International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Dynamic Resource Allocation for Real-Time Task Scheduling in Cloud-Fog Computing: A Cost-Effective and Low-Latency Approach

Hardik Mahendrabhai Patel[1], Dr. Kirit Modi[2]

Submitted: 27/04/2023 **Revised**: 25/06/2023 **Accepted**: 05/07/2023

Abstract: The act of implementing a work organizing in cloud fog processing significantly contributes to our understanding of the technology's capabilities. Programmer can make instructed choice about optimizing their Cloud-Fog computing applications by comprehending the effectiveness of the scheduling algorithm. Task scheduling Procedure in Cloud-Fog computing serve the purpose of resource allocation for tasks, aiming to achieve maximum efficiency standard, including Processing time transmission capacity and Node quantity utilization. Various performance requirements able to meet by employing dissimilar scheduling algorithms, each with unique Pros and cons. By conducting achievement evaluation, one capable of detect the most suitable arranging technique supporting a specific use taking in to consideration feature like as Processing time, node usage, and assignment preference. In addition, conduct investigation provides insights into the adaptability of work organizing method and optimization possibilities for different workloads. Within a Cloud-Fog environment, an algorithm is implemented to facilitate task execution, considering essential parameters like time required and the Number of fog nodes used, among others

Keywords: *Priority, Fog Computing, Energy Efficient, Task Scheduling, Resource Utilization*

1. Introduction

The field of cloud computing has been constantly evolving, and there have been several notable advancements in recent years that mark a new era in this technology. Here are some key developments shaping the new era in cloud computing are Edge Computing, Fog Computing, IOT, Machine Learning. Edge computing, fog computing, MEC, distributed cloud, and 5G networks address IoT difficult by enabling low latency, high reliability, and enhanced user experience through localized conducting and distributed assets nearer to the boundary. [1]

The integration of Internet of Things (IoT) technology in intelligent factories enables heterogeneous devices to connect, transforming isolated information islands into interconnected systems. Intelligent manufacturing, driven by personalized production and flexible production lines, demands real-time, energy-efficient, and reliable computing systems. [2]

Fog computing establishes a virtual network bridging IoT devices and cloud servers, offering computing and storage services at the network's edge. It comprises distributed, low-performance computers resembling end-users, forming a distributed architecture. When a service request is made from a network's terminal computer, fog computing filters, pre-processes, and analyses data, reducing the burden on cloud data centres. The proliferation of IoT technologies leads to an increasing number of mobile devices at the network's edge, demanding storage and analysis of vast data for diverse user requests. While cloud computing excels in data storage and processing, its distant central servers introduce significant delays, greatly impairing quality of service, particularly in time-sensitive applications. [3]

One solution is to leverage cloudlets, which enable local computations and data processing, reducing communication and processing costs. However, their limited coverage area and resource constraints hinder ubiquitous computing. Another approach is Mobile Cloud Computing (MCC), offering cloud advantages for cell phone users by distributing tasks between edge devices and the cloud. Nevertheless, resource constraints in MCC devices lead to increased delay. To overcome these limitations, fog computing is recommended, extending Decentralization of cloud support to perimeter network. in Nearer by proximity to gadgets and detectors. This ensures efficient and timely data processing and analysis. [4]

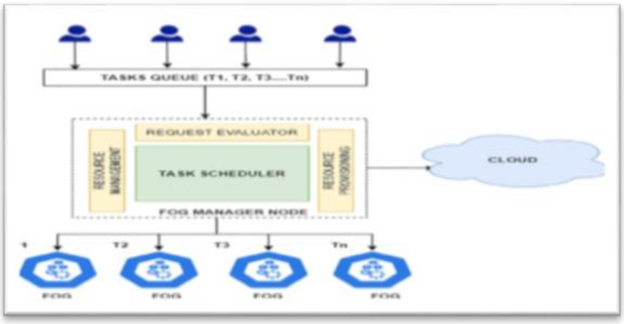

Fig 1. Task Scheduling in Fog Computing

[1] *Research Scholar, Sankalchand Patel University, Visnagar, Gujarat, India*
[2] *Professor, Head Department of CE/IT Faculty of Engineering and Technology, Sankalchand Patel University, Visnagar, Gujarat,India.*
** Corresponding Author Email:profhmp9@gmaill.com*

