A Framework for Modelling and Simulation of Data Processing with Fog Computing in Internet of Things Infrastructure

Ahmed Ali[1,2], Sultan Ahmad[1*] and Ahmed Ghneimat[1]

[1]Department of Computer Science, College of Computer Engineering and Sciences,
Prince Sattam Bin Abdulaziz University, Al-Kharj, 11942, Saudi Arabia.
[2]Higher Future Institute for Specialized Technological Studies, Cairo, Egypt.

[*]Corresponding Author: Sultan Ahmad

Summary
FOG Computing is an alternative to meet the latency and processing requirements that the Internet of Things (IoT) is facing. However, even having the necessary processing capacity, the proper provisioning and management of the data plays a fundamental role to ensure the maximum use of FOG computing at the same time to minimize the traffic on network and the routes. A mathematical model is proposed for the load balancing of IoT devices to FOG using GAMS and a simulator made in Python to measure and check the behavior of the optimized architecture over time. A significantly lower loss of information was achieved when using this proposed model to address the issues with GAMS in comparison with normal and random distribution model.

Key words:
Fog Computing, Edge Computing, Big Data, IoT, Cloud Computing.

1. Introduction

Cloud computing with its "pay what you consume" model allows today's large companies in the world to manage their data without paying private data centers and huge costs. Its scalability allows to extend the services without degrading performance. However, applications that require low latency and devices in their vicinity to meet their requirements cannot rely on the Cloud, due to its centralization model, low latencies are expensive (Bonomi, Milito, Zhu, & Addepalli, 2012)[1]. Statista predicted IoT devices worldwide to be around 30 billion by 2020 with further predictions to cross 75 billion by 2025 [2] that can be shown in Figure.1. Most of the devices with applications for home automation, energy saving, elderly care, education, localization among countless other functions. The true value of the data generated by these devices occurs when they are connected to FOG computing, allowing the combination of information from various sources [3].

Knowing the connection architecture of the IoT with FOG as well as the capacity and requirements of all the devices, it is possible to perform an optimization to ensure that all needs are satisfied and that all devices are used as much as possible. All this, ensuring that possible unforeseen connections do not collapse communication. For all the above, this project focuses on maximizing the use of communication channels between IoT devices and FOG computing, ensuring that the load of each of these is balance. This, in order to reduce the amount of information lost in data transfer.

The proposed solution consists of mathematical modeling a generalized network architecture in GAMS, maximizing the load balance in the networks and obtaining the percentage of information to be sent from each IoT to each FOG node in response. With this, the architecture in which each IoT randomly generated data streams was simulated. This simulation is compared with another whose load balancing was random. The use and traffic load of the network and the information lost was significantly lower when using the simulation result for the direction of information in the network.

This document consists primarily of a general description, the objectives, the state of the art and the importance of the project. Next, the problem to be solved with its specification and restrictions is defined. After this, the design process of the solution to the problem and the implementation of the chosen alternative with its analysis are discussed. Then the results obtained are validated and finally the work carried out is concluded. For a better understanding of the document, it is recommended to have knowledge of network architecture or at least a general context in FOG and Internet of Things (IoT)[4][5].

2. Problem Statement

The Internet of Things (IoT) has revolutionized computing by allowing millions of devices to connect to each other

Manuscript received November 5, 2020
Manuscript revised November 20, 2020
https://doi.org/10.22937/IJCSNS.2020.20.11.12

