Energy Optimization in Fog Computing to Improve Quality of Service

MSc Research Project
Cloud Computing

Abimbola Ogungbe
Student ID: 19124813

School of Computing
National College of Ireland

Supervisor: Vikhas Sahni

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Abimbola Ogungbe
Student ID:	19124813
Programme:	Cloud Computing
Year:	2019
Module:	MSc Research Project
Supervisor:	Vikhas Sahni
Submission Due Date:	17/08/2020
Project Title:	Energy Optimization in Fog Computing to Improve Quality of Service
Word Count:	6047
Page Count:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

I agree to an electronic copy of my thesis being made publicly available on TRAP the National College of Ireland's Institutional Repository for consultation.

Signature:	
Date:	16th August 2020

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Energy Optimization in Fog Computing to Improve Quality of Service

Abimbola Ogungbe
19124813

Abstract

Fog computing is an extension of cloud that brings the cloud closer to where data is created and acted upon. Fog computing is introduced so as to reduce the disadvantages arising from cloud computing. The key aim of Fog computing is to provide the devices with an optimum service level which is fast and effective. However, Fog is a new emerging computing paradigm which still needs standardization, particularly in the aspect energy optimization problems. To study the problem of energy consumption in fog computing, we propose the use of an energy aware load balancing algorithm. An energy aware load balancing algorithm is developed in this paper in order to allocate tasks to fog devices in an energy aware manner. Each allocation considers the energy capacity of the fog devices. This research work is successful in that it is able to reduce the energy that is consumed in a fog device. Introducing the use of renewable source of energy, the power consumption in fog/IoT environment is also reduced. Finally, based on simulations and numerical results, we showed the effectiveness of the proposed algorithm and its performance over default algorithm in the simulator.

1 Introduction

In recent years, Cloud computing is seen to be one of the fastest growing technologies in the world of internet and one of the valuable research fields. This growth in cloud computing as of today has greatly made individuals and organizations see the importance of using the cloud Buyya et al. (2017). Cloud computing offers various services such as: on demand self-service, scalability and so much more. In these services, the cloud and the end users are wide distance apart from each other. This distance causes delays and shortage of resources while executing a request.

Fog computing was introduced by CISCO as an extension of cloud computing. It creates a virtual platform that provides storage, communication (network) and many other task services in a fog environment to customers in close proximity. Fog computing acts as a link between the cloud and the end user and it brings cloud services and computing resources closer to end user devices.

The main purpose of fog computing is to reduce the data load on the cloud, increase efficiency and performance by handling the IoT's data, communication needs and speed up response to data request that are sent to the cloud. It reduces traffic on the cloud and is able to maintain load balancing which improves quality of service (QoS).

With advancement in Internet of Things (IoT), information being generated from different IoT devices within a period are summarized and forwarded to the cloud so as to save network bandwidth to a large extent. As a result, latency is being reduced but high energy is being consumed. The reason being that a lot of information is being sent to the fog server at the same time for processing Dastjerdi et al. (2016).

Load balancing technique is an important factor in fog computing. In this technique, loads of requests emanating from end user devices are distributed to the fog device. The loads are also scheduled in conformity with the fog device energy capacity. By distributing and scheduling the loads, reduced energy consumption is achieved

1.1 Motivation and Background

The increasing and exponential growth of IoT devices and industries needs for large data storage gave rise to cloud computing that has all accommodating storage capability and on demand data services. However, it is quite a distance for data to travel between cloud and the point where it is needed. For instance, there are over 60,000 requests per seconds sent to cloud alone from end users and this large request results in slower response time from cloud. This is why Fog computing was introduced to bring cloud capabilities closer to end users as related to IoT. However, with the increasing use of internet of things (IoT), there is so much usage of fog and this give rise to extreme energy consumption.

According to the survey conducted by Mukherjee et al. (2018),there are various research areas required to be solved in fog. One major area that research has not focused on is the energy consumption in fog. To process the data that is being generated and sent by the IoT device often consumes huge amount of energy. Fog computing reduces the amount of data to be sent directly to cloud and this in turn reduces traffic between the computing resources as well as improve the quality of service Buyya et al. (2017). However, it consumes high energy in performing the task. This high energy consumption challenge encountered with fog computing constitutes a major area for research. A major factor to be considered in fog before load scheduling is the energy consumption of the fog device.

1.2 Research Question

According to the research question "How can energy aware load balancing algorithm reduce energy consumption and aid load distribution in fog-IoT environment". High energy consumption is an obvious fundamental challenge encountered in running Internet of Things (IoT) applications in a Fog environment and this turn affects the overall performance of the service. The main objective of Fog computing is to provide the device with an optimal quality of service which is fast and efficient. However, with the increasing use of internet of things, there is so much usage of fog which now leads to extreme energy consumption.

1.3 Research Objectives and Contributions

Different researchers have performed different survey in fog computing, and they were able to open up issues that need to be addressed in fog computing. The major objective in this research is to design, develop and implement an energy aware load balancing algorithm that will solve the problem of energy consumption and aid load distribution. Objective 2: Creating a fog computing environment to perform the test of the implemented algorithm. Objective 3: Getting the results by assigning task and examining and analyzing results proving that energy aware load balancing algorithm is successful when it was tested. In addition to this, the paper is structured as follows. Chapter 2: presents the reason for proposing an energy aware load balancing algorithm Chapter 3: talks about the tools for implementation and algorithm chapter 4: explains the design of the research work Chapter 5: explains the implementation techniques Chapter 6: presents the test results in comparison with the default algorithm.

2 Related Work

Energy consumption in fog is the least research area in most cases Naha et al. (2018) and Hu et al. (2017). This research work focuses on the amount of energy consumed when performing a workload in fog -IoT environment during load scheduling.Plenty of research is being done on different load balancing technique in fog environment. This techniques has been able to solve the problem of service delay and response time. Scheduling and load balancing is seen as the key parameters for improving quality of services which is under review in the literature study. One of the major challenge of scheduling and load balancing is considering the energy of the computing resources (i.e cloud / fog environment) before allocating tasks. In this section, we discuss the existing work done in load balancing and energy optimization in fog to point out the drawbacks from the literature work and form the basis of our solution.