Table 1. Compression Between Cloud- Fod Computing

Parameters	***Cloud Computing***	***Fog Computing***
Location of Processing	Centralized data centers	Distributed computing closer to data sources
Latency	Higher latency due to distant data centers	Lower latency due to proximity to data
Bandwidth	High bandwidth consumption for data transfer	Lower bandwidth consumption for local processing
Scalability	Highly scalable infrastructure	Scalability depends on available Fog resources
Energy Efficiency	Requires large energy consumption for servers	More energy-efficient with localized processing
Cost	Cost-effective for large-scale processing	Cost may vary based on infrastructure and scale

2. Related Work

2.1. The task Scheduling Fog Computing

The system initiates resource availability assessment on fog nodes to cater to client requirements. In case of insufficient resources, the request is forwarded to the cloud layer. Furthermore, each client request is processed within the fog node and prioritized accordingly. The cloud layer consists of cloud data centers at the top, while the fog layer is managed by multiple fog nodes or Fog Servers, equipped with virtual machines and compact data centers. The Fog Server Manager oversees the management of processors and virtual machines (VMs) on the Fog Server. The client layer, located at the bottom, sends requests to nearby fog servers.

To allocate appropriate tasks to Virtual Machines (VMs), the Whale Optimization Algorithm (WOA) is employed. The program efficiently identifies underutilized and over utilized VMs, leading to the deactivation of underused VMs, resulting in substantial energy conservation.

Static and dynamic task scheduling algorithms are the two types of scheduling systems. The fog node is used in static scheduling to make all task-related information available. Since all task-related information is disseminated in dynamic task scheduling [7], the fog node cannot access all of it. Connecting them to the cloud layer for further data handling allows them to take advantage of their computational capabilities. The deployment of cloud data centres, cloud data processing, and data carriers are supported by the cloud stratum, which is the topmost level. This makes it possible to offer enormous data storage space, powerful

computing power, and processing capabilities for a variety of data [9]. All of the tasks are shown combined and in process in the picture below. [13]

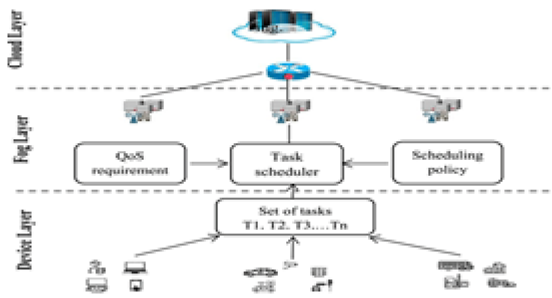

Fig 2. Task Scheduling in Fog Computing

2.2. Hybrid Heuristic Algorithm [8]

In the ever-evolving landscape of optimization research, traditional algorithms have often fallen short in tackling complex real-world problems efficiently. However, a new dawn has arrived with the emergence of Hybrid Heuristic Algorithms. This revolutionary approach combines the strengths of multiple heuristic techniques, culminating in an unparalleled level of problem-solving capabilities. Breaking boundaries and defying limitations, Hybrid Heuristic Algorithms harness the collective intelligence of diverse methodologies, leading to solutions that were once deemed unattainable. As researchers delve into this unexplored territory, they uncover a perfect blend of creativity and precision, unveiling innovative pathways to tackle the most intricate challenges. Join us on this captivating journey as we reveal the cutting-edge advancements and transformative potential of Hybrid Heuristic Algorithms, reshaping the future of optimization research as we know it.

2.3 Energy Efficiency in Fog Computing

The total energy consumed by the system is referred to as Energy Consumption. This energy is utilized by some of the grid elements, such as fog, transducer, access, and so on. The calculation of this energy expenditure can be done using a mathematical expression. in where EC is Energy consumption, CT means time taken by task executing, LUT is Utilization time by energy, HP means power consumption by host. [12] In fog computing while calculating the total energy if we have to calculated then we examine the total energy used where we compute the all task will be completed [15]. In our proposed algorithm we also calculated the total energy used while completed the all the task in the various host.

