A Review on Fog Computing, Resource Allocation Strategies and its Applications

[1]S.Harika,[2]Dr.B.Chaitanya Krishn

ABSTRACT-- *Fog computing is a paradigm that provides services to user requests at the edge networks. As the definition suggests, the fog computing platform lies between the cloud servers and the users. In a fog-enabled environment, the devices at the fog layer usually perform operations related to networking such as routers, gateways, bridges, and hubs. Researchers envision these devices to be capable of performing both computational and networking operations, simultaneously. Although these devices are resource-constrained compared to the cloud servers, the geological spread and the decentralized nature help in offering reliable services with coverage over a wide area. Further, with fog computing, manufacturers and service providers offer their services at affordable rates. Another advantage of fog computing is the physical location of the devices, which are much closer to the users than the cloud servers. Such placement of the devices reduces operational latency significantly. In particular, Fog Computing refers to a distributed computing infrastructure confined on a limited geographical area within which some Internet of Things (IoT) applications/services run directly at the network edge on smart devices having computing, storage, and network connectivity, named Fog Nodes, with the goal of improving efficiency and reducing the amount of data that needs to be sent to the Cloud for massive data processing, analysis and storage. This paper proposes an efficient strategy on resource allocations and also defines and specifies the applications of fog computing.*

Keyywords-- *A Review on Fog Computing, Resource Allocation Strategies and its Applications*

I. INTRODUCTION

Multiple numbers of devices can be easily accessible using IoT technology through immense connectivity of internet. Cloud computing empower people to use services and hardware at faraway distances which is managed by a arbitrators. Examples should be internet stockpile, social media trademarks, web-mail, and online trade requests. This paradigm permits to ingress the information from anywhere, anyplace that having a grid generation continuously. From the application viewpoint, divergent devices data congregate and dump at desperate locations uniquely and articulation to use the expedient should be non-identical.

In the era of big date based computation-intensive applications, data generated from SIoT devices are generally processed in a cloud infrastructure. However, it may be inefficient to send the large data of IoT devices to the remote cloud system, especially for time-sensitive applications. To address this issue, Fog Computing (FC) method has exhibited the right features. FC provides low-latency computing cloud services at the edge of the network where data is generated. To implement the FC architecture, edge agents are evolved to the edge clouds, called

[1] *Research scholar,Department of cse,KL University*

[2] *Assoc. ProressorDepartment of cse, KL University*

Received: 27 Mar 2020 | Revised: 20 Apr 2020 | Accepted: 05 May 2020

cloudlets, by being equipped with a certain computation power capability. Therefore, it is an enabler for the emerging SIoT systems while handling the rapid development of computation-intensive applications. Compared to the traditional cloud computing, the FC method with SIoT system can greatly improve the efficiency by providing the powerful computing resource through one-hop wireless connections

Recently, a new trend has been happening; that is, the computing is pushed to the network edge devices due to their progressively enhanced computation capacity. This is called mobile fog computing (MFC) or mobile edge computing (MEC) [2, 3], where the network edge devices perform computing tasks instead of remote clouds. Thanks to closer distance to end users, the latency is less and thus real-time tasks can be achieved via MFC, which is an effective supplement to cloud computing. MFC is applicable to delay sensitive tasks while cloud computing to sophisticated but delay-insensitive data processing work.

Game Theory (GT) and Matching Theory (MT) has emerged as powerful mathematical tools to model and solve a large class of tasks offloading problems. In particular, the authors formulate a computation offloading game to model the competition between EDs and provide an efficient allocation of the FNs computation capabilities. The goal here is to reduce computation energy consumption and delay. Due to the high time complexity to reach the equilibrium, a near-optimal (i.e., sub-optimal) resource allocation scheme is also outlined and investigated in by resorting to computer simulations. MT, in its turn, has been considered as a promising mathematical framework recently used as an alternative to the GT in performing resources allocation for wireless networks. An important property of the MT is the capability of supporting distributed solutions, overcoming the typical limits of other well known approaches as global optimization and GT, usually resulting in high complexity and signaling overhead.