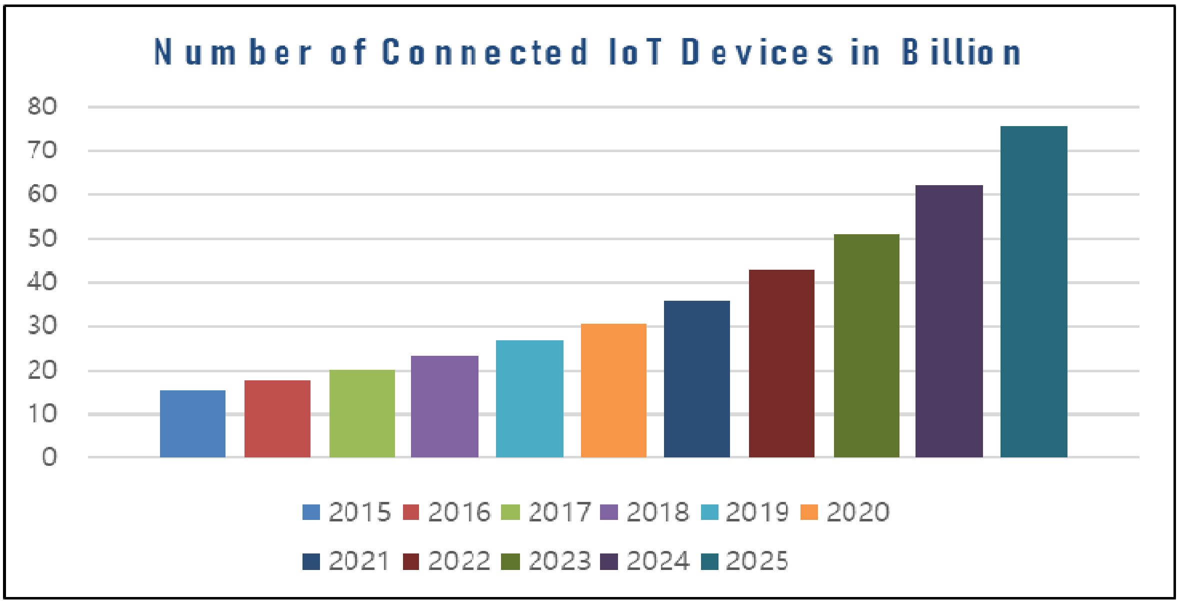

Fig. 1 No. of connected Devices worldwide in IoT (Source - Statista ,2019).

predominantly through wireless networks. This results a consequence as a high amount of data that must be processed and led through a varied group of devices towards Cloud computing, where data is securely stored[6]. Therefore, network traffic can become congested with the amount of data sent. Hence the idea of Fog computing was born. “Fog Computing is a highly virtualized platform that provides computing, storage and networking services between edge devices and traditional Cloud Computing data centers, typically, but not exclusively, located at the edge of the network” (Bonomi et al., 2012). This is how traffic to the cloud is reduced since the data is sent to several fog devices, and then it is sent to the centralized Cloud. However, ensuring quality of services such as the capacity, memory and delay time required for data transmission from IoT devices to Fog is a difficult task and this problem should be addressed, as it may cause data loss.

Due to the advancement in technologies with the time and the development of tools that allow the integration of most devices through the internet (Internet of Things), the amount of data transmission is increasing every day [7]. For this reason, the objective of this project is to optimize the load balancing of data transmission between IoT and Fog computing devices, focusing on the maximum use of existing communication channels. The other major objectives are designing the mathematical model according to the problem, optimizing the load of the network, simulating and checking the effectiveness of the optimization and its impact.

3. Related Works

The brand new Computing paradigms might achievable to meet QoS requirements, application synchronization and revelation. Some related works have been done which are very much related to this problem. The work in an article by Hamid Reza et.al presented MIST, that is scheme based data analysis approach for Fog with optimization of resource provisioning[8]. Their work can be used for the purposes of detecting the devices of IoT. In this work an optimization of the resources is carried out which can respond to the problem in question, but not the boat from the load balance of the data shared between IoT and Fog.

On the other hand, Yan Sun and Nan Zhang propose a structure for Fog computing and presented an algorithm that allows optimizing the random integration of spare resources in the network [9]. However, this work helps to improve network performance from an integration point of view. In an another work, the authors have explored the effects of Fog Computing and other computing technologies on the growing Big Data, Data Mining and its analytics techniques, especially in integrated Cloud-Fog IoT infrastructure [10].

Regarding integration of Cloud-Fog IoT, many other works have been done. Manuel et.al in their paper discussed and explored State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing [11]. The current models and environments of computing have changed so dramatically following the unexpected fast improvements in technologies. The new paradigm shift from Large Data centers to small centers of data everywhere is being famous and adoptable based on cyberspace supporting as cloud computing, Fog Computing, Edge Computing and Internet-of-Things (IoT), and other large-scale computing environments [12][13].

4. Design and Specification

The flow and processing requirements of every IoT devices must be satisfied in their initial state. It is possible that in the simulation the architecture does not meet this requirement since IoT devices have a variable data generation rate and FOG have a fixed memory and processing capacity[14]. In this situation the acceptable solution is the one with the least losses. There is the possibility that the architecture is robust and, regardless of the direction of the information, it is capable of executing the task. However, this case will not be taken into account since in real life resources are limited.

