Simulating fog computing in OMNeT^{++}

Noor Razaq Obaied Al-Rubaie[1], Rafal Nader Neamah Kamel[2], Raghda M. Alshemari[3]
[1]Department of Nursing, Al-Mustaqbal University College, Babylon, Iraq
[2]Department of Pharmacy, Al-Mustaqbal University College, Babylon, Iraq
[3]Department of Anesthesia Techniques, Al-Mustaqbal University College, Babylon, Iraq

Article Info

Article history:

Received Jun 28, 2022
Revised Aug 13, 2022
Accepted Aug 28, 2022

Keywords:

Channel allocation
Cloud computing
Fog computing
Internet of things
OMNeT^{++}

ABSTRACT

Fog computing is a technology architecture in which data from IoT devices is received in real time by a number of nodes. These nodes process the data they receive in real time, with millisecond reaction times. The nodes communicate analytical summary data to the cloud on a regular basis. Fog computing scenario demands higher output, reduced latency, and greater performance as demand and requirements for improving performance in IoT applications grow. The resources allocation in effective manner in the fog environment is also a major problem in IoT-fog computing. Fog computing has been considered as a necessity within several IoT resources domains. In this paper the proposed fog simulation environment is focused on IoT sensors, fog node, and cloud as the used network architecture. However, the network features are properly explored in the proposed system and they are evaluated based on the throughput, latency, and channel allocation.

Corresponding Author:

Noor Razaq Obaied Al-Rubaie
Department of Nursing, Al-Mustaqbal University College
51001, Babylon, Iraq
Email: noor.razaq@uomus.edu.iq

1. INTRODUCTION

Fog computing is a creative architecture that emigrates some data center features to the server's edge. Fog computing is a distinct architecture based on edge servers that distributes limited computing, hardware, and networking capabilities between end devices and traditional cloud computing data centers. The primary goal of fog computing is to ensure low and predictable latency in latency-sensitive internet of things (IoTs) applications such as healthcare services [1].

In real-time applications, fog computing has a lot of advantages. It is often utilized in IoT applications that deal with real-time data. It functions as a more advanced kind of cloud computing [2]. It serves as a link between the cloud and end users (closer to end users). It can be utilized in both machine-to-machine and human-to-machine scenarios [3].

Fog computing is based on bringing networking resources closer to the nodes that generate data at the lowest perceptual layer. Fog is a highly virtualized platform that connects end nodes in an IoT context to traditional cloud computing, storage, and networking services [4]. In addition, resource management techniques are often divided into six groups. Which contribute to the application development process and the possibility of scheduling and providing resources and the process of unloading resource allocation and load balancing in some of the services it provides, where the use of the unit for the central processing and reaction time as well as productivity, energy consumption, data movement, and the percentage of service migration available to know the number of active nodes of the tools on which many researchers rely [5].

Required processes for enabling the fog computing assessment, network model can be determined for further evaluation [6]. Based on the proposition of the fog computing setting, different sets of resources and capabilities such as fog layer interpretation, resource selector, processing performance and evaluation, history analyzer, resource selector, task scheduler, IoT devices allocation, and management can be addressed [7]. However, these aspects can be interpreted to comply with the specific requirements of fog computing setting [8]. The only significant distinction between fog and cloud is that fog computes services at the network's edge. The importance of fog nodes as a result of their unrivaled gains is summarized in [9]: i) large storage areas, ii) heterogeneity-low latency, iii) allocation and interaction with a small number of IoT devices IoT tracking and proximity to end users, iv) network flexibility-virtual network support, v) strict security, vi) tolerance to faults, and vii) data outsourcing restrictions-infrastructure or hardware device management [10].

Fog computing provides several advantages in real-time applications. It's frequently used in IoT applications that deal with real-time data. It's a more sophisticated kind of cloud computing. It acts as a conduit between the cloud and the end consumers (closer to end users). It's suitable for both machine-to-machine and human-to-machine communication. The main layer of the fog computing sorted as follow [11]:

- The physical layer: this is the basic layer of the network that deals with the components that have the service of connecting to the internet and generating data, as it includes many nodes, physical and virtual sensors [12].
- The monitoring layer: it controls when and how the next job will be conducted. Nodes' energy usage is also monitored so that appropriate measures (such as task offloading) may be implemented in real time based on the sensor's condition [13].
- Data management is the responsibility of the preprocessing layer. Here, data is analyzed, and data cutting and filtering are conducted depending on the findings in order to reduce unwanted communication [14].
- The storage layer: this layer is in charge of storing data in a fog. The majority of the data in fog is only kept temporarily. The cloud is better for long-term storage since it has greater resources [15].
- The fog security layer: it protects data by providing proper security and privacy functions before delivering it over a public or susceptible channel [16].