In this paper, we focus on the computing resource allocation problem in fog computing networks and the main objective of this paper is to develop resource allocation strategy to maximize the cost efficiency. We organize our paper as follows. Related work is addressed in Section 2. In Section 3 , we describe the fog computing network model. In Section 4, we discuss about the applications of fog computing. In Section 5, we present the resource allocation strategy. Finally, Section 6 concludes the paper.

II. RELATED WORK

*A. **OVERVIEW ON FOG:***

Fog computing – a term originally coined by Cisco—is in many ways synonymous with edge computing. In contrast to the cloud, fog platforms have been described as dense computational architectures at the network's edge. Characteristics of such platforms reportedly include low latency, location awareness and use of wireless access. Benefits include real-time analytics and improved security. While edge computing or edge analytics may exclusively refer to performing analytics at devices that are on, or close to, the network's edge, a fog computing architecture would perform analytics on anything from the network center to the edge.

*B. **NEED FOR FOG:***

IoT dispenses with big data investigative, Mobile and cloud computing where IoT is trending automation. Internet of things guarantees to implement high-range functions related to homes, automation of vehicles as smart works. To increase the possibilities and requirements evaluation, IoT needs a maturing creativity to make an application reliable at gateway functionalities.

IoT collaborates with cloud to store the data gathered from appliances whereas it requires backup and more time to perform an action regarding implementing application of IoT. Hence, it's become a drawback; IoT needs a progress in smart way and need quick action. CISCO coined to invent fog computing. Fog computing emeses with IoT applications whereas the fog nodes can analyze the time-period data at edge of networks and sends the data which is selected by fog to cloud for historical and future employment.

C. FOG COMPUTING VERSUS CLOUD COMPUTING

Many use the terms fog computing and edge computing interchangeably, as both involve bringing intelligence and processing closer to where the data is created. However, the key difference between the two is where the intelligence and compute power is placed.

a. Cloud Computing*:*

1. Data and applications are processed in a cloud, which is time consuming task for large data.

2. Problem of bandwidth, as a result of sending every bit of data over cloud channels.

3. Slow response time and scalability problems as a result or depending servers that are located at remote places.

b. Fog Computing*:*

1. Rather than presenting and working from a centralized cloud, fog operates on network edge. So it consumes less time.

2. Less demand for bandwidth, as every bit of data's were aggregated at certain access point instead of sending over cloud channels. 3. By selling small servers called edge servers in visibility of users, it is possible for a fog computing platform to avoid response time and scalability issues.

*D. **APPLICATIONS OF FOG COMPUTING***

Fog computing is an emerging technology that is basically used for Internet of Things .Fog computing fetches data and services from network centre to the network edge. Similar to Cloud, data, compute, storage, application services are given to the end-users by the fog.

1.HEALTHCARE: Patient monitoring system in real time critical care units.

2.OIL & GAS: Pipeline monitoring for leaks, fire, theft etc.

3.ENERGY MANAGEMENT: Smart grid control with switching between alternative energy sources. 4.AGRICULTURE: Smart farms with crop monitoring and irrigation control systems.

5TRANSPORTATION: Fleet management and vehicle health monitoring systems for trucks, buses. 6.RETAIL: Tracking of shopping carts and automatic billing systems. 7.SMART HOMES: Safety and comfort systems like fire alarms, climate control, intruder detection.

III. FOG COMPUTING SYSTEM MODEL

The fog computing architecture is highly scalable, is an abstract entity, and delivers different levels of services to the cloud user, achieve economies of scale, and delivers on- demand and dynamic contents and services through virtualization. Fog computing offers cloud resources (e.g., servers, networks, applications, storage, and services) over the Internet, which can be rapidly provisioned and released with minimal management effort or service provider interaction. The system architecture of a single host in the fog server (cloud) with three layers is depicted in Fig. 1. The hardware layer consists of raw hardware resources (e.g. processor, main memory, secondary storage, and network bandwidth), which are virtualized. VMM or hypervisor like Xen, VMware, UML, and Denali act as an interface between the host operating system and VMs. The VMM also allows multiple operating systems to run applications on a single hardware platform concurrently. Different number of heterogeneous applications can run on each guest operating system or VM.