Taking into account the problem presented above, we want to design a mathematical model with which the load balancing of the data flow between the IoT and Fog devices can be optimized and can maximum use the bandwidth. For optimization, the main restriction is the license required by the different optimization tools. Xpress and Gams were taken into account as options. However, Gams was used given that the necessary license for the optimization of real-type cases was accessed, that is, with a number of IoT devices greater than 1000.

On the other hand, there was a restriction regarding the tool used for validation through simulation. A tool was needed to allow different parameters to be set to the devices, such as storage and processing capacity for Fog devices and number of IoT devices. Additionally, it was necessary for the tool to manage the use of each connection. Taking into account these restrictions, tests were carried out with various simulators such as NS3, iFogSim [15] and Opnet, however, the simulators did not meet the requirements, so a custom-made simulator was finally used for this work.

For the design an iterative process was carried out, having a model, it was reviewed and the elements that were not in accordance with the needs of the problem were changed. First, a model was designed that complied with applied Kirchhoff's laws and took the architecture as a graph. However, the use of these laws was incorrect since the problem cannot be represented with a standard graph. On the other hand, another model was created that took the restrictions into account, but the objective function was not in accordance with the needs of the problem, it did not represent the maximum use of the connection and bandwidth. Finally, a mathematical model was built that took the correct elements from the previous models and was improved by adding the restriction of the size of flows.

First we defined different sets of devices. Like it is defined N : No. of IoT devices, M : No. of Fog devices and F : Flow Then some variables are created as needed to define restrictions:

c (i , j) → capacity of the route between i and j

d (i , f) → Weight of the flow f that goes from i

store (j) → capacity of storage of j

proc (j) → capacity of processing of j

freq (i) → frequency of í i.e of flows of i

In addition, the variable defined x (i , j , f) that takes a value between 0 and 1 depending on the flow rate f sent by link i and j .

Later, the objective function, which is the maximum of the sum of the amount of memory required for all flows was set multiplied by x (i , j , f) divided the capacity of each Fog thus :

$$\boldsymbol{\alpha} = Max\ (\ \boldsymbol{\alpha}\ (\ i\ ,j\ ))$$

$$c\ (\ i\ ,j\ )\ *\ \boldsymbol{\alpha}\ (\ i\ ,j\ ) = \sum d\ (\ i\ ,f\ )\ *\ x\ (\ i\ ,j\ ,f\ )$$

Finally, the constraints are defined. First, the memory constraint: the memory available on the Fog device must be greater than the memory required by the stream. Second, the processing capacity restriction: the processing capacity in the fog device must be greater than the frequency of the flows. Finally, total of the weight of the flows passing through the link between i and j must be less than the binding capacity.

Storage = d(i , f) * x(i , j , f) store(j) * x(i , j , f)

Processing = freq(i) * x(i , j , f) ≤ proc(j) * x(i , j , f)

Size of N devices data flow

$$= \sum_i \sum_j \sum_f (\ d\ (\ i\ ,f\ ) * x\ (\ i\ ,j\ ,f\ ))\quad \le c(i,\ j)$$

The above equation will give total data flow in the model designed for test and simulation.

5. Implementation and Results

Regarding the implementation of the optimization, the mathematical model designed previously in GAMS was reflected. To verify the model, tests were first performed with a small case of 2 Fog devices and 5 IoT devices. When making the corresponding corrections and collaborating the proper functioning of the model, a real case was entered with more than a thousand IoT devices.

First, the sets of IoT devices, Fog and flows were defined as given below.

```
Sets
i Devices IoT  /n1 * n2 000/
j Devices Fog  /m1* m3 /
f flow         /f1*f4/
```

Next, the parameters required for the restrictions were added and can be shown as below.

```
cap(i,j)   capacity of route
/            m1     m2
n1           500    1024
n2           1024   556
n3           800    700
n4           956    256
n5           534    1024/
stor(j) Storage capacity of FOG devices
in KB
/m1          8000
m2           20000/
proc(j) Processing capacity of Devices in
HZ
/m1          1.5
m2           2
/
freq(i) sending frequency of IoT devices
/n1          0.1
n2           1.2
n3           1
n4           0.5
n5           1/
```

Then, the variables was defined as shown below:

```
Variables
X(i, j, f) indicates the percentage
of f to be sent by link i, j
max(i,j)  maximum use of each link
alpha   minimization   of   maximum
utilization
load  ……..  alpha  =  e=  smax((i,j),
max(i,j));
```