The most related works in term of simulation fog computing network performance have been discussed and overviewed as follow: Ushakova *et al*. [17] relied on the creation of simulation models for the infrastructure of fog computing based on OMNeT^{++}, in addition to tools for assessing the state of the network and the dynamics of its PlayStations on a set of statistical models to generate special patterns that contribute to the movement of data and measure the extent of infrastructure development further. Famá *et al*. [18] presented a research and architectural concept for integrating IoT application development and simulation in edge and fog computing network environments. The proposed architecture is based on the node-RED application middleware and the FogNetSim^{++} extension. In this integration, a manager component is in charge of sharing data between the middleware and the simulator. The tests' results show how the integration works and how they can be used to verify IoT applications.

According to Ahammad *et al*. [19], to increase quality of service (QoS) for the IoT ecosystem, they recommended merging software-defined networking (SDN) with fog computing (FC). A method based on virtual partition is proposed using the architecture. Then, using the iFogSim simulator, a use case is presented and assessed. When comparing the fog with SDN execution to the cloud-only execution, the results reveal a considerable increase in various QoS measures. In comparison to prior similar use cases, the results indicate improved outcomes for energy usage, network utilization decrease, and reduce the latency.

Abdulqadir *et al*. [20] explore the latest in cloud and fog computing, as well as its confluence with IoT, emphasizing the benefits and challenges of adoption. It also focuses on cloud and fog design, as well as emerging IoT technologies that are boosted by the cloud and fog concept. Finally, open topics are discussed, as well as possible testing suggestions for cloud storage and fog computing, as well as IoT.

Research by Khumalo *et al*. [21], the goal of this study is to see how fog computing can help underserved areas overcome the economic barrier of low ARPU while providing a high-quality service that meets critical performance requirements for customers. They propose a fog-based architecture that uses device-to-device connectivity as well as local compute, storage, and communication to reduce communication costs and overcome 5G implementation budgetary constraints. According to preliminary simulation results, the design of the proposed increases throughput by a factor of two while reducing round-trip time by a factor of 4.

Lawal *et al*. [22] used network modeling toolsets to get around some of the drawbacks of traditional cloud computing, such as high energy consumption. An examination determined the best network simulation software tool for critically analyzing fog computing and standard cloud computing in terms of data management, planning, and energy efficiency. Using the iFogSim network simulation toolkit, they show that introducing fog computing layer technology reduces energy usage when compared to typical cloud computing architecture. Furthermore, the distinctions between fog and cloud computing data centers in terms of energy usage and data management are investigated.

Research by Gonçalves *et al*. [23], a method was adopted to contribute to the process of improving the special simulator called MobFogSim, which includes dynamic network slicing in service management for mobile fog networks, as well as contributing to adding an empirical evaluation of why the effect of network slicing in the fog environment to contribute to the reception of mobile devices and may lead to an increase in resource consumption and the posting tool for requests generated from the nodes. In this paper, simulating fog computing in OMNeT^{++} has been proposed. It is totally presented as follows: i) introduction, ii) method, iii) results and discussion, and iv) conclusion.

2. METHOD

The architecture of the proposed system involving three layers. The first layer is the hosts to transmit data. The second layer is fog node, which is control on the network component, and resource allocation through network transmission. The third layer is the top-most layer (cloud) is where the cloud components are located for historical data storage and big data processing. Due to the fog node has the main view on the network connection and data traffic pass through out so it will build the network setting during the initialize state. The proposed system aims to achieve high network performance for all the types of computing nodes in the fog environment. So, after network executed and once the transmission costs of the links are calculated depending on the host devices traffic to the fog node, which redirect the traffic to the cloud network element for response to specific link required or for data storage service on the cloud. In addition, the main block of the proposed system showed in Figure 1, which it describes the simulation stages based on OMNeT^{++} to create the proposed fog network topology with the three main elements as the host devices, fog node, and cloud to provide HTTP, file upload/download services.