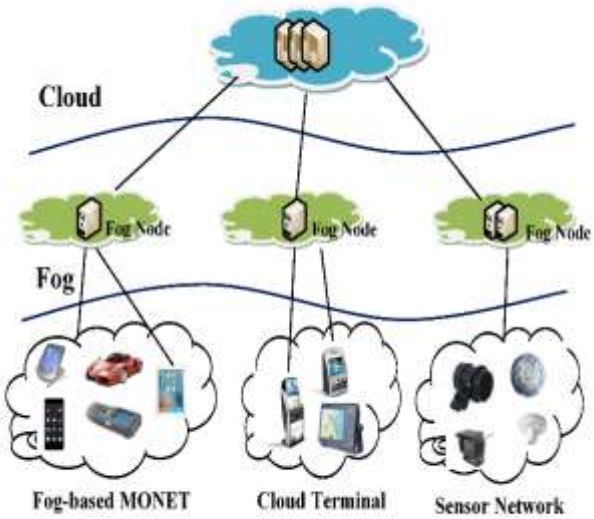

Figure 1: Three-layer fog computing architechture.

We consider a three-layer fog computing network where each user can submit computing service to a set of neighboring FNs deployed by a set of CDCs. We assume that there are total M CDCs, labeled as C = {c1, c2, ···, cM}. In the virtualized network, all the CDCs serve N users, labeled as U = {u1, u2, ···, uN }. Furthermore, in the physical network, in order to reduce the service delay and achieve real-time fast- response interaction, each CDC can offload its service submitted by users to the local FNs. We assume that there are K FNs in the network, labeled as F = {f1, f2, ..., fK}. Let λn be the workload arrival rate of user λn, ∀n ∈ {1, 2, ···, N}, and define the unit amount of computing resources that can be distributed by each FN as computing re-source block (CRB), which provides computing service at the rate of μ. However, since the users cannot access the CRBs directly, the users are required to receive the virtualized services from the CDCs, and with the management of CDCs, the CRBs of the FNs can finally be allocated to the users. The physical data transmission network between FNs and users satisfies the SecondNet topology [23], where the network resources can be guaranteed for the data center services.

As illustrated in Fig. 1, the interactions among CDCs, FNs, and users can be described as follows.

1. Interaction between CDCs and users. In fog computing networks, each users can only access to the CDCs in a virtualized link, and the CDCs provide virtualized CRBs for the users. Different users have various tolerance of service delay,

which affects the decision for the number of CRBs. In addition, the cost of the provided CRBs set by CDCs is another key metric for a higher utility.

2. Interaction between CDCs and FNs. In addition, the FNs are only accessible to the CDC, too. After providing virtualized CRBs to the users, the CDCs needs to pair FNs distributedly and try to offload the services of users to the FNs by renting the CRBs from the FNs in a physical link. The cost of the rented CRBs from FNs is the critical criterion for CDCs, and a reasonable cost is necessary to maximum utility.

3. Interaction between FNs and users. The distance between each FN and each user is different, which will affects the transmission cost. Therefore, within one CDC, the pairing between FNs and users needs to be considered.

We assume that different CDCs offer data services with different requirements and each user has a preference list over all CDCs, denoted as

$L^{user}_n = [L_{c1}, L_{c2}, \cdots, L_{cM}], \forall n \in \{1, 2, \cdots, N\}$, (1)

and hence the each user is required to subscribe to at most one CDC.

In addition, according to the services prices set by all FNs, each CDC has a preference list on each FN while each FN also has different preference on all CDCs with different relations and trading history, i.e.,

$L^{CDC}_m = [L_{f1}, L_{f2}, \cdots, L_{fK}], \forall m \in \{1, 2, \cdots, M\}$, (2) and

$L^{FN}_k = [L_{c1}, L_{c2}, \cdots, L_{cM}], \forall k \in \{1, 2, \cdots, K\}$. (3)

Furthermore, according to the transmission distance, each FN also has a preference list over all users while each user have preference over FNs based on the rent, i.e.,

$L^{FN}_k = [L_{u1}, L_{u2}, \cdots, L_{uN}], \forall k \in \{1, 2, \cdots, K\}$, (4)

and

$L^{user}_n = [L_{f1}, L_{f2}, \cdots, L_{fK}], \forall n \in \{1, 2, \cdots, N\}$. (5)

According to the preferences, each user purchases the optimal number of virtualized CRBs from the CDCs first. Then, CDCs and FNs needs to be paired to decide the CRBs for users. The most important step is achieving an FN-user pairing result with optimal number of CRBs.