2.1 Fog Computing Survey

With the recent development in internet, there is increase in users' requirements which give rise to more and more data to be processed. This development has led to the emergence of cloud computing, fog computing and edge computing. This is why Zhiguo Qu and Li (2020)did a survey on the study on QoS optimization and energy saving in cloud, fog, Edge and IoT. Major problems of quality of service, resource allocation, server optimization and load balancing encountered are pointed out and analysis of different solutions that were proposed by existing researchers were made. Zhiguo Qu and Li (2020) arrived at the conclusion that this survey will enable readers have a deeper knowledge on the different computing models and on aspect of QoS as well as energy optimization.

Naha et al. (2018) and Hu et al. (2017) performed a comprehensive survey on fog computing in the aspect of Internet of Things. He explained the importance of fog computing when processing Internet of Things (IoT) request on the cloud. The survey also went ahead to talk about the characteristics of fog computing and compare the characteristics with cloud computing. Each paper came to the conclusion that fog computing reduces data transfer time and the amount of network bandwidth used. Naha et al. (2018) and Hu et al. (2017) threw more light on the challenges encountered when performing a request on a running IoT application in fog. However, one of the major challenges identified during this process was resource allocation and energy management.

Throwing more light into fog computing, Mahmud et al. (2018) analysed the challenges encountered in fogs when acting like the intermediate layer between the IoT devices and Cloud. The challenges discussed were in terms of resource architecture, service quality, security related issues, power management issue and so much on. However, the challenges identified were classified and mapped according to available existing works and the different approaches towards addressing the challenges. With this mapping, Mahmud et al. (2018) were able to identify current research gaps in fog computing and this aided them in proposing research direction that can be pursued in the future, such as load balancing and energy optimization.

Minh-Quang Tran and Pham (2019)) proposed a unique method to task placement on fog computing to improve performance of the IoT devices in terms of response time, energy and cost reduction. They did this by analysing and modelling IoT application. They introduced a systematic fog computing framework consisting of multiple intelligent tiers that support IoT service which resulted in efficient task placement approach for optimizing IoT application on fog computing. Their approach optimally made use of the fog devices that are available on the network edge for improving the performance of IoT services in terms of response time, energy, and cost reduction. However, they were able to conduct a performance analysis by implementing a program for optimization of task placement on the fog landscape considering the network performance.

2.2 Load Balancing in Fog Computing

The emergence of load balancing in fog computing gives rise to reduced energy consumption. The authors in Hayat et al. (2019) discussed an efficient way to solve energy consumption in terms of resource allocation. They were of the opinion that as the use of IoT devices increases, the usage of cloud increases. This increment makes the Fog environment more active which in turn leads to more energy consumption. Hayat et al. (2019) introduces a load balancing algorithm that will be used to balance loads of requests across different servers. The load balancing algorithm was able to distribute load to different fog node based on available virtual machine (VM) checking which VM has low latency ratio so as to save time. Saving time implies reduction in energy consumption. They also gave a brief insight into how energy is consumed. Energy consumption occurs in three forms namely; when request is gotten from the end user to the fog, when the requests is being processed by the fog and when the request is sent out to the cloud for resource. Although their solution is feasible but, they only focused on assigning requests to available VM's with low latency ratio without considering the capacity of the VM's. They also focused on solving the energy consumed when a request is being processed by the fog rather than solving the problem of energy consumption in all the three forms.

Xiaolong Xu and Liu (2018) proposed a dynamic resource allocation method (DRAM) in this paper for load balancing in a fog environment. They pointed out that analysing the computing nodes used is required for load balancing in a fog environment during the execution of IoT applications. With their algorithm, they achieved a load balancing analysis for various types of computing nodes required for load balancing in a fog and cloud environment.

Xuan-Qui Pham and Eui-Nam Huh (2016) proposed a heuristic based algorithm for task scheduling in a cloud-fog environment. This algorithm enabled the fog server to communicate with its fog nodes and the cloud so as to achieve effective and good workflow execution time for quality of service (QoS).They proposed a problem and solved the problem by scheduling task according to scheduling priorities as well as assigning the task according to a capable fog node that will execute the task on time. This algorithm was also used to reduce the need for obtaining cloud resources. After testing, there was a balance between performance of application execution and reduction of cost. However, the algorithm was done to guarantee that task execution is executed on time not considering the energy consumed during execution.

The authors in Khattak (2019)) introduced the use of foglets to handle requests that are sent to the cloud. The solution requires fewer requests to be sent to the cloud in order to achieve reduced delay as well as improve quality of service. This was achieved by even distribution of load on the fog layer which resulted in load balancing and effective utilization of resources. This utilization of the fog node was evaluated using an Ifogsim tool which aided in load balancing among the fog nodes.Khattak (2019) was able to achieve utilization of resources by getting tasks done evenly; making sure that there is no wastage of resources. All devices were in use to achieve even load distribution.

Luo et al. (2020) introduced three strategy in this paper for solving energy consumption and latency aware. Luo et al. (2020) first introduced a latency-aware energy-efficient continuous data-flow optimization strategy for a continuous flow of data in an IoT-Fog-Edge Computing framework. This strategy was tested on an E-health Monitoring system. Secondly, they introduced an anomaly detection strategy to evaluate the energy consumed and latency that occurs between the Fog nodes and IoT device. Lastly, they introduced a block coordinate descend based max-flow (BCDM) algorithm to monitor the performance of the E-health system by analysing the total workload. Based on this research, Luo et al. (2020) only considered the latency property of data flow service leaving the aspect of energy being consumed. However, both Khattak (2019) and Luo et al. (2020) did not consider the energy optimization in their proposed solution

2.3 Energy Optimization in Fog Computing

Liu et al. (2018) introduced a queuing method to study the energy consumption and delayed performance for a mobile fog computing system. These queuing models were applied to the IoT device, fog and cloud servers to minimize energy consumption and performance delay. In order to hold on to their obtained results gotten from using these queuing models, they introduced the use of an IPM - based algorithm by addressing the formulated issue of a multi-objective problem associated with different constraints. Furthermore, different simulation tests were also carried to test the effectiveness and performance of their proposed algorithm and all the results came out positive.

The authors in Fan et al. (2018) used an energy efficient and latency aware algorithm for reducing energy consumption in a geo-distributed cloud data centre taking cognisance of the latency factor and energy required to attain a specific output. The proposed Energy - Efficient Latency-Aware Data Deployment Algorithm (ELDD) was done in two forms. The first form is to determine the total number of data chunks and sort them according to the way they are accessed by users. The second form is to merge them into one large data segment and look for the appropriate server to accommodate the merged data segment so that energy consumption will be reduced when the data is processed at once. The algorithm was also designed to power off free servers that are not in use to reduce energy consumed when not in use. The algorithm proposed was effective in data centres based on their simulation results. However, the area that wasn't put into consideration was the energy consumed when merging the data into one large data segment.