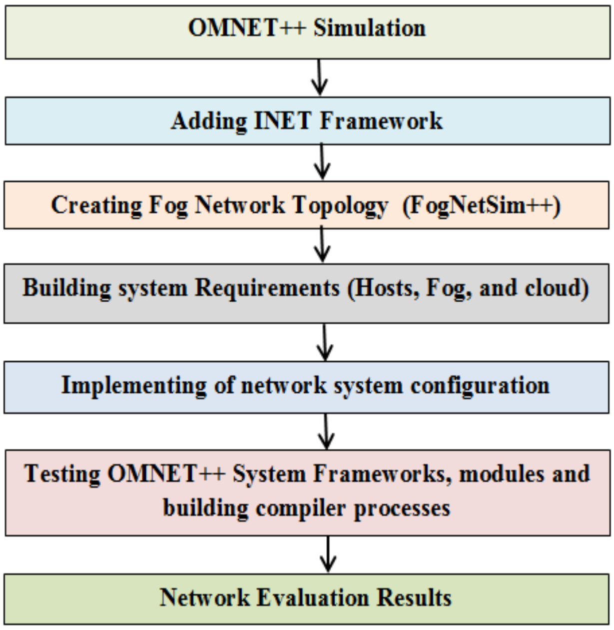

Figure 1. Block diagram of the system

2.1. Fog implementation environment

The proposed system was tested in an environment that met the parameters listed in Table 1. Furthermore, the suggested system's code was created in the OMNeT^{++} simulation tool's C^{++} language. While Table 2 presents the specifications of each network elements which effected on the resource allocation process, with network configuration IP address, network interface, and port number.

Table 1. Environment specifications for the proposed system

Operating systems	Windows 7, 32-Bit
CPU	Core (TM) I 7-4210U
RAM	8.00 GB
Implementation tools	OMNeT++ 4.6, INET 3.3.0, and FogNetSim++

Table 2. The specification of the fog environment

Node IP	The behavior	Network interface
10.0.0.2/30	IoT sensor 1	eth0-eth0
10.0.0.6/30	IoT sensor 2	eth0-eth1
10.0.0.22/30	IoT sensor 3	eth0-eth5
10.0.0.30/30	IoT sensor 4	eth0-eth7
10.0.0.14/30	IoT sensor 5	eth0-eth3
10.0.0.18/30	IoT sensor 6	eth0-eth4
10.0.0.1/30	Fog Node	eth0-to-eth8
10.0.0.10/30	Cloud	eth0-eth2

2.2. The proposed method

The major goal of the suggested technique, which is based on fog control data flow to the cloud, is to maximize the system throughput by maximizing the usage of each host's processing capabilities and reducing the average task response time. The fog node receives and forwards all task requests to the cloud for processing. In fog environment, the architecture has three layers: the first layer is hosts, the second layer is fog, and the last layer is the cloud resources, as it showed in Figure 2.

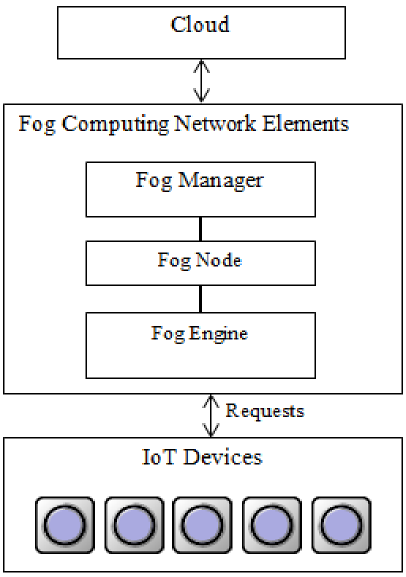

Figure 2. The proposed system architecture in fog computing environment

2.3. Evaluation metrics

The suggested system is based on the following three assessment criteria:

- The number of packets received divided by the total number of packets transmitted is known as throughput, and the equation that was utilized as [24]:

$$throughput = \frac{number\ of\ recieved\ packets}{total\ number\ of\ send\ packets} \quad (1)$$

- The latency time is the time it requires for the packet to travel from sender to receiver, and it is used to calculate the delay time simulation parameter for full packets. The (2) is used to calculate it [25]:

$$d_{trans} = {}^{L}/_{R} \quad (2)$$

Where d denotes the delay time in seconds, L denotes the packet length in bits, and R denotes the data rate in bits per time unit [25].

