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

Job Offloading Techniques for Increasing Mobile Cloud Computing's Energy Efficiency

Nitinkumar Rajnikant Pandya [1]*, Dr. Ankit Shah [2]

Submitted: 10/09/2022 **Accepted:** 20/12/2022

Abstract: Mobile Cloud Computing (MCC), a new phase in cloud computing technologies, combines the cloud with mobile networks. MCC puts a strong emphasis on quality of service (QoS) and maximizes the utilization of the mobile network's resources to increase network efficiency. The great computational power and scalability of the cloud are utilized by MCC. It also eliminates geographical and temporal limitations, enabling work offloading for mobile users. There is now a wide range of research on mobile cloud computing that aims to use the prevalence of mobile devices for processing in order to get beyond restrictions in computational capability, storage capacity, and internal battery life. In this research, we investigated several job allocation methods for MCC's battery usage and energy efficiency, and we presented a more effective model that takes Service Level Agreement into account (SLA). In summary, computational offloading is a potential strategy for reducing execution time (redeemable energy) and enhancing battery life in mobile devices.

Keywords: *Job offloading, Data Center, Energy Efficiency, Service Level Agreements (SLA), and Mobile Cloud Computing*

1. INTRODUCTION

The demand for computationally demanding and power-hungry mobile apps grows along with the general use of smart devices like smartphones, tablets, IoT-based devices, and other multimedia devices, creating an unheard-of global spike in mobile data traffic [1].

The differences between mobile devices with low resources and apps that demand a lot of resources are still hard to repair, even when processors or new computing technologies are increasing day by day with technological innovation [1]. The battery life of mobile devices is reduced when completion times and energy usage rise, and as a result, service quality diminishes.

The primary method of computation used in mobile cloud computing is a form of compute offloading that allows mobile users to send mobile cloud servers time- and resource-intensive computations. The user can choose to offload part or all of their data processing to cloud servers rather than utilizing their mobile device, and the cloud will subsequently deliver the results to the mobile device.

The efficiency of leisure time as well as work productivity is constantly improving thanks to the usage of a variety of applications to make mobile devices more powerful and portable [2]. The processing limitations of mobile devices are being addressed by a number of ongoing efforts on MCC, storage size, and battery life by leveraging their popularity.

[1] *Research Scholar, Sankalchand Patel University, Visnagar, Gujarat, India*
ORCID ID: 0000-0002- 7724- 9862
[2.] *Associate Professor, Department of Artificial Intelligence &Data Science, Faculty of Engineering and Technology, Sankalchand Patel University, Visnagar, Gujarat, India.*
ORCID ID: 0000- 0002- 4651- 4651
** Corresponding Author Email: nitin.pandya85@gmail.com*

MCC has attracted a lot of interest and grown significantly due to the rise of mobile devices. Transferring mobile apps to the cloud from constrained mobile devices is required [3]. Mobile consumers will gain additional application options, a great user experience, greater battery life, powerful and reliable computing, and expanded storage space, among other advantages, as a result.

The processing and storage capacities of mobile devices are among their many drawbacks. In a distant cloud, the wide geographic distribution of processing resources would cause significant network delays and expensive service prices. The fundamental technology employed by MCC, which would be based on a distinct unloading approach, has the potential to convert a sizable portion of the work that will be sent for the computation. Figure 1 shows the MCC's basic architecture, which consists of four key elements: Mobile devices, Wi-Fi or wireless access points (APs), the internet, cloud resources, including data centres, and those four are listed in that order.

The Wi-Fi or Wireless Access Point (AP), communicates with the Internet via Bss or Access Points to connect mobile devices (APs) including some SLA Parameters. The cloud Resources including the data center is a resource pool generated by a reasonably big server that performs user Cloud services like computation and storage are made available in accordance with the controller's instructions. a brain linked to the server providing the mobile network service receives the initial requests and data from mobile users. The cloud will receive these requests over the internet. [1] [3] [5].