IV. APPLICATIONS OF FOG COMPUTING

***Smart Traffic Lights**:*

Video camera that senses an ambulance flashing lights can automatically change street lights to open lanes for the vehicle to pass through traffic. Smart street lights interact locally with sensors and detect presence of pedestrian and bikers, and measure the distance and speed of approaching vehicles. Intelligent lighting turns on once a sensor identifies movement and switches off as traffic passes. Neighboring smart lights serving as Fog devices coordinate to create green traffic wave and send warning signals to approaching vehicles. Wireless access points like Wi-Fi, 3G, road-side units and smart traffic lights are deployed along the roads. Vehicle-to-Vehicle, vehicle to access points, and access points to access points interactions enrich the application of Fog computing.

Received: 27 Mar 2020 | Revised: 20 Apr 2020 | Accepted: 05 May 2020

***Connected car*:**

Autonomous vehicle is the new trend taking place on the road. A software is used to add automatic steering, enabling literal "hands free" operations of the vehicle. Starting out with testing and releasing self-parking features that don't require a person behind the wheel. Fog computing will be the best option for all internet connected vehicles why because fog computing gives real time interaction. Cars, access point and traffic lights will be able to interact with each other and so it makes safe for all. At some point in time, the connected car will start saving lives by reducing automobile accidents.

Smart Grids:

Smart grid is another application where fog computing is been used. Based on demand for energy, its obtained ability and low cost, these smart devices can switch to other energies like solar and winds. The edge processes the data collected by fog collectors and generate control command to the actuators. The filtered data are consumed locally and the balance to the higher tiers for visualization, real-time reports and transactional analytics. Fog supports semi-permanent storage at the highest tier and momentary storage at the lowest tier.

Self Maintaining Train:

Another application of fog computing is self maintaining trains. A train ball-bearing monitoring sensor will sense the changes in the temperature level and any disorder will automatically alert the train operator and make maintenance according to. Thus we can avoid major disasters.

Wireless Sensor and Actuator Networks (WSAN):

The real Wireless Sensor Nodes (WSNs), were designed to extend battery life by operating at predominantly low power. Actuators serves as Fog devices which control the measurement process itself, the consistency and the oscillatory behaviors by creating a closed-loop system. For example, in the lifesaving air vents sensors on vents monitor air conditions flowing in and out of mines and automatically change air-flow if conditions become dangerous to miners. Most of these WSNs entail less bandwidth, less energy, very low processing power, operating as a sink in a unidirectional fashion.

Smart Building Control:

In decentralized smart building control wireless sensors are installed to measure temperature, humidity, or levels of various gaseous components in the building atmosphere. Thus information can be exchanged among all sensors in the floor and the reading can be combined to form reliable measurements. Using distributed decision making the fog devices react to data. The system gears up to work together to lower the temperature, input fresh air and output moisture from the air or increase humidity. Sensors respond to the movements by switching on or off the lights. Observance of the outlook the fog computing are applied for smart buildings which can maintain basic needs of conserving external and internal energy.

IoT and Cyber-Physical Systems (CPSs):

Fog computing has a major role in IoT and CPSs. IoT is a network that can interconnect ordinary physical objects with identified address using internet and telecommunication. The characteristic of CPSs is the combination of system's computational and physical elements. The association of CPSs and IoT will transform the world with computer based control and communication systems, engineered systems and physical reality. The object is to integrate the concept and precision of software and networking with the vibrant and uncertain environment. With the growing cyber physical systems we will be able to develop intelligent medical devices, smart buildings, agricultural and robotic systems.

Software Defined Networks (SDN):

SDN is a growing computing and networking concept. SDN concept together with fog computing will resolve the main issues in vehicular networks irregular connectivity, collisions and high packet loss rate.SDN supports vehicle to-vehicle with vehicle-toinfrastructure communications and main control. It splits control and communication layer, control is done by central server and server decides the communication path for nodes.