– Channel resource allocation: it represents the maximum utilization of the resources available in a fog system [26].

3. RESULTS AND DISCUSSION

The propose system results based on the two case studies. The 1st is the network evaluation system with the 5 IoT devices. The 2nd is the case of 10 IoT devices evlaued to recognize the effecting of the network with increasing number of nodes.

3.1. The 1st case study

The proposed system in the 1st case study based on 5 IoT devices. As Figure 3 showed the network topology. The Table 3 showed the system results.

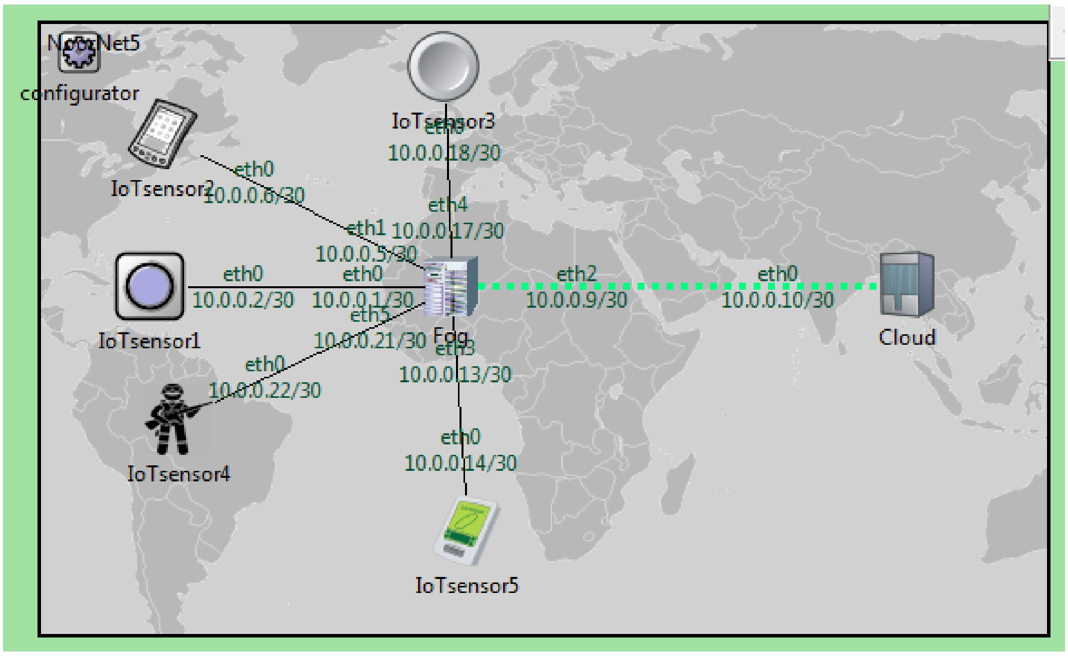

Figure 3. The proposed implementation system in OMNET++ environment

Table 3. The throughput packets, latency, and channel allocation of the 1st case study

Requests	Total no. of packets	Throughput packets/Bps	Latency/ms	Channel allocation (%)
IoT-device 1	1,414	38.217	17800.462	96
IoT-device 2	317	31.71	10414.986	98
IoT-device 3	789	41.57	19515.24	97.2
IoT-device 4	584	44.973	13845.459	95
IoT-device 5	442	36.853	12812.04	96
Cloud	15,945	192.119	83938.20	99.2
Fog	15,536	194.210	80868.54	98.5
Packet size	1,024 bytes			

3.2. The 2nd case study

It is based on the 10 IoT device with the main evlauation metrics, Figure 4 showed HTTP request by nodes. Figure 5 showed the replay to the request by fog node and Table 4 showed the evaluation metrics. The proposed system's results show that as the number of devices in the network grows, the impact on the evlauation metrics grows, as shown in Figure 6.