Energy = EC + (CT - LUT) *HP

$$\text{Total Energy Consume Cost} = \sum^{n} [\text{Current cost} + (CT - LUT) * HP]$$

3. Proposed Methodology

In the proposed methodology, we outline a task scheduling approach that focuses on optimizing energy consumption in Fog Computing. Firstly, we consider the characteristics of each task, such as computation requirements and communication demands, to determine the most suitable Fog node for its execution. To enhance energy efficiency, we employ a dynamic scheduling algorithm that takes into account real-time priorities and resource availability. This algorithm intelligently adjusts task assignments based on the current system state and energy constraints.

Furthermore, we incorporate energy-aware decision-making techniques that consider the energy consumption profiles of Fog nodes. By selecting energy-efficient nodes for task execution, we aim to minimize overall energy usage in the system.

To evaluate the effectiveness of our proposed methodology, we conduct experiments and simulations using realistic workload scenarios. These evaluations provide insights into the energy savings achieved through our approach and demonstrate its superiority over existing scheduling techniques. Overall, our proposed methodology offers a practical and efficient solution for energy optimization in Fog Computing, contributing to the sustainability and cost-effectiveness of Fog-based systems.

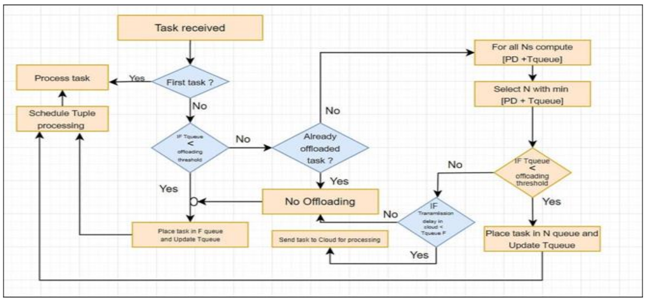

Fig 3. – Proposed Work Flow

3.1 Proposed Algorithm

CSj: Request "j" undergoing Cloud server processing using an algorithm.

FSj: Request "j" being processed by Fog server utilizing an algorithm.

FSM: Each Fog Server (FS) has a Finite State Machine (FSM).

Reqj: Request from user "j" is efficiently executed utilizing an algorithm.

SCDT: delay time determined through computational methodology.

DTj: Permissible latency for request "j" assessed using computational techniques.

TTj: Threshold value for each task.

T: Total task t1, t2, t3, ..., tn Subtask of the total task

Pre-emptive Task Priority Scheduling Algorithm

Step 1. for all reqj in FS & in priority Queue {

// Task assignment for each request (reqj) in FS and priorityQueue

}

1.reqjSent= sendRequestToNearestFogServer(reqj, nearestFogServer);, nearestFogServer);

2. The Fog server calculates the delay time Tj using the equation:

Tj = (SCDT * DTj) / (TTj - reqj)

3. Verify the task's time constraint:

if (((STest(T) / C) + STest,j) > delayj(T)),

Reject req

4.Alternatively, enqueue the request in the priority queue using the Priority Assignment Module

(Prj) = Priority(delayj(T), STest(T))

if (Prj == R)

Place reqj in QR

else if (Prj == N)

Place reqj in QN

else if (Prj == O)

Place reqj in Q

Process Request in FS or in Cloud

Service provisioning based on request priority and subscription category employing a priority-based algorithm.

1.For x = P, Q, R:

2. while (Qz! = NULL) {

3. for each (request in Qz):

4. if (Eq(4) holds for currentFS) {

// request in the currentFS

}

5. if (!Eq(4) holds for current FS && fog Layer Resources Available) {

}

6. if (!Eq(4) holds for current FS && !fog Layer Resources Available && cloud Resources Available) {

4. Performance Evaluation & Result Discussion

In this section we are analysis the discuss the various results, and also compare the exiting algorithm with our proposed algorithm, we have implement all the simulator work in iFogSim [16]. Here in table I have mention describe the various properties and tools specification. Researchers are enabled by the iFog Simulator to create virtual fog computing environments, where various scenarios and configurations can be simulated to evaluate the system's efficiency, resource allocation, communication protocols, and overall performance.