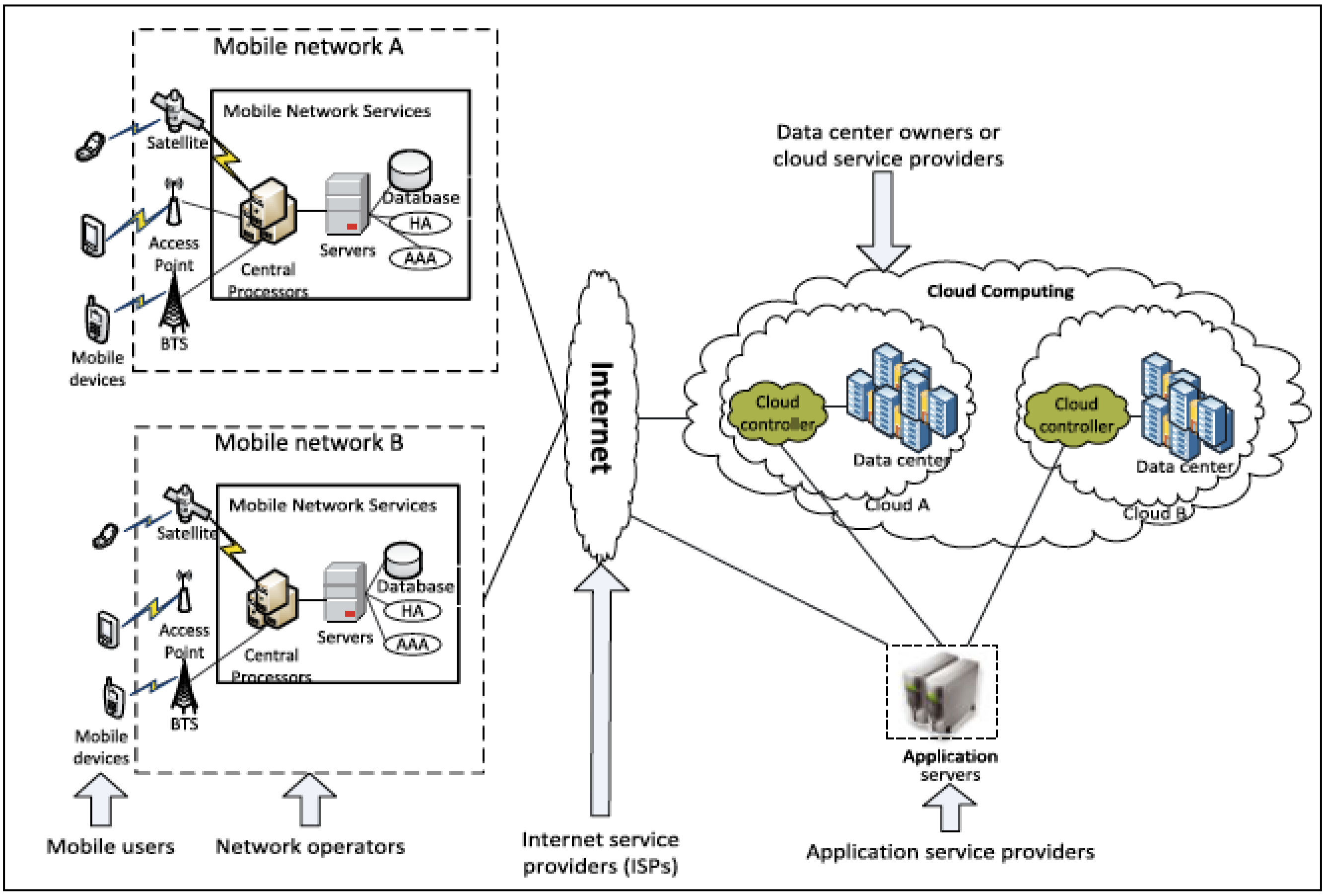

Fig 1. Basic Architecture of Mobile Cloud Computing System

1.1 THE PAPER'S ORGANIZATION

The remainder of this article is structured as follows: In the second section, we discuss the advantages and uses of mobile cloud computing. We give a brief introduction to mobile cloud computing in the third part. with Different job offloading algorithms in MCC. In the fourth section we have discussed our proposed algorithm for Proposed Energy Efficiency Algorithm using SLA constraints. Finally, we conclude our work and some open Research area for future works are identified.