Objective function was specified as below .

```
load  ……..  alpha  =  e=  smax((i,j),
max(i,j));
```

Subsequently, the restrictions were set as shown

```
alph(i,j)  …    c(i,j)  max(i,j)=  e=
sum((f), d(I,f) *x(I, j, f));

storage(j)               ..sum((i,f),
d(I,f)*x(i,j,f))= 1=store(j)

processing(j)    ..sum((i,f),  freq(i)
*x(i,j,f))= 1=proc (j)

flow(i,j)                  ..sum((f),
d(i,f)*x(i,j,f))= 1=c(i,j);
```

Finally, the solution to use was selected as using

```
solve   transport   using   minimizing
alpha;
```

As a base case, we defined an architecture with 2 Fog devices, 5 IoT devices and each device sent a flow with the following weights as below figure.

flow1 → 675 MB
flow2 → 512 MB
flow3 → 500 MB
flow4 → 856 MB

With this architecture and the previous model, the result shown in figure was obtained:

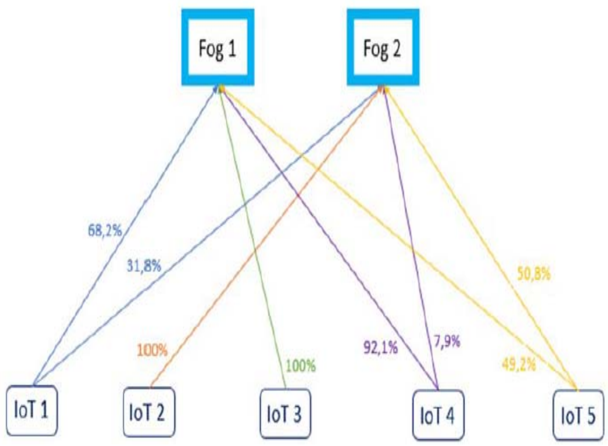

Fig. 2 Connected IoT Devices with Fog Devices with data flow in percentage

In the figure-2, the results obtained were clearly graphically shown and thus verify their correct operation.

There, you can see the links between the device and the respective percentage of flow that must be sent by each channel to comply with the optimization. In this way the implementation could be validated and then used in a larger case. The results obtained from this case can be seen better reflected in the simulation.

6. Simulation and Validation

Using Python, an application was created whose parameters are the same used in the GAMS model, flows, IoT and fogs with all their characteristics. The last parameter is a matrix with the percentages that the flow generated by IoT n must send to FOG n.

In the simulator an IoT is defined by a unique name, a necessary processing and a time interval for sending data. A FOG device for a unique name, processing power and memory. A flow represents the exchange of data between an IoT and a FOG and is defined by a percentage, a size, and a source and destination. The simulation begins with all the IoTs generating a signal in time, if the simulation time is equal to the time of the signal, it is transferred to the FOGs through a flow. The FOG device attends its pending queue until its time is equal to that of the simulation or until it ends, after this it checks if it has enough resources to attend to the request, if so, it is stored in a priority queue ordered by time arrival, if not added to losses. The simulator returns the average losses, CPU usage and memory usage in each FOG device for each minute of the simulation in an excel file and in graphs.

3 validation tests have been done. The first is to direct the entire flows of a uniform group of IoT to a specific fog. The second is to randomize the percentage of each flow to send from each IoT to each FOG. The last one is to use the answer given by the optimization. The use of the mathematical model is expected to significantly reduce the losses of the other simulations. Each simulation will be done 10 times and a z-test with a significance of 95% will be used to compare if the mean of losses in the 3 tests is the same or if it can be said that the losses using optimization are lower. Each IoT generates data in an interval of 1 to 5 minutes.

The optimization result was obtained using CONOPT allowing its replicability. Each simulation ran for 100 minutes. For each one, the total KB lost, the value of the objective function, the average memory and CPU usage in each minute were saved. 2000 IoT were used, with 4 flows each, 3 FOG, m1, m2, m3. There are links from all IoTs to all FOGs.