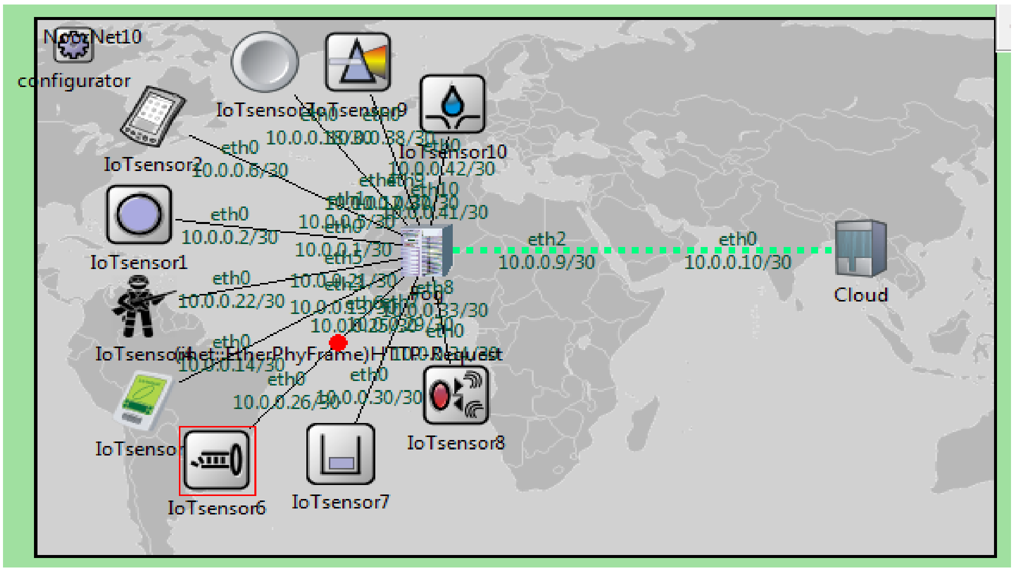

Figure 4. The proposed HTTP request by IoT sensor device

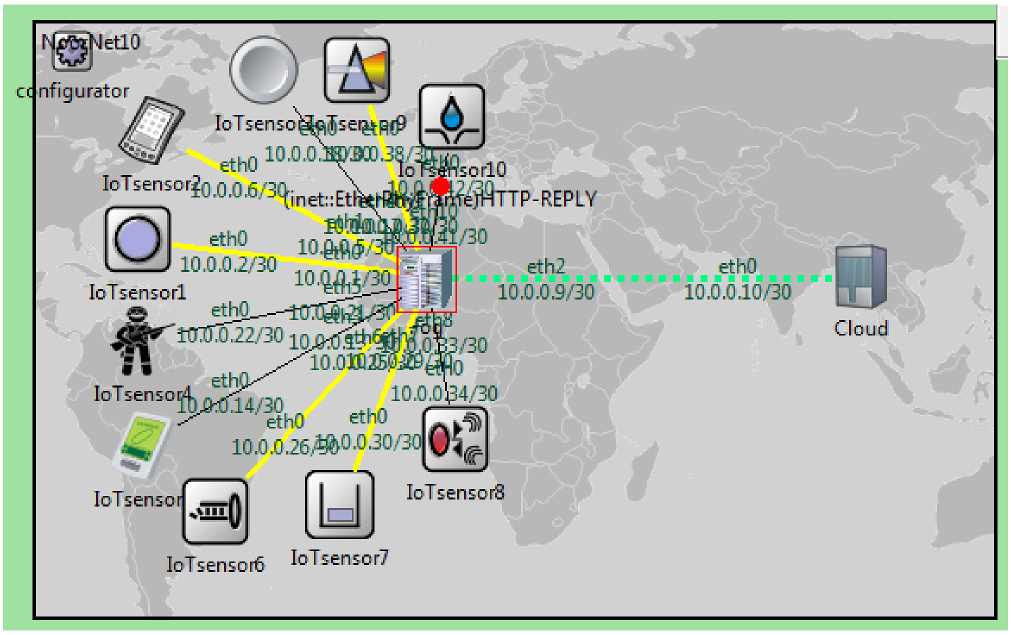

Figure 5. The proposed fog node replied to the request by IoT nodes

Table 4. The throughput packets, latency, and channel allocation of the 2nd case study

