huang, X. Ye, " Poster Abstract: Smart Urban Surveillance Using Fog Computing," 2016 IEEE/ACM Symposium on Edge Computing (SEC), pp. 73–78, doi:10.1109, 2016

[4] Pham, Xuan-Qui, and Eui-Nam Huh. "Towards task scheduling in a cloud-fog computing system." 2016 18th Asia-Pacific network operations and management symposium (APNOMS). IEEE, 2016.

[5] Kamyab Khajehei, "Role of virtualization in cloud computing",International Journal of Advance Research in Computer Science and Management Studies, Volume 2, Issue 4, April 2014.

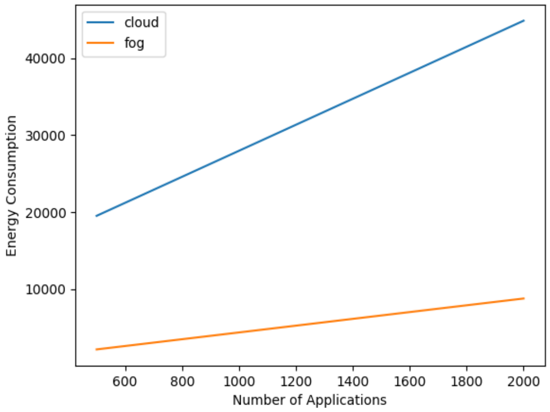

Fig. 6. Energy Consumption

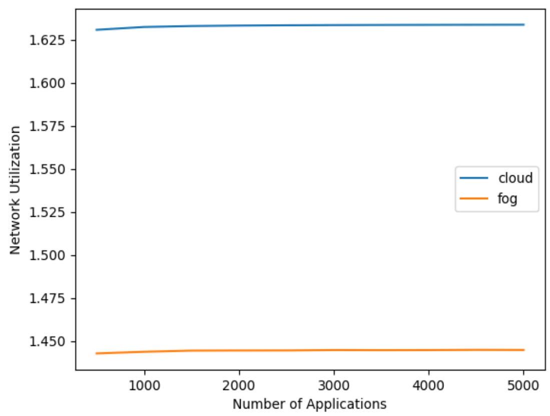

Fig. 7. Network Utilization

[6] Pham, Xuan-Qui, and Eui-Nam Huh. "Towards task scheduling in a cloud-fog computing system." 2016 18th Asia-Pacific network operations and management symposium (APNOMS). IEEE, 2016

[7] B. Wickremasinghe, "Cloud Analyst: A CloudSim Based Tool ForModeling And Analysis Of Large Scale Cloud Computing Environments", MEDC Project Report 2009

[8] Ullman J. NP-complete scheduling problems. J Comput-Syst Sci 1975; 10(3): 384–393, http://www.sciencedirect.com/science/article/pii/S0022000075800080

[9] S.Sindhu, and S. Mukherjee, "Efficient task scheduling algorithms for cloud computing environment," in High Performance Architecture and Grid Computing, Communications in Computer and Information Science, 2011, vol. 169, 2011, pp. 79-83

[10] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog Computing and Its Role in the Internet of Things," in Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, New York, NY, USA, 2012, pp. 13–16.

[11] Arabnejad H and Barbosa JG. List scheduling algorithm for heterogeneous systems by an optimistic cost table. IEEE T Parall Distr 2014; 25(3): 682–694.

[12] Wang G, Wang Y, Liu H, et al. HSIP: a novel task scheduling algorithm for heterogeneous computing. Sci Programming 2016; 2016: 3676149, http://dx.doi.org/ 10.1155/2016/3676149

[13] Wang G, Wang Y, Liu H, et al. HSIP: a novel task scheduling algorithm for heterogeneous computing. Sci Programming 2016; 2016: 3676149, http://dx.doi.org/ 10.1155/2016/3676149

[14] Bitam, Salim, Sherali Zeadally, and Abdelhamid Mellouk. "Fog computing job scheduling optimization based on bees swarm." Enterprise Information Systems 12.4 (2018): 373-397.