2. MOTIVATION

The major goals of scheduling tasks in mobile cloud computing are to reduce execution time, resource usage, cost, and energy usage. To achieve QoS and enhance resource utilization, several research projects on work scheduling are now underway in MCC. The bulk of these research use different evolutionary algorithms to reduce task execution time or cost, however they do not ensure SLA adherence.

When dealing with various applications in the MCC environment, such as NLP issues, GPS, video optimization, face recognition, image processing, crowd computing, social media, etc., which demand high computational power, offloading computing task to a remote server can accelerate computing performance.

The MCC is beginning to see the value of computation offloading as a way to speed up task execution and extend the battery performance of mobile devices. We have covered a number of current work offloading techniques in the following section.

3. JOB OFFLOADING METHODS IN MCC

The computational cleverness algorithm may solve a number of challenging problems as an optimization strategy in industrial production and academic research such as the industries or research areas like genetic algorithms (GA), evolutionary programming, ant colony optimization (ACO) [1]. Job scheduling in MCC has generally throughout the MCC era, attention was placed on reducing processing time, cost, resource use, and energy consumption.

To give mobile users speedy access to improved processing, storage, and reliability services, a close-by resource-rich architecture is situated. The primary issues with mobile devices were attempted to be resolved in earlier MCC task scheduling studies by conducting parallel processes on local and cloud servers rather than only the mobile device [1].

3.1 Whale Optimization Algorithm (WOA) [1]

In task scheduling, experimental approaches to optimization problems have become more prevalent, and the effectiveness of optimization varies with the difficulty. There are several advantages to using these methods. The multi-objective optimization issue is solved using this approach.

A distinctive swarm optimization technique called the WOA imitates the predation behavior of humpback whales. Finding the optimum solution replaces the predation process used by humpback whales.

3.2 Job Allocation Mechanism (JAM) [2]
The JAM is made up of a primary device that splits tasks and distributes them to connected mobile devices, users, and devices that carry out duties after getting work requests from smartphone management. The primary device regularly checks performance measures like CPU and RAM as well as the connection status of client devices in order to manage job requests.

JAM allows the primary device to add and remove the devices. When more mobile devices are connected, the main device collects and provides resource data to the linked client devices, such as CPU, RAM, and battery level. The main device is reset when exceptions occur due to incapacity to serve and other issues and faults, since the information from the client devices attached to the main device is shared.

3.3 Mixed Integer Linear Programming (MILP) [4]
To optimize cloudlet selection and computing resource allocation, a MILP method is suggested. They look at how to pick the optimal cloudlet from a wide range of possibilities from the perspective of the service provider and how to correctly distribute resources to satisfy customer needs.

In the case of several cloudlets, a two-stage method for the distribution of resources. The plan can determine if the mobile phone service request has to be performed on the cloudlet or on the remote cloud server.

3.4 Successful Cloudlet Selection Offloading Policy (Eceso) [5]
To reduce overall energy consumption, this algorithm will allocate the best cloudlet to the end user depending on available network and system resources. They term it a Binary Integer Programming (BIP) problem and suggest a decomposition technique based on distributed linear relaxation heuristics.

The local level unloading choices are used to address the AP's offloading subproblem and affect users who really are linked to the same access point (AP).

The global level offloading decision ensures that the local level offloading decision level's offloading solution complies with the cloudlet resource limits.

The different offloading techniques or algo. in MCC is summarize as following table 1.

Table 1: Summary of Different Offloading Techniques

Sr. No	Reference	Goal of the Paper	Outcome	Methods
1	Peng, Hua, et al. 2019	to optimization the job completion time and energy consumption with this algorithm.	Outstanding efficiency and operating cost performance, allowing for practical solutions to MCC.	Whale Optimization Algorithm (WOA)
2	Yi, Gangman, et al. 2018	to minimise the amount of battery used when processing large amounts of data in MCC using JAM.	Comparing JAM to the allocation that did not include battery consumption rate, work processing performance was enhanced.	Job Allocation Mechanism (JAM)
3	Liu, Li, and Qi Fan. 2018	Optimize resource distribution while satisfying user needs in a multi-cloudlet scenario.	The suggested method works better in terms of resource use, system reward, and access latency.	Mixed Integer Linear Programming (MILP)
4	Mazouzi, Houssemeddine, et al. 2018	to reduce the total amount of energy used by customers.	Performance enhancements in terms of energy consumption in comparison to current offloading techniques	ECESO Offloading Heuristic
5	Liu, Li, et al. 2019	in MCC, a survey is offered to classify four major challenges for computation offloading.	A different taxonomy of concerns discovered in this field is offered, as well as solutions to these issues.	NA