Requests	Total No. of packets	Throughput packets/Bps	Latency/ms	Channel allocation
IoT-device 1	523	24.23527	21580.12	94
IoT-device 2	247	23.34543	20986.23	92
IoT-device 3	314	26.98	23638.25	95
IoT-device 4	378	31.89914	29849.85	90
IoT-device 5	244	22.39019	21897.63	91
IoT-device 6	613	18.94995	17348.37	89
IoT-device 7	268	25.22771	22623.24	88
IoT-device 8	349	29.4075	25867.72	92
IoT-device 9	272	24.8746	20934.85	89.3
IoT-device 10	298	27.37116	25887.37	93
Cloud	14350	162.8182	98135.11	88.5
Fog	13982	164.6647	94911.967	89.7
Packet size		1,024 Bytes		

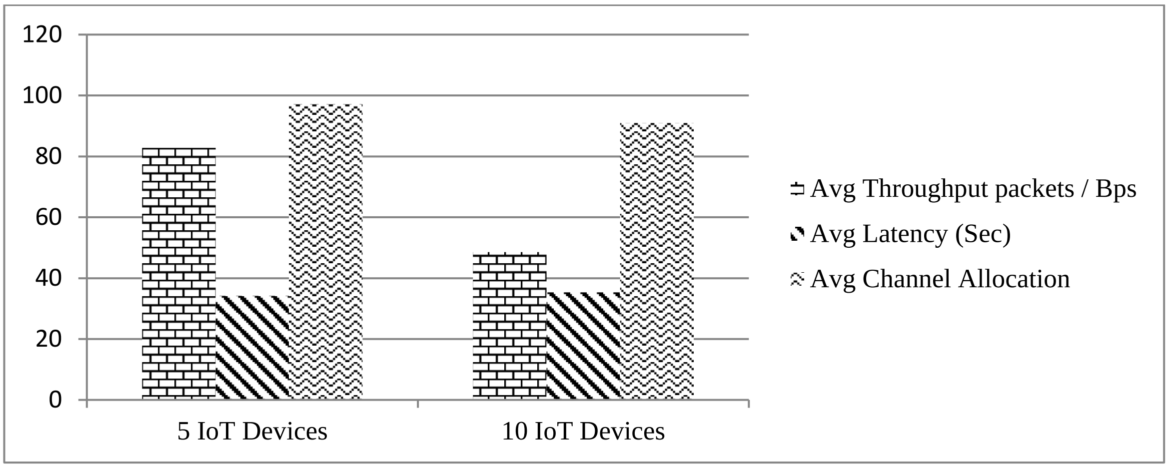

Figure 6. The average payload throughput, average delay, and average of channel allocation of the proposed system

4. CONCLUSION

Fog computing refers to a distributed processing system in which computer, storage, and software are located between the data source and the cloud. Running the IoT application in the fog situation is difficult because of the resource allocation difficulty. The resource allocation strategy used to regulate the degree of QoS efficiency in cloud computing is the most important factor. This study combines IoT with FC to provide an IoT-based fog computing paradigm. The quality checks were done with OMNeT^{++} and the fog NetSim^{++} addon. The proposed scenario with FC is particularly useful, according to the article, since packets from comparable directions are added to the network root fog node manager using learning automats, and data exchange rates are utilized to control delay time, channel availability, and data exchange rates. The proposed FC scenario is particularly helpful, according to the article, since packets from comparable directions are added to the network root fog node manager using learning automats and data exchange rates are employed to regulate delay time, channel allocation, and throughput.

ACKNOWLEDGEMENTS

The authors are grateful for the financial and technical support from Al-Mustaqbal University College (grant code: MUC-G-0322).