[15] Topcuoglu H, Hariri S and Wu MY. Performance-effective and low-complexity task scheduling for heterogeneous computing. IEEE T Parall Distr 2002; 13(3): 260–274, http://dx.doi.org/10.1109/ 71.993206

[16] Dastjerdi, Amir Vahid, and Rajkumar Buyya. "Fog computing: Helping the Internet of Things realize its potential." Computer 49.8 (2016): 112-116

[17] Garey, Michael R., and David S. Johnson. "Crossing number is NP-complete." SIAM Journal on Algebraic Discrete Methods 4.3 (1983): 312-316.

[18] Hong, Hua-Jun, Pei-Hsuan Tsai, and Cheng-Hsin Hsu. "Dynamic module deployment in a fog computing platform." 2016 18th Asia-Pacific Network Operations and Management Symposium (APNOMS). IEEE, 2016

[19] Giang, Nam Ky, et al. "Developing iot applications in the fog: A distributed dataflow approach." 2015 5th International Conference on the Internet of Things (IOT). IEEE, 2015.

[20] Minh, Quang Tran, et al. "Toward service placement on Fog computing landscape." 2017 4th NAFOSTED conference on information and computer science. IEEE, 2017.

[21] M. Maheswaran, S. Ali, H. J. Siegel, D. Hensgen, and R. F.Freund, "Dynamic mapping of a class of independent tasks onto heterogeneous computing systems," Journal of Parallel and Distributed Computing, vol. 59, no. 2, pp. 107–131, 1999.

[22] T. D. Brauny, H. Siegely, N. Becky et al., "A Comparison Study of Static Mapping Heuristics for a Class of Meta-tasks on Heterogeneous Computing Systems," parallel distributed computing, vol. 61, no. 6, pp. 810–837, 2001.

[23] Azizi, Sadoon, Fariba Khosroabadi, and Mohammad Shojafar. "A priority-based service placement policy for Fog-Cloud computing systems."

[24] Apat, Hemant Kumar, Bibhudatta Sahoo, and Prasenjit Maiti. "Service Placement in Fog Computing Environment." 2018 International Conference on Information Technology (ICIT). IEEE, 2018.

[25] Iorga, M.; Goren, N.; Feldman, L.; Barton, R.; Martin, M.; Mahmoudi, C. Fog Computing Conceptual Model; NIST: Gaithersburg, MD, USA, 2018.

[26] Skarlat, Olena, et al. "Towards qos-aware fog service placement." 2017 IEEE 1st international conference on Fog and Edge Computing (ICFEC). IEEE, 2017.

[27] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski, and P. Leitner,Optimized IoT service placement in the fog, Service Oriented Computing and Applications, 11(4) (2017), 427-443

[28] Q. T. Minh, D. T. Nguyen, A. Van Le, H. D. Nguyen, and A. Truong, Toward service placement on Fog computing landscape, 4th NAFOSTED conference on information and computer science, (2017), 291-296

[29] C. Guerrero, I. Lera, and C. Juiz, On the Influence of Fog Colonies Partitioning in Fog Application Makespan, 2018 IEEE 6th International Conference on Future Internet of Things and Cloud (FiCloud), (2018), 377-384

[30] H. Gupta, A. Vahid Dastjerdi, S. K. Ghosh, and R. Buyya, iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments, Software: Practice and Experience, 47(9) (2017), 1275–1296.

[31] Mahmud, Redowan, Ramamohanarao Kotagiri, and Rajkumar Buyya. "Fog computing: A taxonomy, survey and future directions." In Internet of everything, pp. 103-130. Springer, Singapore, 2018.

[32] Wen, Zhenyu, et al. "Fog orchestration for internet of things services." IEEE Internet Computing 21.2 (2017): 16-24.

[33] Meng, Xianling, Wei Wang, and Zhaoyang Zhang. "Delay-constrained hybrid computation offloading with cloud and fog computing." IEEE Access 5 (2017): 21355-21367.

[34] Mebrek, Adila, Leila Merghem-Boulahia, and Moez Esseghir. "Efficient green solution for a balanced energy consumption and delay in the IoT-Fog-Cloud computing." 2017 IEEE 16th International Symposium on Network Computing and Applications (NCA). IEEE, 2017

[35] Ullman, Jeffrey D. "NP-complete scheduling problems." Journal of Computer and System sciences 10.3 (1975): 384-393.