4. Proposed Work

There are some challenges, we found in the existing algorithm for Energy Efficiency in MCC. Few of them are pointed here as below,

- Power/Battery Consumption due to wireless network traffic
- Job scheduling algorithm with delay due to network Connectivity/Availability
- Remote Process Execution Time
- Violating the QoS (including SLA)
- Distance between smartphone and wi-fi router

The Energy Efficiency algorithms also include some SLA Parameters which should be applicable on 3 different level of MCC like,

- SLA for IAAS
- SLA for PAAS
- SLA for SAAS

As per the proposed work meet to the requirement of Energy Efficiency, we will work on the parameters related to SLA on Platform as a service (PAAS). Some of the Parameters related to PAAS are as below,

- Integrity
- Scalability
- Pay as you go
- Number of developers required

When we will work in the MCC environment, for the Energy Efficiency using SLA parameters in PAAS, we find some challenging factors which affect to achieve the SLA violation minimizing,

- Network Speed
- Wireless Connectivity
- Bandwidth
- Delay
- Scalability through Wi-fi Network
- Utilization

From the above-mentioned factors, the working methodology of the existing algorithm for job allocation mechanism using Energy Efficiency and Minimizing the SLA Violation, our proposed work is divided into two main parts, as shown in figure 2

1. **Energy efficient job allocation for minimizing battery consumption**
2. **Minimizing the SLA Violation**