Wan et al. (2018) introduced an energy aware load balancing and scheduling (ELBS) method to solve the problem of load balancing and energy consumption. They performed their experiment by testing their solution in a prototype candy smart factory where they made use of robots and raspberry pie. This was made possible through optimal scheduling and load balancing for the working robots. The requests that were sent through the raspberry pie are executed by different robots so as to save energy. However, the model proposed is only suitable in the context of manufacturing industry which is highly not feasible in the coming years.

In Mahmoud et al. (2018) proposed an efficient energy aware allocation algorithm based on task request to solve energy consumption. The algorithm allocates incoming tasks to different fog devices based on CPU capacity. If a current fog device cannot meet the task processing demands, the algorithm sends the task to another fog device that can accommodate the request. The main purpose of the algorithm was to improve energy efficiency in Fog Device. This was done by improving the round robin algorithm and dynamic voltage frequency scaling. The improved round robin algorithm achieved task allocation based on the CPU capabilities while the dynamic voltage frequency scaling algorithm was included to balance the load in the CPU when the task is allocated to it. However Wan et al. (2018) and Mahmoud et al. (2018) only solved the problem of energy consumption when a request is been processed only in a fog environment instead of in a Fog-IoT environment.

Toor et al. (2019) focused their direction of research on energy consumption and Quality of Service (QoS) as a key metrics in fog-cloud computing. They discussed about the fog computing sustainability in respect to the increasing demand of IoT devices. These IoT devices are later found to be consuming more energy since they are up and running in the cloud. Each of them sending different request to the fog that are later relayed into the cloud. They introduce the use of green energy to help reduce this power consumption by proposing an energy-aware scheme for Fog–IoT computational environment. The methodology utilized green energy for powering the IoT device while performing its workload on the fog node.

2.4 Fog Computing Techniques Used in Saving Energy

The authors in Al Faruque and Vatanparvar (2016) introduced the use of energy management as a service, as a way of controlling power consumption. They went ahead to say that using energy management as a service in fog computing is scalable and adaptable. They proposed two prototypes which include Home Energy Management and Micro grid level. After different experiment and tests were carried out, micro grid has proven to be the best for residential and commercial use. The micro grid consists of three different homes connected to a transformer. With this, they were able to monitor the power consumption and measure the load gotten from each home and the transformer current being used. On the other hand, Home Energy Management platform was used mainly in companies that convert an existing home to a smart home. With this, they were able to decrease the total energy consumption in this area by controlling the smart devices. What they failed to target is the power consumption used in the processing of this information to the cloud and the energy used in data transfer.

Mebrek et al. (2017) introduced the use of an efficient green solution to obtain a balanced energy consumption by proposing an Evolutionary Algorithm approach to investigate the power consumption and service delay in fog computing. Their main objective was to find out how energy can be saved and Quality of Service (QoS) improved in fog with respect to the increasing demand of IoT devices. The algorithm made use of an improved solution of genetic algorithm to solve the problem of energy optimization in the IoT device and improve Quality of Service (QoS). The authors concludes that the proposed algorithm was successful when tested.

Barros et al. (2019) paper suggested the use of fog computing to solve the problem of power requirement. This is achieved by getting rid of the high amount of data that is being passed to the cloud and thereby reducing response time. That is, instead for a response to take a longer time to get, fog computing helps reduce the response time. The time reduction in getting the response to request makes energy to be saved. They made this possible by introducing a scheduling algorithm and Grid-Lab simulator for application so as to reduce delay for users. The algorithm made use of PID calculator to calculate energy rate that is used and from the expected result, manages the energy consumed by reducing the time it takes for a request to be processed in the cloud.

In regards to the research by Jalali et al. (2016), it expatiates on how the use of fog computing reduces energy consumption in cloud computing. It compared the energy consumption of a service provided by centralized data centre in cloud and a nano data centre in fog. They proposed a flow based and time-based energy consumption algorithm that makes the nano data centre consume less energy. The flow-based energy consumption algorithm consists of equipment such as routers shared between users. The time-based algorithm monitors and reduces the time it takes for the equipment to provide access to the users. According to the experiment, they were able to see that if they have more equipment in-between users, reduced energy consumption will be achieved. This is why they came to the conclusion that deploying the algorithm into the nano data centres in fog computing will save energy. The fact being that the applications were implemented in different smart IOT devices (equipment) which resulted to less energy consumption .The only thing they did was to run some application on nano data centre platform so as to save energy consumption that occurs in the centralized data centre.

In Deng et al. (2016), the author introduced a mathematical framework to investigate the trade-off between power consumption and delay in a cloud and fog computing system. They compared the power consumption used in a cloud and fog environment during workload allocation and used the proposed framework to investigate the energy consumed and service delay. However, they were able to solve this by dividing their formulated workload allocation problem (PP) into three sub problems to achieve an optimal workload allocation between the fog and the cloud. Based on the simulation and numerical results gotten, it confirmed that fog computing consumes less power when the allocation of workload is introduced.

2.5 Discussion

Each techniques used in the literature review are different from each other and they all have their pros and cons in terms of solving energy consumption and improving quality of service(QoS) in fog.

Based on the reviewed literature, we developed a table shown in Table 1 to summarize the relevant algorithm according to their proposed year. This table is aimed at providing a quick recap on each algorithm reviewed in our literature alongside their core idea for development, advantages and limitations.

Many algorithms have been proposed by many authors in fog computing. These algorithms are useful in Achieving low latency, and reduce response time in aspect of energy saving. This is why our proposed algorithm is keenly focused on reducing the consumption of energy in fog and analysing the different energy capacity of a fog device before load scheduling. In this paper, a new energy aware load balancing algorithm is implemented in Fog Computing to improve Quality of Service. The proposed approach is aimed at not only reducing energy consumption but also in aiding load distribution.