Benefits of Fog Computing

Extending the cloud closer to the things that generate and act on data benefits the business in the following ways:

- ***Greater business agility:***

With the right tools, developers can quickly develop fog applications and deploy them where needed. Machine manufacturers can offer MaaS to their customers. Fog applications program the machine to operate in the way each customer needs.

- ***Better security:***

Protect your fog nodes using the same policy, controls, and procedures you use in other parts of your IT environment. Use the same physical security and cyber security solutions.

- ***Deeper insights, with privacy control:***

Analyze sensitive data locally instead of sending it to the cloud for analysis. Your IT team can monitor and control the devices that collect, analyze, and store data.

- ***Lower operating expense:***

Conserve network bandwidth by processing selected data locally instead of sending it to the cloud for analysis.

V. RESOURCE ALLOCATION STRATEGY IN FOG COMPUTING NETWORKS

A. ***Double-Matching Resource Allocation Problem***

We assume that each user has purchased the optimal numberof virtualized CRBs from the CDCs. According to the preferences based on cost efficiency, CDC-FN and FN-user need to be paired. These two pairing problems can be formulated in a unified form as a binary-integer programming problem based on the system metric maximization.

Let x i,j be a binary variable. X i,j = 1, if the jth object on the side is paired with the ith object on the other side, otherwise, xi,j = 0. Mathematically, the pairing problem can be formulated as max I i=1 J j=1xi,jηi,j , (17a) s.t. Jj=1xi,j = 1, ∀i, (17b)Ii=1xi,j = 1, ∀j, (17c) where constraints (17b) and (17c) indicate that each object on the side can be only allocated to one object on the other side.

In the formulated problem, the cost efficiency is adopted as the metric, i.e., ηi,j . With the consideration of double two-sided matching, we need to get the cost efficiency in CDC-FN pairing and FN-user pairing, respectively. In CDC-FN paring, the cost efficiency is defined by the ratio between the utility of CDC and FN and the cost, and each user served by (m, k) CDC-FN pair should be considered.Thus, the cost efficiency can be give asηi,j |CDC−F N = ηm,k = N n=1U CDC m |n + U F N k τnmτnkCD n, (18)and the cost efficiency of FN-user

Received: 27 Mar 2020 | Revised: 20 Apr 2020 | Accepted: 05 May 2020

pairing can be defined by the ratio between the utility of FN and user and the cost as ηi,j |F N−user = ηk,n = U F Nk + U USER n Ni=1 τikCD I , (19)where the utility of CDC, FN and user can be given by (11), (13) and (15), respectively.

The problem form in (17) which include two pairing problems is a classical integer linear programming problem, which is NP hard. However, in three-layer fog computing networks, the two pairing problem cannot be considered independently, since the selected CDC will affect the pairing between FN and user. Thus, we reformulate the two pairing problem as a unitary double- matching optimization problem as max m=1Kk=1Nn=xm,k,n · ηm,k,n, (20a)s.t. m=1Kk=1xm,k,n = 1, ∀n, (20b)where the binary variable, xm,k,n = 1, if the CDC cm serve user un by offloading the service of user un to the FN fk, otherwise, xm,k,n = 0. Here, we assume that the CRBs of one user can be supported by only one CDC and only one corresponding FN while both each CDC and FN can serve multipleusers. Therefore, the constraint (20b) implies that each user can be only assigned to one CDC and one FN. The unitary metric, ηm,k,n, can be given based the description above as ηm,k,n = Utotal m,k,n CD n , (21)where Utotal m,k,n, given in (22), denotes the total utility when cm, fk, and un are paired to allocate resources. U CDC m |k,n denotes the utility of CDC cm for FN fk and user un. U F N k |n denotes the utility of FN fk for user un. U USER n |m denotes the utility of user un for CDC cm. Thus, the double-matching problem is more complex than problem (17) and is also an NP-hard problem.

Next, we will study this problem from a novel perspective and develop an efficient heuristic algorithm. Actually, the formulated double-matching problem is a two-step matching problem as illustrated in Fig. 3, where each user has purchased the optimal number of virtualized CRBs from the CDCs, and the purchased CRBs needs to be allocated to CDCs and FNs. Thus, for each CRB of all user, one CDC and one FN need to be selected.