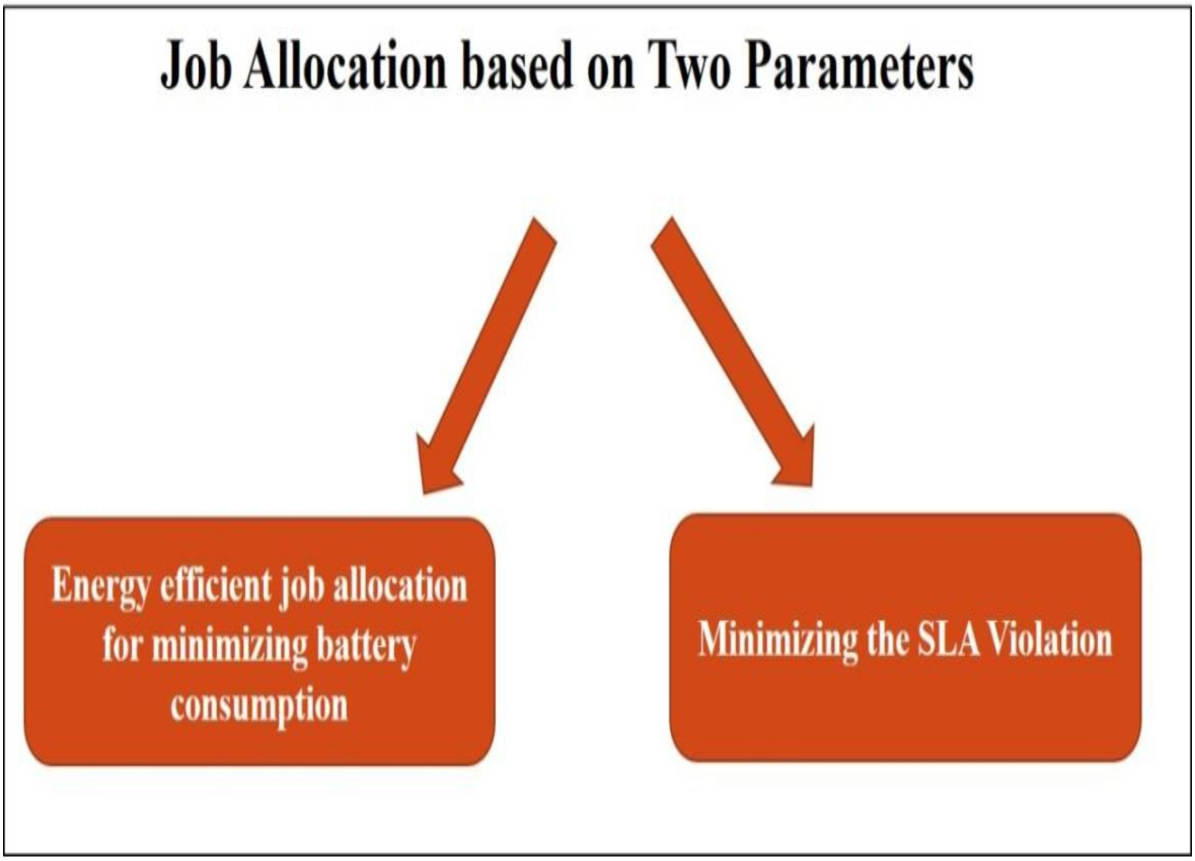

Fig 2. Proposed Work of job allocation Mechanism has divided in two main tasks

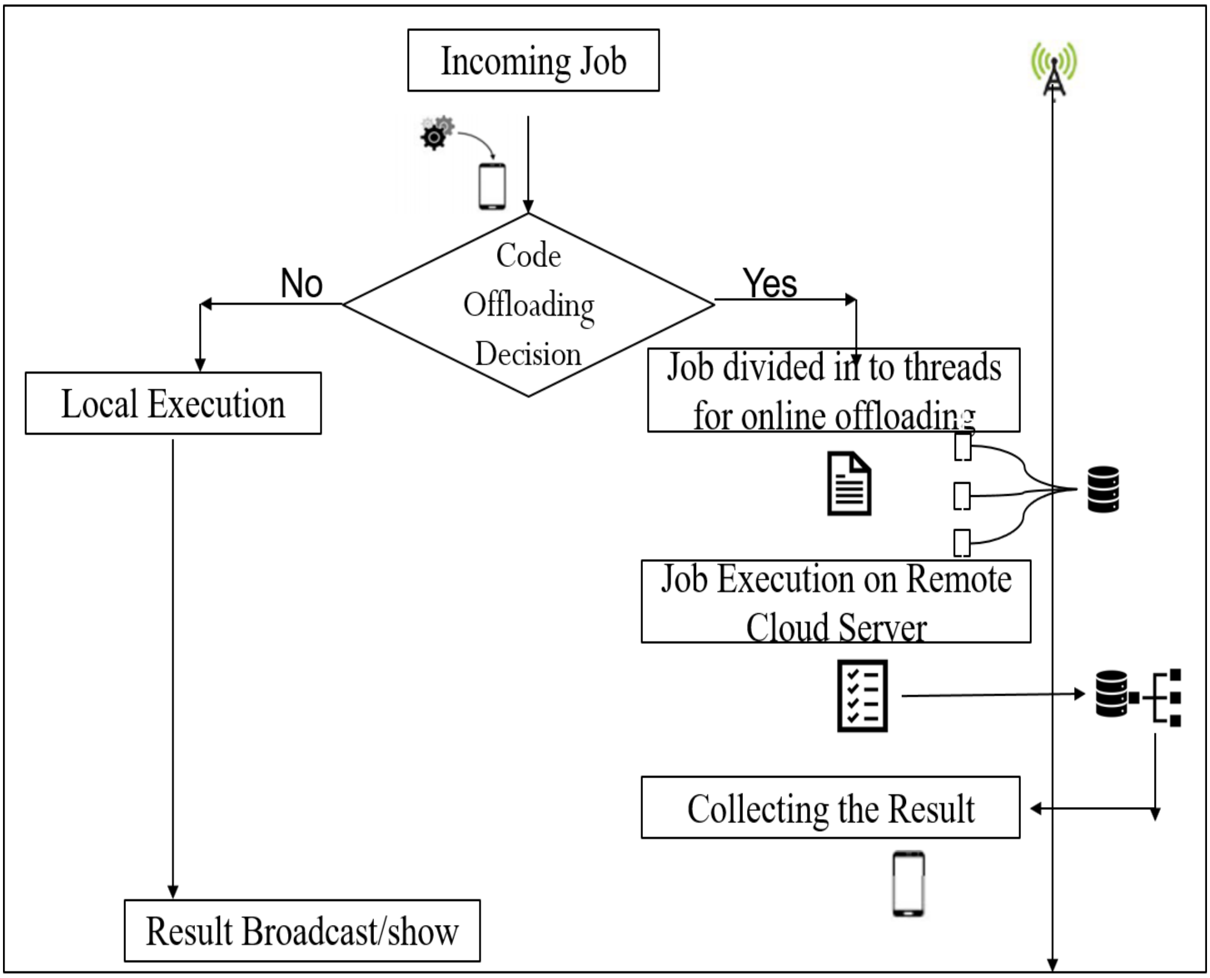

Fig 3. Proposed Novel Job Allocation Methodology in MCC

4.1 Proposed Energy Efficiency Algorithm using SLA constraints

Some basic Notation and Equation for Energy Consumption.
C- Computation task requires to execute the instructions
M - Speed of the instruction per second of Mobile Devices using SLA Constraints
S- Speeds in instructions per second of the cloud server using SLA Constraints
D - Data exchanged between cloud server and Mobile Devices
T - Transmission rate of the MD via wireless network,
E_c- Energy Consumed by Mobile Devices for Computing,
E_i- Energy Consumed by Mobile Devices while being idle,
E_{tr}- Energy Consumed by Mobile Devices for Transmission(send/receive) the data
Basic Energy Consumption Equation is expressed by,

$$\mathbf{E = \{(E_c*(C/M) \; * (C/S)*(D/T))-(E_i+ E_{tr})\}} \qquad \textbf{(Eq. - 1)}$$

Proposed Algorithm:

Input: *Set of Mobile Devices (users) MDi ,*
Wi-Fi Access point AP,
Mobile Cloud Server C_r
Output: *Minimize the Energy Consumption using Mobile Cloud Offloading*

1. job *waitlist L, n* ← 0, or ∀*m* ∈MD_i *using SLA, through wi-fi AP*
2. Check/decide if the job will be executed on local MD or Mobile Cloud Server C_r including SLA Constraints
3. If the job will be executed on MD locally then,
   Measure the Energy/Power Consumption using SLA Constraint using Eq-1.
4. If the job will be executed on Mobile Cloud Server using Wi-Fi Network then,
   Do the job offloading online and divide the task in the n number of thread (not many threads), and execute it on a single Cloud Server.
   we will measure the Energy /Power Consumption using SLA Constraint using Eq-1.
5. Compare with the Existing Algorithm with the Energy Consumption and SLA minimization.

This Proposed algorithm is explained in the Figure -3. When the incoming job is come for the scheduling then it will first, we have to decide that the code is being offload or not?

If it is yes then do the job offloading online and divide the task in the n number of thread (not many threads), and execute it on a single Cloud Server. we will measure the Energy /Power Consumption using SLA Constraint using Eq-1.

If the job will be executed on MD locally then, measure the Energy/Power Consumption using SLA Constraint using Eq-1.

5. Conclusion & Future Work

We found a number of issues with computational offloading in the MCC environment. Offloading calculations is becoming a possible method for extending the battery life of mobile devices and lowering execution time (energy savings). When moving computationally demanding activities to the cloud in the MCC environment with SLA in mind, computation offloading is a crucial issue to take into account. Our proposed algorithm efficiently offloads the application without violating SLA.

In future work, we will test our algorithm with the standard computation-intensive jobs and monitor how effective is our proposed approach compared to the existing solutions.