CASE I: The result of the optimization for the large-scale case is presented in figure 3 and 4 part A and B

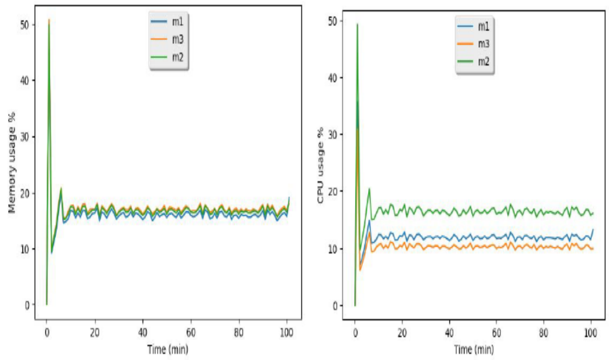

Fig.3 : Part A on the left shows the memory usage in the FOGs, part B on the right shows CPU usage for one of the CASE I simulations.

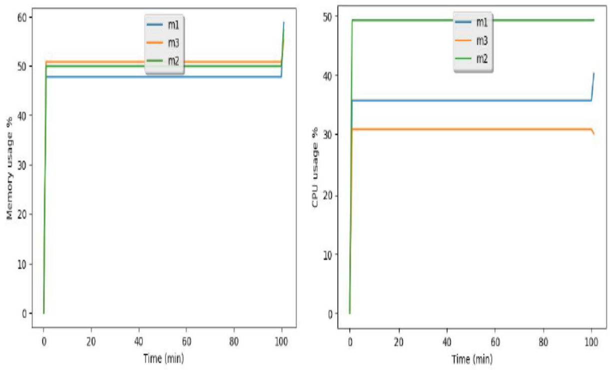

Fig.4 : Part A on the left presented the memory usage in FOGs, part B on the right the CPU usage for another CASE I simulations.

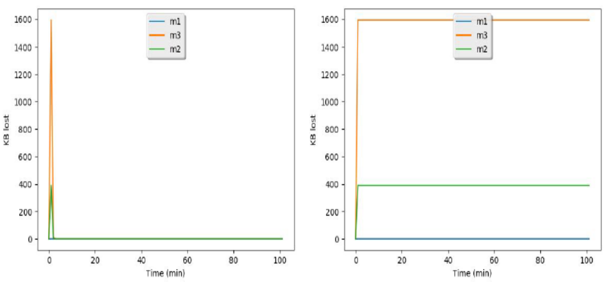

Fig.5: data lost (in KB) in two of the simulations for the case of CASE I.

CASE II: Assigning ⅓ of the flow of each IoT to each FOG, the following results were obtained

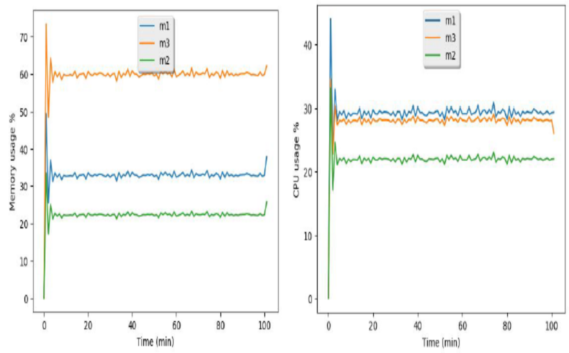

Fig. 6 : Part A on the left shows the memory usage in the FOGs, part B on the right shows CPU usage for one of the CASE II simulations.

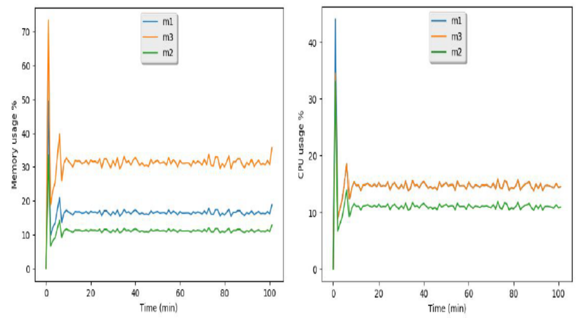

Fig. 7 : Part A on the left presented the memory usage in FOGs, part B on the right the CPU usage for another CASE II simulations.

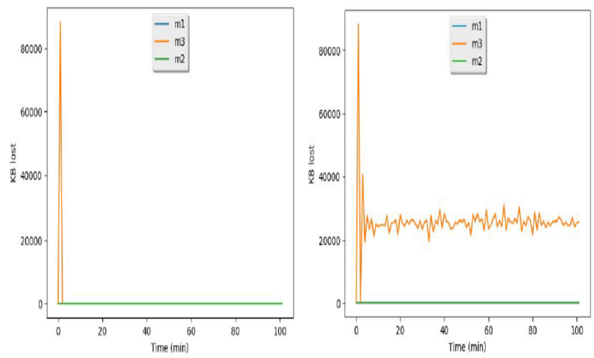

Fig. 8: data lost (in KB) in two of the simulations for the case of CASE II.