Table 2. Requirement Specification

Tools Requirement	***Specification***
Operating System	Windows 11
Storage	8 GB RAM
Simulator	iFog
Programming Language Used	Java
IDE	NetBeans

As here I have mention snapshot of implementation work in Fig.1 mention the task allocation status of FCFS algorithm for exiting.

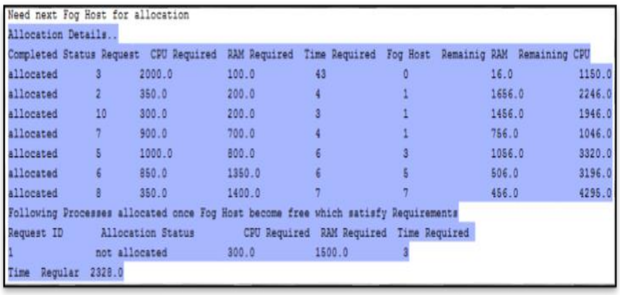

Fig 4. - Task allocation status in FCFC Algorithm

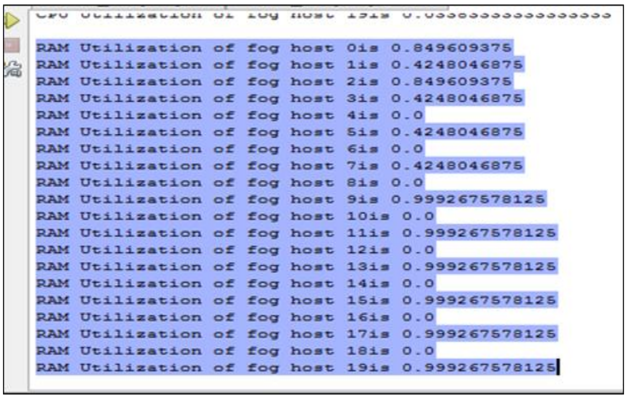

Fig 5. – Total RAM Utilization in task Scheduling

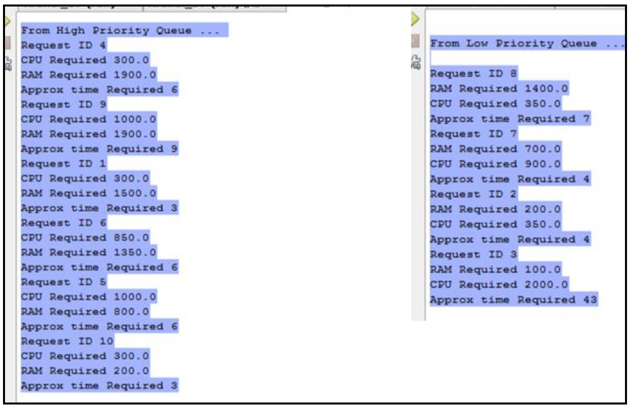

Fig 6. – High/Low Priority Request in Task Scheduling

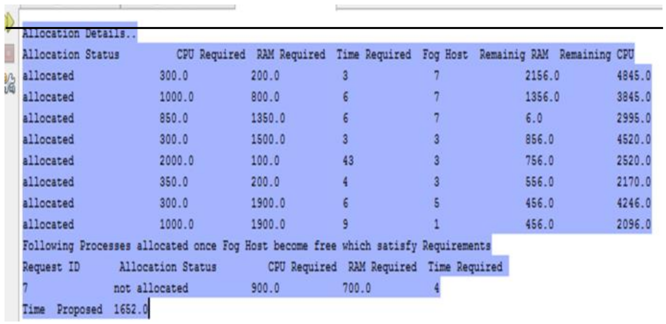

Fig 7. – Proposed algorithm status while allocation in Task Scheduling

4.1 Performance Analysis & Compression

Table 3. Time taken with respect to number of requests

Virtual Machine-50 Hosts - 30	**Time (in ms)**	
No. of User Requests	Simple priority based allocation (spba)	priority based allocation using load comfort index (pbalci)
5	**663**	**431**
10	**938**	**659**

15	**1121**	**824**
20	**1540**	**1239**
25	**1842**	**1474**
30	**2120**	**1811**