Year	Algorithm	Core Idea	Limitation	Authors
2016	Heuristic Based Algorithm	Scheduling Tasks according to priorities	They scheduled task based on priorities not considering the energy consumed during execution.	Xuan-Qui Pham and Eui-Nam Huh (2016)
2016	Mathematical Framework to investigate the trade-off between power consumption and delay in cloud-fog environment	Divided their primary problem into three sub problems to achieve optimal workload allocation	They only observed the power consumption in a cloud-fog environment. They did not offer solutions on how to solve the problem of power consumption but only solved the problem of delay	Deng et al. (2016)
2018	Energy Aware Allocation Algorithm	Allocates tasks based on CPU capacity	they considered energy consumed after task allocation	Mahmoud et al. (2018)
2019	Load Balancing Algorithm	Load distribution to available VM with low latency ratio	Did not Consider the VMs capacity	Hayat et al. (2019).

Table 1: Summary of Some Reviewed Algorithm

3 Methodology

With the regular advancement of cloud technologies, developers are faced with more pressure to handle the problem of energy consumption, load distribution, performance and configuration of application. A lot of cloud providers are offering pay-as -you-go services with flexible infrastructure. This has become a concern in selecting the best cloud tool for most individual. After reviewing various cloud simulation platforms like cloudsim, cloud analyst and so much more, we observed that ifogsim provides the ability for us to add our customized algorithms for load balancing and virtual machine configurations. Also, ifogsim is being used for most of the research projects currently under study Mahmoud et al. (2018),Mebrek et al. (2017), Minh-Quang Tran and Pham (2019), Toor et al. (2019) and Buyya and Srirama (2019).

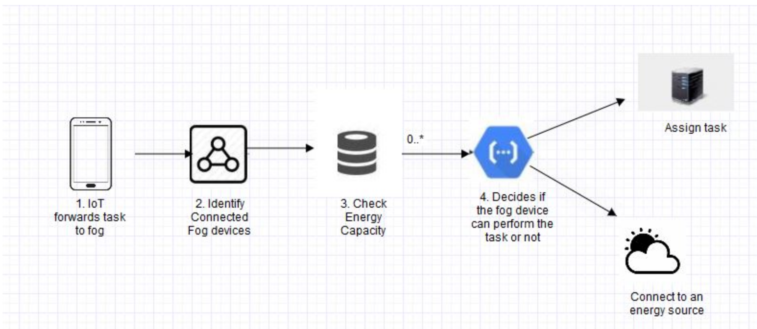

Figure 1: Methodology Overview

3.1 iFogSim

iFogSim is a tool used to simulate the proposed algorithm by using creating virtual environment. It is a basic simulation tool found in cloudsim which allows them to communicate between each other. It is used to model and simulate the network of internet of things, cloud and fog computing. The architecture of ifogsim was developed with simulation functionalities of Cloud Sim, Buyya and Srirama (2019). iFogSim has five major classes, i.e., fog-device, sensor, tuple, actuator, and application. In addition to that, the simulated services available in iFogSim are power monitoring service and resource management service. iFogSim supports simulations on the scale expected in the context of IoT.

The key components of the iFogSim required for the proposed work are:

- Fog Device: It consists of accessible memory, processor, storage, up-link and down-link bandwidths which represents the communication capacity of fog devices
- Sensor: Its acts as IoT sensors which explains the output characteristics of the sensor and identifies the tuple arrival rate.
- Actuator: It explains a method to perform an action on arrival of a tuple from the application class
- Tuple: This acts as a core unit of communication between entities in the fog
- Application:This determines how the application modules are placed or scheduled in the fog device.

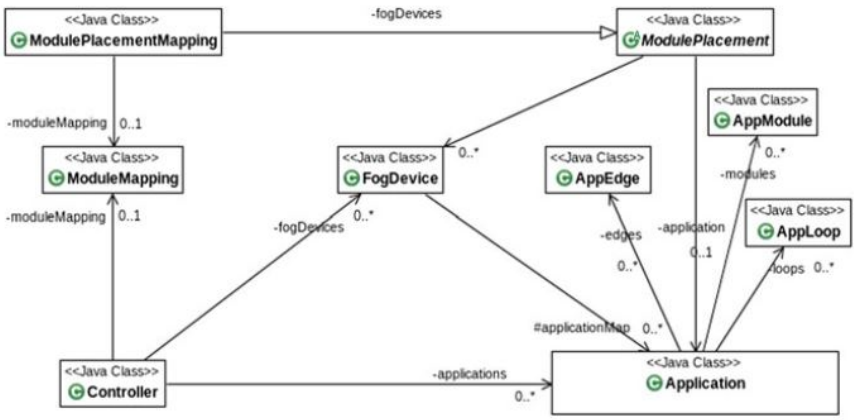

Figure 2: Classes of iFogSim Buyya and Srirama (2019)

3.2 Eclipse IDE

The iFogSim was executed in a java based integrated development environment (Java IDE). IFogSim supports Java, and thus this is our default programming language for developing the algorithms

3.3 Selection of Evaluation Parameters

For the purpose of this Research, the following parameters were selected for the evaluation of the system. The target of the algorithm is to determine the current Energy level of the fog nodes, calculate the Energy Source and calculate Energy consumed.

Selection of evaluation parameters: The parameters for the evaluation of the system were picked such that the performance of the algorithm can be noted clearly. The following parameters were selected.

- Energy Level - this is the percentage of the current Energy Level of the Fog devices. This helps to determine which fog is ready to accept and process tasks
- Energy Source. Based on the current energy level, the algorithm should be able to determine the current source of Energy being used by the Fog. This could be Renewable or Non-Renewable Energy.
- Energy Consumed. This refers to the total amount of energy consumed during the process life-cycle.

3.3.1 Implemented Algorithm

Algorithm 1 Energy-aware load balancing algorithm

```
Mode = Full Power (FP)
while energy <= 40% do
   load schedule == device power + sourced energy
   Step 1: procedure EnergyManagement(battery, weather)
   Step 2: Input Battery Status (Integer value between 1 - 100)
   Step 3: Input Weather Status (sunny or cloudy)
   Input : fogDevicesList, IOTs
   Output : LoadBalancing in an energy-aware manner
   Fog devices (Fog servers) implement LoadBalancing to distributed Request across
   Nodes
   allocatedDevice = NULL
   for fogDevice in fogDevicesList do
      for module in modulesToPlaceList do
         estimateConsumedEnergyBeforeAllocation(fogDevices, module)
         if fogDevice is suitable for Request then
            allocatedDevice = fogDevice
         else if fogDevice is not suitable then
            search for fogDevice upwards && trigger Renewable Energy Source
         end if
         place module on allocatedDevice
      end for
   end for
end while
```

The algorithm aims at ensuring load balancing is achieved in an energy-aware manner. The research algorithm makes use of the energy- aware load balancing algorithm to push tasks to fogs with adequate Energy capacity. The distribution of loads ensures that each fog is not overloaded. At the start of the simulation, the energy level of each fog is retrieved. The Energy level of each fog nodes also determines each's energy source. The two energy sources are Renewable and Non-renewable energy. At the end of the simulation, the total energy consumed is also retrieved.