Acknowledgments

This research was supported by my Guide Dr. Ankit Shah, 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, Nitin Kumar Pandya have done the research under the guidance of Dr. Ankit shah, identifying the Gap between the Cloud Computing and Mobile Cloud Computing, the Main issue and challenges of the Energy Efficiency in Mobile Cloud Computing (Mobile Devices), create the novel algorithm for it implement it. Moreover, we have tried to minimize the SLA violation in the terms of PAAS level in the cloud (not in simulator)

Conflicts of Interest

Manuscript Title: Job Offloading Techniques for Increasing Mobile Cloud Computing's Energy Efficiency

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. Nitinkumar Rajnikant Pandya

[1] Ph.D. Scholar, Sankalchand Patel University, Visnagar, Gujarat, India

2. Dr. Ankit Shah

[2]Associate Professor, Department of Artificial Intelligence & Data Science, Faculty of Engineering and Technology, Sankalchand Patel University, Visnagar, Gujarat, India.

References

[1]. Zhang, Feifei, et al. "Online learning offloading framework for heterogeneous mobile edge computing system." Journal of Parallel and Distributed Computing 128 (2019): 167-183.

[2]. Li, Chunlin, et al. "Stochastic computation resource allocation for mobile edge computing powered by wireless energy transfer." Ad Hoc Networks 93 (2019): 101897.

[3]. Alaria, S. K., A. . Raj, V. Sharma, and V. Kumar. "Simulation and Analysis of Hand Gesture Recognition for Indian Sign Language Using CNN". International Journal on Recent and Innovation Trends in Computing and Communication, vol. 10, no. 4, Apr. 2022, pp. 10-14, doi:10.17762/ijritcc.v10i4.5556.

[4]. Mazouzi, Houssemeddine, NadjibAchir, and Khaled Boussetta. "Maximizing Mobiles Energy Saving Through Tasks Optimal Offloading Placement in two-tier Cloud." Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems. ACM, 2018.

[5]. Baccarelli, Enzo, Michele Scarpiniti, and Alireza Momenzadeh. "EcoMobiFog–Design and Dynamic Optimization of a 5G Mobile-Fog-Cloud Multi-Tier Ecosystem for the Real-Time Distributed Execution of Stream Applications." IEEE Access 7 (2019): 55565-55608.

[6]. Gupta, D. J. . (2022). A Study on Various Cloud Computing Technologies, Implementation Process, Categories and Application Use in Organisation. International Journal on Future Revolution in Computer Science &Amp; Communication Engineering, 8(1), 09–12. https://doi.org/10.17762/ijfrcsce.v8i1.2064

[7]. Zhang, Yongmin, et al. "Efficient Computation Resource Management in Mobile Edge-Cloud Computing." IEEE Internet of Things Journal (2018).

[8]. Liu, Li, et al. "A survey on computation offloading in the mobile cloud computing environment." International Journal of Computer Applications in Technology 59.2 (2019): 106-113.

[9]. Wu, Linlin, and Rajkumar Buyya. "Service level agreement (SLA) in utility computing systems." Performance and dependability in service computing: Concepts, techniques and research directions. IGI Global, 2012. 1-25.

[10]. Kose, O., & Oktay, T. (2022). Hexarotor Yaw Flight Control with SPSA, PID Algorithm and Morphing. International Journal of Intelligent Systems and Applications in Engineering, 10(2), 216–221. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/1879

[11]. Peng, Hua, et al. "Joint optimization method for task scheduling time and energy consumption in Mobile Cloud Computing environment." *Applied Soft Computing* 80 (2019): 534-545.

[12]. Zhou, Bowen, and Rajkumar Buyya. "A group-based fault tolerant mechanism for heterogeneous mobile clouds." *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. 2017.

[13]. Linda R. Musser. (2020). Older Engineering Books are Open Educational Resources. Journal of Online Engineering Education, 11(2), 08–10. Retrieved from http://onlineengineeringeducation.com/index.php/joee/article/view/41

[14]. Tang, Wenyi, et al. "A novel task allocation algorithm in mobile crowdsensing with spatial privacy preservation." Wireless Communications and Mobile Computing 2019 (2019).

[15]. Wang, Tongxiang, et al. "Efficient multi-tasks scheduling algorithm in mobile cloud computing with time constraints." Peer-to-Peer Networking and Applications 11.4 (2018): 793-80