CASE III: Randomizing the percentage of each flow sent from an IoT to each FOG, the following results were obtained

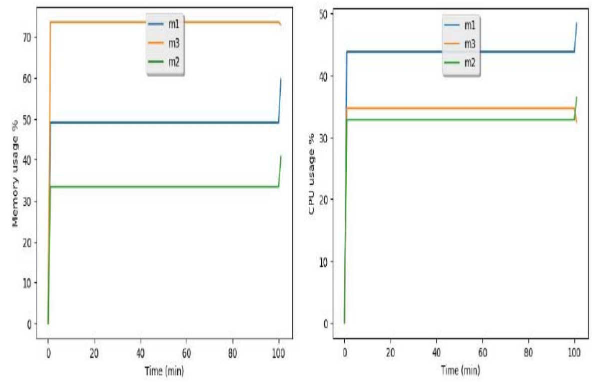

Fig. 9 : Part A on the left shows the memory usage in the FOGs, part B on the right shows CPU usage for one of the CASE III simulations.

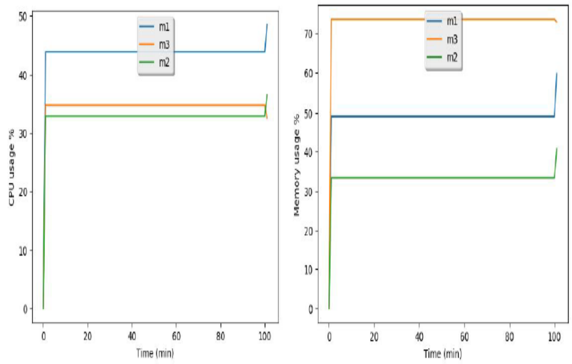

Fig. 10 : Part A on the left presented the memory usage in FOGs, part B on the right the CPU usage for another CASE III simulations.

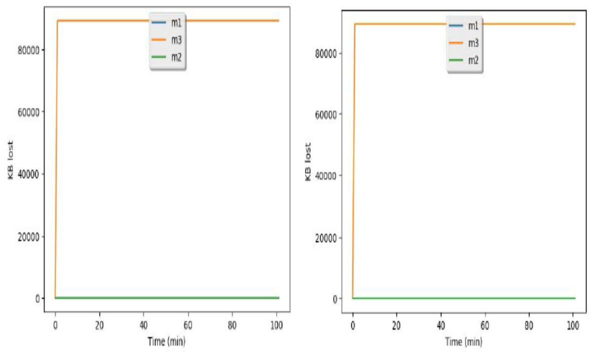

Fig. 11: data lost (in KB) in two of the simulations for the case of CASE III.

The Table no. 1 presents the value of the objective function for each case. Figure 19 the total losses for each simulation

Table 1. Value of the Objective Function for each case

Case	α
CASE I	0.405
CASE II	0.266
CASE III	0.469

Taking H_0 as the null hypothesis and H_1 as an alternative hypothesis, a t-test was performed with 95% confidence. Table 2 presents the results when comparing the average losses of the optimized case with the other two cases. Minitab software was used to obtain the results.

$H_0 = \mu\ opti - \mu\ sample = 0$

$H_1 = \mu\ opti - \mu\ sample < 0$

Table 2. Results of the t-tests

Case	p-value	Conclusion
CASE I vs CASE III	0	Optimization is better
CASE I vs CASE II	0	Optimization is better

7. Conclusion

This project was divided into three main stages, the first was the mathematical modeling of the problem. This part of the project was developed iteratively: a model was designed, reviewed with the consultant looking at correct and incorrect aspects of the model, and redesigned until the correct model was obtained. The second stage was optimization. In this phase of the project, the model of the previous part was implemented in GAMS, this optimization was carried out with a base case of 2 Fog and 5 IoT devices, the results were verified and the model was adjusted to achieve the optimum expected. At the end of this phase, a model was obtained that allows optimizing cases of any size. The last stage was the validation stage, which was carried out through simulation. For the simulation, a custom-made simulator had to be developed since the simulators found as Opnet and NS3 do not allow entering the necessary parameters or keeping a record of the measurements required to check the status of the simulation. In terms of results, the average use of resources in the FOGs stabilizes after a while. Additionally, there are losses in all cases after 1 minute. This may be due to the fact that small packets that take less time to process are generated more frequently than other packets whose memory and processing requirements are much higher. Despite this, they are maximum 1600 KB per minute.