4 Design Specification

A major challenge in Load balancing is the ability of the algorithm to efficiently schedule tasks to fog servers in a manner that enables the entire Network system to have a Highly Significant Quality of Service. In order to model a Load Balancing Algorithm that ensures Improved Quality of Service, we created an algorithm which improves on the one created by Mahmoud et al. (2018) While their Research focused on Energy state after allocation of task, this research retrieves Energy level of each fog node before allocating task. The following diagram shows the flowchart for the algorithm used is shown in the figure below

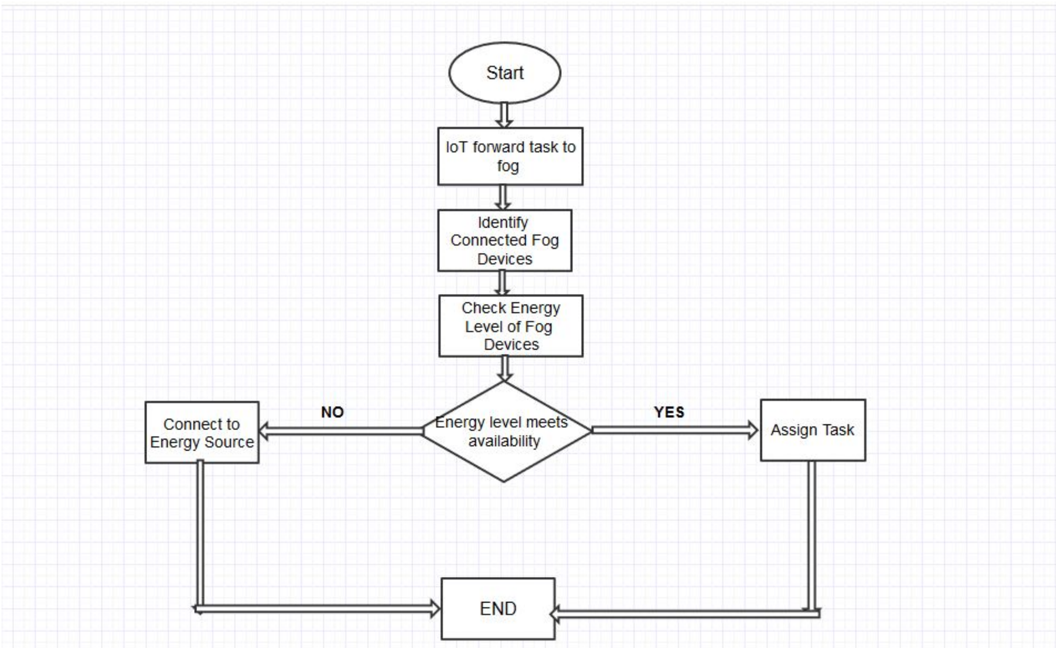

Figure 3: Flowchart for Model Design

Figure 2 shows the proposed method. In this model, different tasks are sent to different fog nodes for processing. This is why our proposed algorithm is introduced to ensure that load schedule on the fog nodes are determined by the energy capacity at the time the tasks are been assigned. The algorithm initially first detects the energy level of all fog nodes and then determines which fog is ready to process the tasks. This methods occurs repeatedly to enable other pending tasks to be processed. This model does not only assign task to fogs, it also ensures that the energy source of each fog is determined based on its current battery level.

4.1 Implemented Network Architecture

The network architecture describes the interaction mode between all gadgets involved. The devices rely on both renewable and non-renewable energy source. Each option is based on the current energy level of the fog server. The Network Architecture is as shown in the figure below.

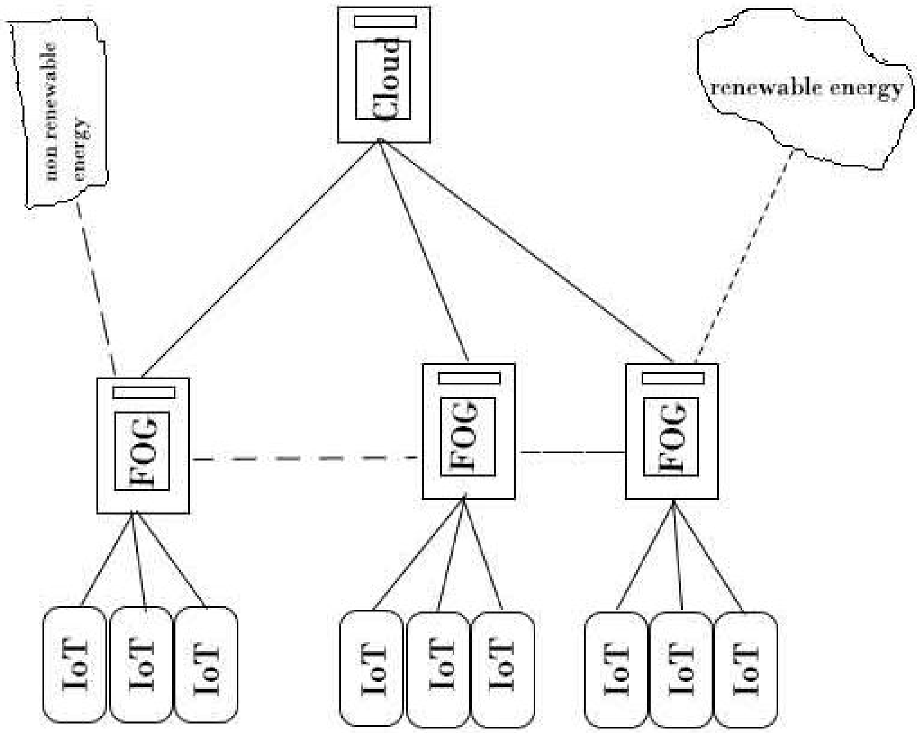

Figure 4: Network Architecture for Proposed Model

4.2 Configuration

The proposed Architecture targets a consistent distribution of tasks from IoT devices. Energy has been a major factor in the efficient service quality across fogs. The fog nodes rely on either renewable or non-renewable energy during task processing. The configuration for the cloud and fog Servers are tabulated below.