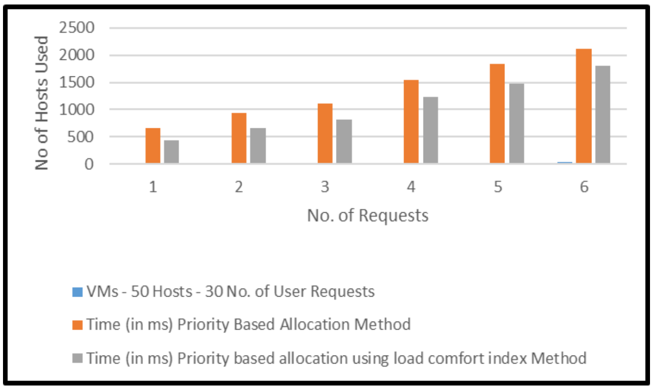

Fig 8. Compression between SPBA & PBALCI respect to time

Table 4. Number of hosts used with user requests

Virtual Machine-50 Hosts - 30	**No. of Hosts Used**	
No. of User Requests	Simple priority based allocation (spba)	priority based allocation using load comfort index (pbalci)
5	**2**	**1**
10	**4**	**3**
15	**4**	**3**
20	**5**	**5**
25	**6**	**5**
30	**7**	**6**

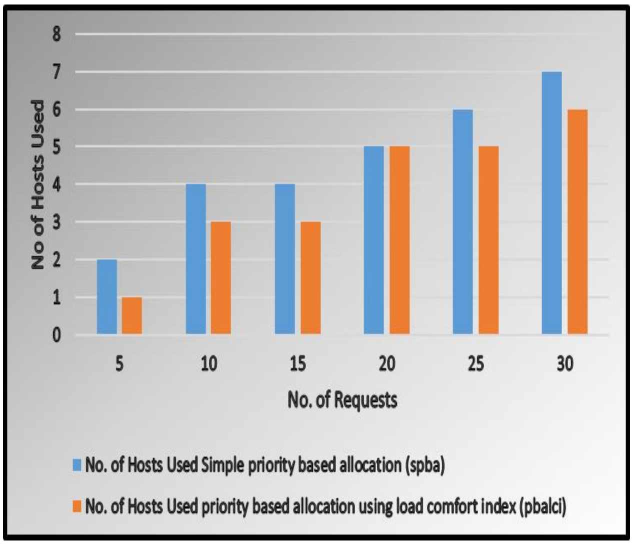

Fig 9. Compression between SPBA & PBALCI respect to host

5. Conclusion

Mission panning set of well-defined steps play a critical role and have a substantial impact on the effectiveness and calibre of cloud-fog data processing amenities. These algorithms enhance inventory exploitation, reduce support slowness. And increase network effectiveness. Moreover, the ability to enable task execution in a distinct environment strengthens the security of fog computing systems. In addition, work scheduling algorithm development is an ongoing process, recently steps being created annual progress the previous ones. Universal task programming procedure performs a significant contribution to architecture of cloud-fog information processing, and their proper application can Enhancing the Efficacy and Reliability of Fog Computing Services through Intelligent Resource Allocation and Task Scheduling. When checking, the number of nodes used, the amount of time needed for execution, and other pertinent factors are taken into account.

Acknowledgements

This research was supported by my Guide Dr. Kirit Modi,and my DPC member Dr.Chirag Patel, we are thankful to my colleagues who provided expertise that greatly assisted the research, although they may not agree with all of the interpretations provided in this paper.

Author contributions

In this Research article, I, Hardik Patel have done the research under the guidance of Dr. Kirit Modi, identifying the Gap between the Cloud Computing and Fog Computing, the Main issue and challenges of the optimized the energy along with the execution time. Moreover, we have tried to minimize the number of host used.

Conflicts of Interest

Manuscript Title: Dynamic Resource Allocation for Real-Time Task Scheduling in Cloud-Fog Computing: A Cost-Effective and Low-Latency Approach

The authors whose names are listed immediately below certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Author Names:

1 Hardik Mahendrabhai Patel

1PhD Scholar, Sankalchand Patel University, Visnagar, Gujarat, India.