Acknowledgments

This project was supported by the deanship of scientific research at Prince Sattam Bin Abdulaziz University, Alkharj, Saudi Arabia under research project number 2019/01/11534.

References

[1] Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). Fog Computing and Its Role in the Internet of Things. In Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing (pp. 13-16). New York, NY, USA: ACM. https://doi.org/10.1145/2342509.2342513

[2] https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/ (Access Available on 11 Nov,2020)

[3] Andersson, M.. Short range low power wireless devices and Internet of Things (IoT). Contributed By Digi-Key Corporation , 01-1 . 6. https://www.digikey.com/en/articles/short-range-low-power-wireless-devices-and-internet-of-things-iot (Access Available on 11 Nov,2020)

[4] A. V. Dastjerdi and R. Buyya, "Fog Computing: Helping the Internet of Things Realize Its Potential," in Computer, vol. 49, no. 8, pp. 112-116, Aug. 2016. doi: 10.1109/MC.2016.245

[5] Ling C, Lifang L, Xiaogang Q, Gengzhong Z (2017) Cooperation forwarding data gathering strategy of wireless sensor networks. Int J Grid Util Comput 8(1):46–52. https://doi-org.sdl.idm.oclc.org/10.1504/IJGUC.2017.10003009

[6] S. Ahmad, Afzal M.M., Review of Assured Data Deletion Mechanism in Cloud Computing, International Journal of Engineering & Technology (UAE), Vol 7, No 4.5 (2018), Special Issue 5, 2018.

[7] Ahmad S., Afzal M.M. (2020) Deployment of Fog and Edge Computing in IoT for Cyber-Physical Infrastructures in the 5G Era. In: Karrupusamy P., Chen J., Shi Y. (eds) Sustainable Communication Networks and Application. ICSCN 2019. Lecture Notes on Data Engineering and Communications Technologies, vol 39. Springer, Cham

[8] Arkian, HR, Diyanat, A., & Pourkhalili, A. (2017). MIST: Fog-based data analytics scheme with cost-efficient resource provisioning for IoT crowdsensing applications. Journal of Network and Computer Applications , 82 , 152-165. https://doi.org/10. 1016 / j.jnca.2017.01.012 .

[9] Chen, T. Zhang and W. Shi, "Fog Computing," in IEEE Internet Computing, vol. 21, no. 2, pp. 4-6, Mar.-Apr. 2017, doi: 10.1109/MIC.2017.39.

[10] Sultan Ahmad, Mohammad Mazhar Afzal and Abdullah Alharbi (2020). Big Data Analytics with Fog Computing in integrated Cloud Fog and IoT Architecture for Smart Devices, International Journal of Computer Science and Network Security, Vol. 20 No. 6 pp. 171-177; http://paper.ijcsns.org/07_book/202006/20200620.pdf

[11] Manuel Díaz, Cristian Martín, Bartolomé Rubio,State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing,Journal of Network and Computer Applications, Volume 67, 2016, Pages 99-117, ISSN 1084-8045, https://doi.org/10.1016/j.jnca.2016.01.010.

[12] M. Y. uddin and S. Ahmad, "A Review on Edge to Cloud: Paradigm Shift from Large Data Centers to Small Centers of Data Everywhere," 2020 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2020, pp. 318-322, doi: 10.1109/ICICT48043.2020.9112457

[13] Xu, J.; Palanisamy, B.; Ludwig, H.; Wang, Q. Zenith: Utility-Aware Resource Allocation for Edge Computing, In Proceedings of the IEEE International Conference on Edge Computing (EDGE), Honolulu, HI, USA, 25–30 June 2017; pp. 47–54.

[14] C. Chang, S. Narayana Srirama and R. Buyya, Indie Fog: An Efficient FogComputing Infrastructure for the Internet of Things, in Computer, Vol. 50, no. 9, pp. 92-98, 2017

[15] Gupta H, Dastjerdi AV, Ghosh SK, Buyya R (2016) ifogsim: A toolkit for modeling and simulation of resource management techniques in internet of things, edge and fog computing environments. arXiv:abs/1606.02007