Parameters	**Specifications**
System Architecture	x64
Operating System	Linux
Virtual Machine	Xen

Table 2: Cloud Configuration

Parameters	Specifications
RAM	4000 MB
Storage	1000000 GB
MIIPS	2000
Bandwidth	10000

Table 3: Fog Configuration

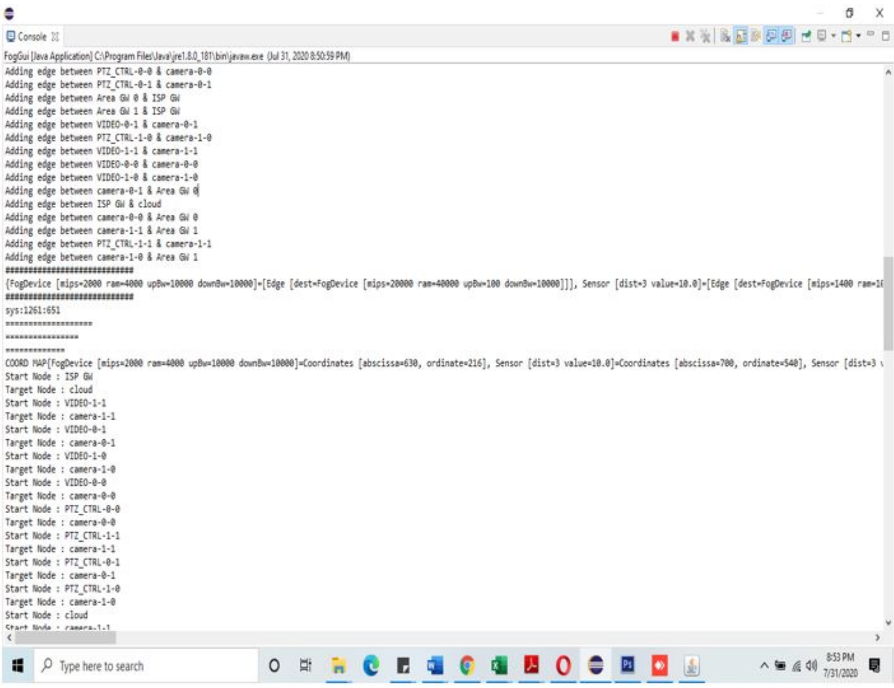

Figure 5: Output Showing the Parameters

5 Implementation

Due to the cost of setting up a live scenario of a cloud computing architecture, the research adopted IfogSim for the design of the Network Architecture. The IfogSim Simulator is suitable for testing different Algorithms ranging from Load Balancing, Virtual machine, task scheduling and lots more. The advantage of using a Simulator is the fact that it can detect anomalies before live deployment of the implementation. The Simulation Test was done on a Computer running on a 64-bit Windows 10 platform with Intel i7 core processor. Since the IFogSim was built with Java, the Algorithm implementation was programmed with Java with the use of Eclipse IDE.

The Java Class for the main implementation is:

- MyFogTest.Java. This class instantiates the Host, Log, Pe, Storage, PowerHost, RamProvisionerSimple class of the Cloudbus package. MyFogtest.java is the main application which starts the simulation. The Simulation starts with the main class invoking methods that create the Network Infrastructures. The Algorithm then assigns task based on Power level.

  This starts the simulated environment and calls the Host, Log, Pe, Storage, PowerHost, RamProvisionerSimple class to start

- The Host class executes the Virtual Machine related actions.
- The Log class performs logging of the simulation.

5.1 Load Balancing Class

In IfogSim, the Load Scheduling is managed by the DataCenter Controller class. This class is found in the CloudSim package. It is a CloudResource that deals with processing of Virtual Machine queries. The DataCentre class was set to process all queries sent by the VM. The VmAllocationPolicy class was set to effectively process and allocate the VMs.

5.2 Power Management Class

A java class named PowerState.java was created to check the Energy Level of the fog devices. The java class contains a static method showPowerLevel(). This static method returns the current energy level of the fog nodes. It also returns the Energy source of each fog. Since showPowerLevel method in class powerState.java is a static method, it is called from MyFogTest class using its own class name. The PowerHost class contained in the CloudSim package calculates the Energy consumption upon simulation of the Load Balancing Algorithm.

5.3 Fog Environment

This section explains about setting up fog environment in an iFogSim.It involves communication between the fog layer,cloud layer and sensor devices. The cloud layer is the cloud data center to provide the required infrastructure and computational resources.

6 Evaluation

To ascertain the efficiency of the proposed energy aware Load balancing algorithm, series of tests are conducted using ifogsim simulation toolkit and the results are also compared against the default algorithms (DCNS) in the Simulation tool.

The Network Architecture designed as case study for executing the load balancing technique in this research is a case of a surveillance camera used for security monitoring. The parameters used for evaluation are.

- Initial Fog Energy: This is the Energy level of the fog before task is assigned.
- Energy Consumed: This refers to the energy consumed during task processing
- Time taken: This refers to the time taken for the completion of the task processing
- Network Cost: This is the network cost of the task processing.
- Waiting Time: This refers to time take taken while a task is waiting for another task to be completed.

6.1 Experiment 1

In this experiment, we are performing 4 number of iteration to determine the average energy consumed during the implementation of our algorithm. This experiment was carried out by starting simulated environment and creating task model for the fog devices for processing. Then the next stage is putting our algorithm to work to check the defined parameters (i.e initial fog energy level) before task allocation. Results from this test are represented below (Figure 5). From the results, we can observe that the average energy consumed during task processing is not as high as the energy that is consumed when using the default algorithm in the simulator. An illustration of the Energy consumption comparison was shown. The chart shows the efficiency level of the proposed algorithm as it was evident that the proposed algorithm gave a better quality of service (QOS).