Therefore, based on the cost efficiency, each user is matched with one CDC and one FN.Furthermore, the matching is based the preferences among CDCs, FNs and users. The preferences are calculated according to the cost efficiency as given in (21). As shown in Fig. 3, each CDC has a preference list for all FNs, and the green line of dashes denotes the preferred FN for each CDC. Similarly, each FN has a preference for all CDCs, and FNs and users have the preference mutually. The double-matching resource allocation is to find optimal CDC-FN-user pairs for total users to maximize the whole cost efficiency as far as possible.

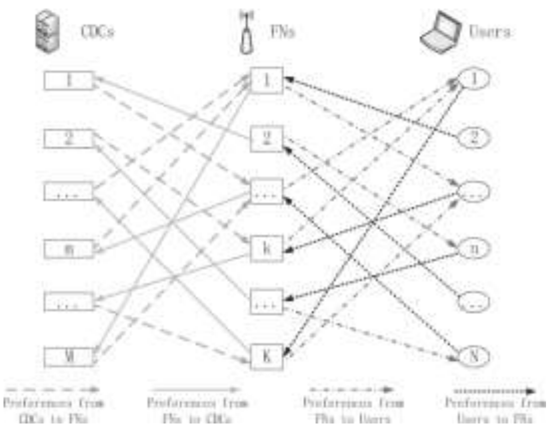

Figure 2: Double matching resource allocation.

VI. CONCLUSION

In this paper we saw an overview of fog computing in section 1, Introduction. Then in section 2 we saw the related works including overview on fog, need for fog, and the differences between fog computing and cloud computing. In section 3 we saw the system model of fog computing. In section 4, we discussed different applications of fog computing like smart traffic lights, connecting cars, smart grids, smart building control, self maintaining trains WSAN, CPSs, SDN, then the benefits of fog computing. In section 5, we discussed about the double matching resource allocation.

REFERENCES

1. D. Puthal, X. Wu, S. Nepal, R. Ranjan, and J. Chen," SEEN: A Selective Encryption Method to Ensure Confidentiality for Big Sensing Data Streams." IEEE Transactions on Big Data (In Press), 2017.
2. R. Deng, R. Lu, C. Lai, T. H. Luan, and H. Liang, "Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption," IEEE Internet of Things Journal, 3(6), pp. 1171-1181,2016.
3. J. Kang, R. Yu, X. Huang, and Y. Zhang,"Privacy-Preserved Pseudonym Scheme for Fog Computing Supported Internet of Vehicles," IEEE Transactions on Intelligent Transportation Systems, 2017.
4. G. S. Aujla, N. Kumar, A. Y. Zomaya, and R. Rajan, "Optimal Decision Making for Big Data Processing at Edge-Cloud Environment: An SDN Perspective," IEEE Transactions on Industrial Informatics, 2017.
5. A. Bouchentouf, "Commodities for dummies," John Wiley & Sons, 2011.
6. Y. M. Al-Saleh, G. Vidican, L. Natarajan, and V. V. Theeyattuparampil, "Carbon capture, utilisation and storage scenarios for the Gulf Cooperation Council region: A Delphi-based foresight study," Futures, 44(1), pp. 105-115, 2012.
7. M. D. Dikaiakos, D. Katsaros, P. Mehre, G. Pallis, A. Vakali, "Cloud computing: Distributed internet computing for it and scientific research," Internet Computing, IEEE, 13(5), pp. 10-13, 2009.
8. S. K. Mishra, D. Puthal, B. Sahoo, P. Jayaraman, S. Jun, A. Y.Zomaya, R. Ranjan. "Energy-Efficient VM-Placement in Cloud Data Center." Sustainable Computing: Informatics and Systems (SUSCOM), (Accepted), 2018.
9. S. K. Mishra, D. Puthal, B. Sahoo, S. K. Jena, and M. S. Obaidat, An adaptive task allocation technique for green cloud computing. The Journal of Supercomputing, pp. 1-16, 2017.
10. L. Guo, S. Zhao, S. Shen, C. Jiang, "Task scheduling optimization in cloud computing based on heuristic algorithm," Journal of Networks, 7(3), pp. 547-553, 2012.
11. D. A. Chekired, and L. Khoukhi, "Smart Grid Solution for Charging and Discharging Services Based on Cloud Computing Scheduling," IEEE Transactions on Industrial Informatics, 2017.
12. T. Kaur, I. Chana, "Energy efficiency techniques in cloud computing: A survey and taxonomy," ACM Computing Surveys (CSUR), 48(2), pp.22, 2015.
13. L. Zuo, L. Shu, S. Dong, C. Zhu, and T. Hara, A Multi-ObjectiveOptimization Scheduling Method Based on the Ant Colony Algorithmin Cloud Computing, IEEE Access, 3, pp. 2687-2699, 2015.
14. L. Zuo, L. Shu, S. Dong, C. Zhu, and G. Han, A Multi-queue Interlacing Peak Scheduling Method Based on Tasks Classification in Cloud Computing, IEEE Systems Journal, 2016.