2. Dr. Kirit J. Modi

[2]Professor, Department of Computer/Information Technology Department, Sankalchand Patel University, Visnagar, Gujarat, India

References

[1] SHUDONG WANG et al." Task Scheduling Algorithm Based on Improved Firework Algorithm in Fog Computing" | IEEE Journals & Magazine IEEE Xplore. https://ieeexplore.ieee.org/document/899824W.-K. Chen, Linear Networks and Systems. Belmont, CA, USA: Wadsworth, 1993, pp. 123–135.

[2] Fulong Xu et al." Adaptive scheduling strategy of fog computing tasks with different priority for intelligent production lines" 0th International Conference of Information and Communication Technology. 10.1016/j.procs.2021.02.064

[3] Asmaa shoker et al." Resource Allocation Strategy in Fog Computing: Task Scheduling in Fog Computing Systems" An International Journal of Communication Sciences and Information Technology 2022.

[4] Mohammad Reza Alizadeh et al ." Task scheduling approaches in fog computing: A systematic review" Int J Commun Syst. 2020; e4583. Wiley online library.com /journal/dac © 2020 John Wiley & Sons, Ltd.

[5] Navjeet Kaur et al." A systematic review on task scheduling in Fog computing: Taxonomy, tools, challenges, and future directions" Concurrency Computat Pract Exper. 2021; e6432. wileyonlinelibrary.com/journal/cpe ©2021 John Wiley & Sons, Ltd.

[6] Guevara, Judy C., and Nelson L. S. da Fonseca. "Task Scheduling in Cloud-fog Computing Systems - Peer-to-Peer Networking and Applications." *SpringerLink*, 18 Jan. 2021, https://doi.org/10.1007/s12083-020-01051-9.

[7] Pejman Hosseinioun et al. "aTask Scheduling Approaches in Fog Computing: A Survey. Transactions on Emerging Telecommunications Technologies /https://onlinelibrary.wiley.com/doi/epdf/10.1002/ett.3792.

[8] Wang, Juan, and Di Li. "Task Scheduling Based on a Hybrid Heuristic Algorithm for Smart Production Line With Fog Computing." Sensors, vol. 19, no. 5, MDPI AG, Feb. 2019, p. 1023. Crossref, https://doi.org/10.3390/s19051023.

[9] Madhura, R., et al. "An Improved List-based Task Scheduling Algorithm for Fog Computing Environment." Computing, vol. 103, no. 7, Springer Science and Business Media LLC, Mar. 2021, pp. 1353–89. Crossref, https://doi.org/10.1007/s00607-021-00935-9.

[10] Kaur, Navjeet, et al. "TRAP: Task-resource Adaptive Pairing for Efficient Scheduling in Fog Computing." Cluster Computing, vol. 25, no. 6, Springer Science and Business Media LLC, July 2022, pp. 4257–73. Crossref, https://doi.org/10.1007/s10586-022-03641-z.

[11] Nguyen, Binh Minh, et al. "Evolutionary Algorithms to Optimize Task Scheduling Problem for the IoT Based Bag-of-Tasks Application in Cloud–Fog Computing Environment." Applied Sciences, vol. 9, no. 9, MDPI AG, Apr. 2019, p. 1730. Crossref, https://doi.org/10.3390/app9091730.

[12] Jaeyalakshmi, M. "Task Scheduling Using Meta-Heuristic Optimization Techniques in Cloud Environment." International Journal of Engineering and Computer Science, Valley International, Nov. 2016. Crossref, https://doi.org/10.18535/ijecs/v5i11.59.

[13] Ghobaei-Arani, Mostafa, et al. "An Efficient Task Scheduling Approach Using Moth-flame Optimization Algorithm for Cyber-Physical System Applications in Fog Computing." *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 2, Wiley, Oct. 2019. *Crossref*, https://doi.org/10.1002/ett.3770.

[14] Hossain, Md Razon, et al. "A Scheduling-based Dynamic Fog Computing Framework for Augmenting Resource Utilization." *Simulation Modelling Practice and Theory*, vol. 111, Elsevier BV, Sept. 2021, p. 102336. *Crossref*,

https://doi.org/10.1016/j.simpat.2021.102336

[15] Jamil, Bushra, et al. “A Job Scheduling Algorithm for Delay and Performance Optimization in Fog Computing.” Concurrency and Computation: Practice and Experience, vol. 32, no. 7, Wiley, Nov. 2019. Crossref, https://doi.org/10.1002/cpe.5581.