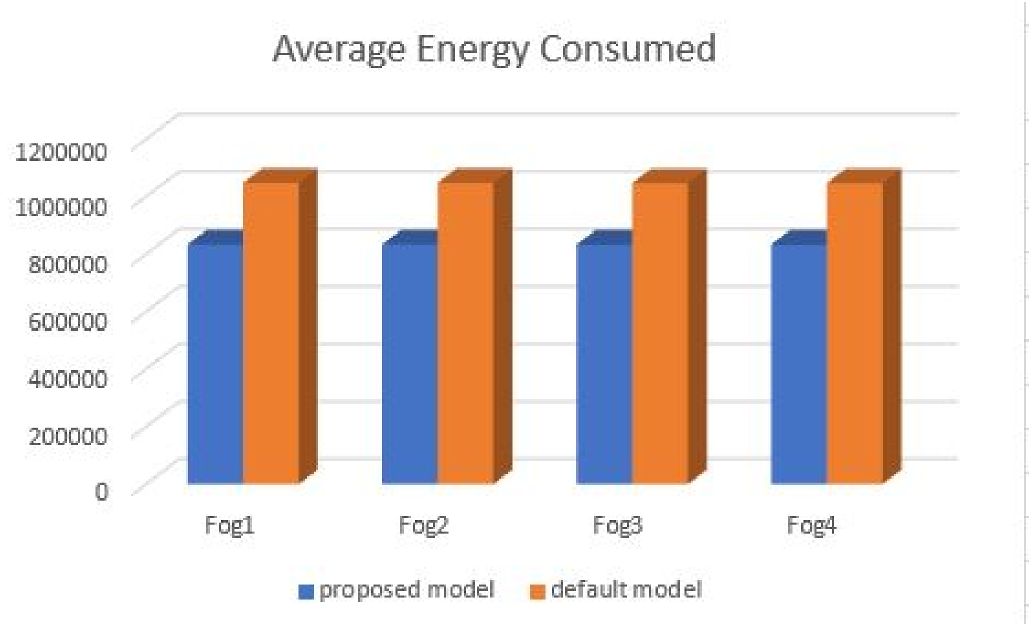

Figure 6: Average Energy Consumed

6.2 Experiment 2

In this experiment. we are comparing the time taken to complete the task using the proposed Algorithm against the time taken in the default algorithm in the simulator. We observed that our technique exceeds and provides improved result as compared to the default techniques. As would be observed from the chart (Figure 6), there is a reduction time of nearly 31% of the total completion time using our proposed algorithm.

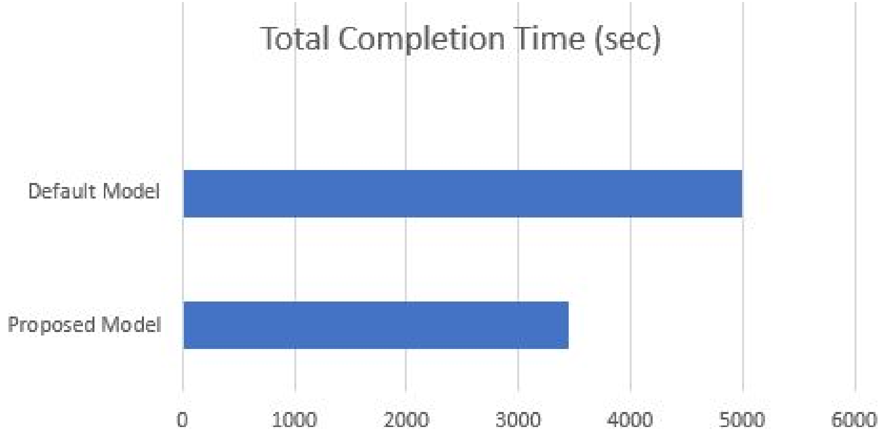

Figure 7: Completion Time of job for each Fogs using Default algorithm and Proposed Algorithm

6.3 Experiment 3

In this experiment, a quantitative evaluation of the average waiting time of tasks for the two algorithms are plotted. It is easy to predict from the plot that the proposed algorithm is performing better than the default algorithms in the simulator. From the results below (figure 7),this means that the quality of service delivered to end user is better with our implemented model.

Figure 8: Average waiting time of the default algorithm and Proposed Algorithm

6.4 Experiment 4

In this experiment, we are observing the total cost of execution. The total cost of execution is the a key aspect that both the cloud provider and cloud user will look at while executing the tasks on the resources. Hence, a plot between the proposed algorithm and the default algorithm (DCNS) is drawn for all fog devices selected. It is evident from figure 8 that the total cost of execution is better for the proposed algorithm when compared to the default algorithm of the simulator. It obvious that there is a drastic reduction in the cost of execution which gives cloud user to have a greater advantage on cost savings.

Figure 9: Total cost of execution of Default algorithm and Proposed Algorithm

6.5 Discussion

From the above experiments, we can conclude that the energy aware load balancing algorithm in fog computing is successful in scheduling of task based on the fog device energy capacity. It even proves that our proposed algorithm ensures that the fog devices is not overloaded with tasks and energy source of each fog is determine based on the current energy level. This method reduced the energy consumed by the fog device as shown in figure 5. Figure 6 shows the comparison result of the total completion time. The average waiting time result was shown in figure 7.Figure 8 shows us the comparison of the cost of execution while processing the data.

However,One issue is that the simulation was repeated on the same number of fog devices to get the average value. Energy-aware load balancing algorithm will be successful in achieving greater results in real time environment. It will also achieve more positive results in using Internet of Things.

The output of the simulation is tabulated against the default Load balancing Algorithm in IfogSim. A tabular representation of the outputs is shown below.

	Proposed Model				Default Model			
	Fog1	Fog2	Fog3	Fog4	Fog1	Fog2	Fog3	Fog4
Average energy consumed	834332.7	834301.7	834100.7	834121.7	1048835.4	1048835.4	1048135.4	1048025.4
Average Waiting time	2184	2182	2188	2180	4301	4323	4300	4310
Cost of execution	806977.6	803977.6	806922.6	816977.6	840045.5	840045.5	820045.5	830045.5
TIME TAKEN	3450				5000			

7 Conclusion and Future Work

With the rapid demand for services and application in IoT devices, fog computing is emerging as one of the most sort after computing infrastructure that helps bring the cloud to end devices. However, In a fog environment there are some major challenges encountered in fog computing such as the energy consumed in a fog device. To reduce this, we carried out research to achieve Energy Optimization for improving Quality of Service. One of the ways to evaluate the reliability of fog computing is by examining the load scheduling approach being used in such network. To achieve this, we proposed an Energy aware load balancing algorithm on the assigning of tasks. In comparison with the default algorithm, the result showed that the proposed Load Balancing policy achieved more Energy optimization in Fog devices.The implemented research algorithm achieved the aim of reducing the use of power energy, energy consumed during task processing and aided load distribution to avoid the fogs from being overloaded. The simulation was done using iFogSim software where the algorithm was implemented and tested.