REFERENCES

[1] R. Neware and U. Shrawankar, ‘‘Fog computing architecture, applications and security issues,’’ *International Journal of Fog Computing (IJFC)*, vol. 3, no. 1, pp. 75-105, 2020, doi: 10.4018/IJFC.2020010105.
[2] P. Habibi, M. Farhoudi, S. Kazemian, S. Khorsandi, and A. Leon-Garcia, "Fog Computing: A Comprehensive Architectural Survey," in *IEEE Access*, vol. 8, pp. 69105-69133, 2020, doi: 10.1109/ACCESS.2020.2983253.
[3] Y. Yang, X. Luo, X. Chu, and M. T. Zhou, ‘‘Fog computing architecture and technologies,’’ In *Fog-Enabled Intelligent IoT Systems*, Springer, Cham, 2020, pp. 39-60, doi: 10.1007/978-3-030-23185-9_2.
[4] G. Caiza, M. Saeteros, W. Oñate, and M. V. Garcia ‘‘Fog computing at industrial level, architecture, latency, energy, and security: A review,’’ *Heliyon*, vol. 6, no. 4, p. e03706, Apr. 2020, doi: 10.1016/j.heliyon.2020.e03706.
[5] V. Karagiannis and S. Schulte, "Comparison of Alternative Architectures in Fog Computing," *2020 IEEE 4th International Conference on Fog and Edge Computing (ICFEC)*, 2020, pp. 19-28, doi: 10.1109/ICFEC50348.2020.00010.
[6] Z. Mahmood and M. Ramachandran, ‘‘Fog computing: Concepts, principles and related paradigms,’’ in *Fog Computing*. Springer, Cham, 2018, pp. 3-21, doi: 10.1007/978-3-319-94890-4_1.
[7] K. Kaur and M. Sachdeva, ‘‘Fog computing in IoT: An overview of new opportunities,’’ *Proceedings of ICETIT 2019*, Sep. 2019, 59-68, doi: 10.1007/978-3-030-30577-2_5.
[8] T. Kim, S. E. Yoo, and Y. Kim, ‘‘Edge/Fog Computing Technologies for IoT Infrastructure,’’ *Sensors*, vol. 21, no. 9, p. 3001, Apr. 2021, doi: 10.3390/s21093001.
[9] K. Matrouk and K. Alatoun, ‘‘Scheduling Algorithms in Fog Computing: A Survey,’’ *Int. J. Networked Distributed Comput.*, vol. 9, no. 1, pp. 59-74, 2021, doi: 10.2991/ijndc.k.210111.001.
[10] P. Hosseinioun, M. Kheirabadi, S. R. K. Tabbakh, and R. Ghaemi, ‘‘aTask scheduling approaches in fog computing: a survey. *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 3, p. e3792, 2022, doi: 1002/ett.3792.
[11] M. S. Sofla, M. H. Kashani, E. Mahdipour, and R. F. Mirzaee, ‘‘Towards effective offloading mechanisms in fog computing. *Multimedia Tools and Applications*, vol. 81, no. 2, pp. 1997-2042, 2021, doi: 10.1007/s11042-021-11423-9.
[12] Z. M. Nayeri, T. Ghafarian, and B. Javadi, ‘‘Application placement in Fog computing with AI approach: Taxonomy and a state of the art survey,’’ *Journal of Network and Computer Applications*, vol. 185, p. 103078, Jul. 2021, doi: 10.1016/j.jnca.2021.103078.
[13] S. P. Ahuja and N. Wheeler, ‘‘Architecture of fog-enabled and cloud-enhanced internet of things applications,’’ *International Journal of Cloud Applications and Computing (IJCAC)*, vol. 10, no. 1, pp. 1-10, 2020, doi: 10.4018/IJCAC.2020010101.