[16] Das, Resul, and Muhammad Muhammad Inuwa. “A Review on Fog Computing: Issues, Characteristics, Challenges, and Potential Applications.” *Telematics and Informatics Reports*, vol. 10, Elsevier BV, June 2023, p. 100049. *Crossref*, https://doi.org/10.1016/j.teler.2023.100049.

[17] Saleh, Safa’a S., et al. “iFogRep: An Intelligent Consistent Approach for Replication and Placement of IoT Based on Fog Computing.” Egyptian Informatics Journal, vol. 24, no. 2, Elsevier BV, July 2023, pp. 327–39. Crossref, https://doi.org/10.1016/j.eij.2023.05.003.

[18] Sandvik, Jens-Petter, et al. “Evidence in the Fog – Triage in Fog Computing Systems.” *Forensic Science International: Digital Investigation*, vol. 44, Elsevier BV, Mar. 2023, p. 301506. *Crossref*, https://doi.org/10.1016/j.fsidi.2023.301506.

[19] Zhao S, Yang Y, Shao Z, Yang X, Qian H, Wang CX. FEMOS: fog-enabled multitier operations scheduling in dynamic wireless networks.

[20] Wan J, Chen B, Wang S, Xia M, Li D, Liu C. “Fog computing for energy-aware load balancing and scheduling in smart factory”. IEEE Trans Industrial Informatics. 2018;14(10):4548-4556.

[21] Yang Y, Zhao S, Zhang W, Chen Y, Luo X, Wang J. DEBTS: “delay energy balanced task scheduling in homogeneous fog networks”. IEEE Internet Things J. 2018;5(3):2094-2106.

[22] Sharma S, Saini H.”A novel four-tier architecture for delay aware scheduling and load balancing in fog environment ”Sustain Comp:Info Syst.2019:24:100355.

[23] Gad-Elrab AA, Noaman AY. “A two-tier bipartite graph task allocation approach based on fuzzy clustering in cloud-fog environment”. Future Gen Comp Syst. 2020; 103:79-90.

[24] Mastoi, Qurat-ul-ain, et al. “A Novel Cost-Efficient Framework for Critical Heartbeat Task Scheduling Using the Internet of Medical Things in a Fog Cloud System.” *Sensors*, vol. 20, no. 2, MDPI AG, Jan. 2020, p. 441. *Crossref*, https://doi.org/10.3390/s20020441.

[25] Suleiman, Husam. “A Cost-Aware Framework for QoS-Based and Energy-Efficient Scheduling in Cloud–Fog Computing.” *Future Internet*, vol. 14, no. 11, MDPI AG, Nov. 2022, p. 333. *Crossref*, https://doi.org/10.3390/fi14110333.

[26] Abdelmoneem, Randa M., et al. “Mobility-aware Task Scheduling in cloud-Fog IoT-based Healthcare Architectures.” *Computer Networks*, vol. 179, Elsevier BV, Oct. 2020, p. 107348. *Crossref*, https://doi.org/10.1016/j.comnet.2020.107348.

[27] Hardik M.Patel, Dr.Kirit Modi,” Optimizing Resource Utilization and Response Time in Cloud-Fog Computing through Task Scheduling Algorithm” Eur. Chem. Bull. 2023, 12(Special Issue 10), 556 – 566.

[28] Prof. Vaishali Sarangpure. (2018). Hybrid Hand-off Scheme for Performance Improvisation of Wireless Networks. International Journal of New Practices in Management and Engineering, 7(03), 08 - 14. https://doi.org/10.17762/ijnpme.v7i03.67

[29] Rajesh, P. ., & Kavitha, R. . (2023). An Imperceptible Method to Monitor Human Activity by Using Sensor Data with CNN and Bi-directional LSTM. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 96–105. https://doi.org/10.17762/ijritcc.v11i2s.6033

[30] Sharma, R., & Dhabliya, D. (2019). Attacks on transport layer and multi-layer attacks on manet. International Journal of Control and Automation, 12(6 Special Issue), 5-11. Retrieved from www.scopus.com