As regards to limitation, the algorithm was tested repeatedly with four different fog devices and on a simulation tool. This research project can be experiment on a real life environment. The future work for this research involves further improvements and a performance test of the proposed method under different scenarios.

References

Al Faruque, M. A. and Vatanparvar, K. (2016). Energy management-as-a-service over fog computing platform, *IEEE Internet of Things Journal* **3**(2): 161–169. Journal Impact Factor:5.863.

Barros, E., Filho, D., Batista, B., Kuehne, B. and Peixoto, M. (2019). Fog computing model to orchestrate the consumption and production of energy in microgrids, *Sensors* **19**(11). Journal Impact Factor: 2.475.

Buyya, R. and Srirama, S. N. (2019). *Modeling and Simulation of Fog and Edge Computing Environments Using iFogSim Toolkit*, pp. 433–465. ISBN = 978-1-119-52498-4.

Buyya, R. et al. (2017). A manifesto for future generation cloud computing: Research directions for the next decade, *ACM Computing Surveys* **abs/1711.09123**(5). Journal Impact Factor:5.550.

Dastjerdi, A., Gupta, H., Calheiros, R., Ghosh, S. and Buyya, R. (2016). Chapter 4 - fog computing: principles, architectures, and applications, *in* R. Buyya and A. V. Dastjerdi (eds), *Internet of Things*, Morgan Kaufmann, pp. 61 – 75. ISBN 978-0-12-805395-9.

Deng, R., Lu, R., Lai, C., Luan, T. H. and Liang, H. (2016). Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption, *IEEE Internet of Things Journal* **3**(6): 1171–1181. Journal Impact Factor:5.863.

Fan, Y., Chen, J., Wang, L. and Cao, Z. (2018). Energy-efficient and latency-aware data placement for geo-distributed cloud data centers, *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST* **210**: 465–474. cited By 1 source: Scopus as on 23/03/2020.

Hayat, B., Nauman, M., Yousaf, S., Mehmood, M. and Saleem, H. (2019). Efficient energy utilization in cloud fog environment, *International Journal of Advanced Computer Science and Applications* **10**. Journal Impact Factor:1.324.

Hu, P., Dhelim, S., Ning, H. and Qiu, T. (2017). Survey on fog computing: architecture, key technologies, applications and open issues, *Journal of Network and Computer Applications* **98**: 27 – 42. Journal Impact Factor: 3.991.
URL: *http://www.sciencedirect.com/science/article/pii/S1084804517302953*

Jalali, F., Hinton, K., Ayre, R., Alpcan, T. and Tucker, R. S. (2016). Fog computing may help to save energy in cloud computing, *IEEE Journal on Selected Areas in Communications* **34**(5): 1728–1739. Journal Impact Factor: 7.172.

Khattak, H., A. H. I. S. e. a. (2019). Utilization and load balancing in fog servers for health applications., *EURASIP Journal on Wireless Communications and Networking* **2019**. Journal Impact Factor:2.407.

Liu, L., Chang, Z., Guo, X., Mao, S. and Ristaniemi, T. (2018). Multiobjective optimization for computation offloading in fog computing, *IEEE Internet of Things Journal* **5**(1): 283–294. Journal Impact Factor: 5.863.

Luo, Y., Li, W. and Qiu, S. (2020). Anomaly detection based latency-aware energy consumption optimization for iot data-flow services, *Sensors* **20**(1). Journal Impact Factor: 2.475.

Mahmoud, M. M., Rodrigues, J. J., Saleem, K., Al-Muhtadi, J., Kumar, N. and Korotaev, V. (2018). Towards energy-aware fog-enabled cloud of things for healthcare, *Computers Electrical Engineering* **67**: 58 – 69. Journal Impact Factor: 1.747.
URL: *http://www.sciencedirect.com/science/article/pii/S0045790618300399*

Mahmud, R., Kotagiri, R. and Buyya, R. (2018). *Fog Computing: A Taxonomy, Survey and Future Directions*, Springer Singapore, Singapore, pp. 103–130. ISBN 978-981-10-5861-5.
URL: *https://doi.org/10.1007/978-981-10-5861-5*$_5$

Mebrek, A., Merghem-Boulahia, L. and Esseghir, M. (2017). 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, Cambridge, MA, USA, pp. 1–4. CORE Ranking: A.

Minh-Quang Tran, Duy Tai Nguyen, V. A. L. D. H. N. and Pham, T. V. (2019). Task placement on fog computing made efficient for iot application provision, *Wireless Communications and Mobile Computing* **2019**. Journal Impact Factor:0.869.

Mukherjee, M., Shu, L. and Wang, D. (2018). Survey of fog computing: Fundamental, network applications, and research challenges, *IEEE Communications Surveys Tutorials* **20**(3): 1826–1857. Journal Impact Factor: 20.230.

Naha, R. K., Garg, S., Georgakopoulos, D., Jayaraman, P. P., Gao, L., Xiang, Y. and Ranjan, R. (2018). Fog computing: Survey of trends, architectures, requirements, and research directions, *IEEE Access* **6**: 47980–48009. Journal Impact Factor: 3.557.

Toor, A., ul Islam, S., Sohail, N., Akhunzada, A., Boudjadar, J., Khattak, H. A., Din, I. U. and Rodrigues, J. J. (2019). Energy and performance aware fog computing: A case of dvfs and green renewable energy, *Future Generation Computer Systems* **101**: 1112 – 1121. Journal Impact Factor: 4.639.
URL: *http://www.sciencedirect.com/science/article/pii/S0167739X19310234*

Wan, J., Chen, B., Wang, S., Xia, M., Li, D. and Liu, C. (2018). Fog computing for energy-aware load balancing and scheduling in smart factory, *IEEE Transactions on Industrial Informatics* **14**(10): 4548–4556. Journal Impact Factor: 5.430.

Xiaolong Xu, Shucun Fu, Q. C. W. T. W. L. W. D. X. S. and Liu, A. X. (2018). Dynamic resource allocation for load balancing in fog environment, *Wireless Communications and Mobile Computing* **2018**. Journal Impact Factor:0.869.

Xuan-Qui Pham and Eui-Nam Huh (2016). Towards task scheduling in a cloud-fog computing system, *2016 18th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pp. 1–4. CORE Ranking: C.

Zhiguo Qu, Yilin Wang, L. S. D. P. and Li, Z. (2020). Study qos optimization and energy saving techniques in cloud, fog, edge, and iot, *Complexity* **2020**: 1076–2787. Journal Impact Factor:1.829.