15. NRDC.https://www.nrdc.org/experts/pierre-delforge/new-study-americas-data-centers-consuming-and-wasting-growing-amounts-energy. 2013.
16. V. B. Souza, X. Masip-Bruin, E. Marin-Tordera, W. Ramirez, andS. Sanchez, “Towards Distributed Service Allocation in Fog-to-Cloud (F2C) Scenarios,” In Global Communications Conference (GLOBECOM), IEEE, pp. 1-6, 2016.
17. N. Bergmann, Y. Y. Chung, X. Yang, “Using swarm intelligence tooptimize the energy consumption for distributed systems’, Modern Applied Science, 7(6), pp. 59-66, 2013.
18. C. W. Tsai, J. J. Rodrigues, “Metaheuristic scheduling for cloud: A survey,” Systems Journal, IEEE, 8(1), pp. 279-291, 2014.
19. Z. H. Zhan, X. F. Liu, Y. J. Gong, J. Zhang, H. S. H. Chung, and Y. Li, “Cloud computing resource scheduling and a survey of its evolutionary approaches,” ACM Computing Surveys (CSUR), 47(4), pp. 63, 2015.
20. INFRARATI. https://infrarati.wordpress.com/2014/03/25/data-center-co2-emissions/. 25 March 2014.
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, 59(2), pp. 107-131, 1999.
22. A. M. Sampaio, J. G. Barbosa, R. Prodan, “Piasa: A power and interference aware resource management strategy for heterogeneous work-loads in cloud data centers,” Simulation Modelling Practice and Theory, 57, pp. 142-160, 2015.
23. L. Zuo, L. Shu, S. Dong, C. Zhu, and Z. Zhou, Dynamic Weighted Load Evaluation Model Based on Self-adaptive Threshold in Cloud Computing, Mobile Networks and Applications, 22(1), pp. 4-18, 2017.
24. T. Shi, M. Yang, X. Li, Q. Lei, and Y. Jiang, “An energy-efficient scheduling scheme for time-constrained tasks in local mobile clouds,” Pervasive and Mobile Computing, Elsevier, Vol-27, pp. 90-105, 2016.
25. M. R. Garey, D. S. Johnson, “Computers and intractability: a guide to the theory of np-completeness,” WH Freemann, New York, San Francisco,1979.
26. D. Klusacek, H. Rudova, “A metaheuristic for optimizing the performance and the fairness in job scheduling systems,” Artificial Intelligence Applications in Information and Communication Technologies, Springer, pp. 3-29, 2015.
27. X. Dai, J. M. Wang, B. Bensaou, “Energy-efficient virtual machines scheduling in multi-tenant data centers,” pp. 210-221, 2016.
28. Z. Wang, K. Shuang, L. Yang, and F. Yang, “Energy-aware and revenue enhancing Combinatorial Scheduling in Virtualized of Cloud Datacenter,” Journal of Convergence Information Technology, 7(1), pp. 62-70,2012.
29. S. Raghavan, P. Sarwesh, C. Marimuthu, and K. Chandrasekaran, “Bat algorithm for scheduling workflow applications in cloud,” In IEEE International Conference on Electronic Design, Computer Networks & Automated Verification (EDCAV), pp. 139-144, 2015.
30. T. S. Somasundaram, K. Govindarajan,“CLOUDRB: A framework for scheduling and managing High-Performance Computing (HPC) applications in science cloud,” Future Generation Computer Systems, 34, pp.47-65, 2014.