[14] P. Maheshwari, A. K. Sharma, and K. Verma, ‘‘Energy efficient cluster based routing protocol for WSN using butterfly optimization algorithm and ant colony optimization,’’ *Ad Hoc Networks*, vol. 110, p. 102317, Jan. 2021, doi: 10.1016/j.adhoc.2020.102317.
[15] U. M. Malik, M. A. Javed, S. Zeadally, and S. U. Islam, "Energy-efficient fog computing for 6G-enabled massive IoT: recent trends and future opportunities," *IEEE Internet of Things Journal*, vol. 9, no. 16, pp. 14572-14594, 2022, doi: 10.1109/JIOT.2021.3068056.
[16] J. Singh, P. Singh, and S. S. Gill, ‘‘Fog computing: A taxonomy, systematic review, current trends and research challenges,’’ *Journal of Parallel and Distributed Computing*, vol. 157, pp. 56-85, Nov. 2021, doi: 10.1016/j.jpdc.2021.06.005.
[17] M. Ushakova, Y. Ushakov, I. Bolodurina, A. Shukhman, L. Legashev, and D. Parfenov, ‘‘Creation of adequate simulation models to analyze performance parameters of a virtual fog computing infrastructure,’’ *Procedia Computer Science*, vol. 186, pp. 603-610, 2021, doi: 10.1016/j.procs.2021.04.182.
[18] F. Famá, D. F. S. Santos, and A. Perkusich, "Integrating an IoT Application Middleware with a Fog and Edge Computing Simulator," *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 2020, pp. 1-6, doi: 10.23919/SoftCOM50211.2020.9238245.
[19] I. Ahammad, M. A. R. Khan, Z. U. Salehin, M. Uddin, S. J. Soheli, ‘‘Improvement of QOS in an IoT ecosystem by integrating fog computing and SDN,’’ *International Journal of Cloud Applications and Computing*, vol. 11, no. 2, pp. 48-66, 2021, doi: 10.4018/IJCAC.2021040104.
[20] H. R. Abdulqadir *et al*., ‘‘A study of moving from cloud computing to fog computing,’’ *Qubahan Academic Journal*, vol. 1, no. 2, pp. 60-70, 2021, doi: 10.48161/qaj.v1n2a49.
[21] N. Khumalo, O. Oyerinde, and L. P. Mfupe, ‘‘A device-to-device enabled fog computing architecture for 5G and IoT in underserved areas,’’ *Southern Africa Telecommunication Networks and Applications Conference (SATNAC)*, 2019.
[22] K. N. Lawal, T. K. Olaniyi, and R. M. Gibson, ‘‘Fog computing infrastructure simulation toolset review for energy estimation, planning and scalability,’’ *International Journal of Sustainable Energy Development*, vol. 9, no. 1, pp. 421-426, 2021.
[23] D. Gonçalves, C. Puliafito, E. Mingozzi, O. Rana, L. Bittencourt, and E. Madeira, "Dynamic Network Slicing in Fog Computing for Mobile Users in MobFogSim," *2020 IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC)*, 2020, pp. 237-246, doi: 10.1109/UCC48980.2020.00042.
[24] R. Prasad, M. Jain, and C. Dovrolis, ‘‘Effects of interrupt coalescence on network measurements,’’ In *International Workshop on Passive and Active Network Measurement*, 2004, pp. 247-256, doi: 10.1007/978-3-540-24668-8_25.
[25] L. Yin, J. Luo, and H. Luo, "Tasks Scheduling and Resource Allocation in Fog Computing Based on Containers for Smart Manufacturing," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4712-4721, 2018, doi: 10.1109/TII.2018.2851241.
[26] Z. Zhou, P. Liu, J. Feng, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Computation Resource Allocation and Task Assignment Optimization in Vehicular Fog Computing: A Contract-Matching Approach," in *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3113-3125, Apr. 2019, doi: 10.1109/TVT.2019.2894851.

BIOGRAPHIES OF AUTHORS

Noor Razaq Obaied Al-Rubaie she worked as an external lecturer at the College of Information Technology, University of Babylon, for morning and evening studies in 2015, 2016, and 2017. A lecture at the College of Information Technology/Department of Information Networks. She holds a certificate of teaching methods and educational qualification. Currently works at the Department of Nursing, Al-Mustaqbal University College. She can be contacted at email: noor.razaq@uomus.edu.iq.

Rafal Nader Neamah Kamel Rafal Nader received Master degrees in Information Technology/Computer Networks from University of Babylon, 2020. she is interested with network security, web design, and computer comunication. Currently works at the Department of Pharmacy, Al-Mustaqbal University College. She can be contacted at email: rafal@uomus.edu.iq.

Raghda M. Alshemari she worked as a lecturer at the College of Information Technology/University of Babylon for 7 years and taught various subjects in the two departments of Information Networks and Information Security. Now, she works as a teacher in the Department of Anesthesia Techniques at Al-Mustaqbal University College. She teaches Computer Applications. Her graduation project won the award for the best project for the completed stages, third place in Iraq in the field of engineering and technology, and she was honored by the Minister of Higher Education and Scientific Research on the Iraqi Science Day, which falls on Thursday 22/9/2016. She has research published within IEEE. She can be contacted at email: raghda.mehdi@uomus.edu.iq.