# MTSA: Multiple Tasks Scheduling Algorithm based on Server Efficiency in Mobile Edge Computing

**S.M. Muthukumari[1†] and Dr. E. George Dharma Prakash Raj[2††],**

Bharathidasan University, Trichy, Tamilnadu, India

**Summary**
Mobile Edge Computing (MEC) may be a computing procedure in which the highlights of Cloud Computing are being expanded to the edge of systems through versatile base stations. In Mobile Cloud Computing (MCC), a few data numbers are given by portable clients, and after that data is exchanged to the inaccessible cloud to render an extra handle all through the Web. Concurring to the over strategy, a few delays happen due to the enormous distance. In this paper, how to allot a few tasks to the server at a time and figure out the need of the approaching request utilizing the Ageing based Task scheduling (ATSA) algorithm conjointly define the server effectiveness of the framework to handle the errands. Based on the above-expressed approach, the assignments can be planned to the server and get distant better; a much better; a higher; a stronger; an improved a distant for better result.

***Key words:***
*ATSA algorithm, Cloud Computing, Mobile Cloud Computing, Mobile Edge Computing, SESA algorithm.*

## 1. Introduction

Cloud Computing may be an advanced computing engineering built on the premise of parallel computing, conveyed computing, cluster computing, lattice computing between all the over capacities calculation mode frequently has the taking after focal points: on- request, compelling, exact and moo- fetched. [1]. Cloud capacity basically offers uncertainly beneficial instruments for preparing, putting away, and a portable gadgets are confined in memory, capacity, processor, etc. This shows an assortment of programs that can work on portable gadgets. [2]. In standard Mobile Cloud Computing (MCC), an expansive volume of data is created by portable clients information is sent to the inaccessible cloud for assist preparing through the Web. As a result of this strategy, it expands to a long period, which makes a tall Conclusion to Conclusion idleness. [3].

Mobile Edge Computing (MEC) takes computational assets closer to versatile gadgets. In Portable Edge Computing, demands arrive from a few versatile gadgets, and exercises are submitted to a computer that's closest to a portable unit. The errand execution is much speedier than ordinary cloud computing technologies. Mobile Edge Computing acknowledges ask from edge clients and sends an ask to the MEC server, the MEC framework gets an ask from the client and requires the Cloud Data Center to extend the execution of the benefit or something else improve the unwavering quality and accessibility of versatile applications.

In this paper an effective task scheduling algorithm has been proposed called as Multiple tasks scheduling algorithm based on server efficiency for Mobile Edge Computing which reduces the overall execution time of the tasks. Every incoming task are independent in their attributes and also priorities.

This paper has been coordinated as follows: Section II. Deals with Related Work followed by Proposed Algorithm in Section III. Section IV describes the Experimental and Analysis with tables and graphs. Section V Concludes the paper.

## 2. Related work

Task Scheduling Based on priority and Resource Allocation in Multi-User Multi- Task Mobile Edge Computing System proposed by Pouria paymard, Natel Mokari et.al., [4] in this paper, the authors studies resource allocation for a Multi-User Multi-Task (MUMT) MEC system and proposed a priority based task scheduling policy to improve the profit of the mobile operator because each task has different priority levels. Sequential Task Scheduling for Mobile Edge Computing using Genetic Algorithm which was proposed by Ahmed A, A Dobre et.al., [5] the objective of this paper is to minimize both latency and offloading failure of the task. And the authors considered sequential task offloading to multiple- edge computing servers to providing ultra-reliable low- latency mobile edge computing.

Multi-user Multi-task Computation Offloading in Green Mobile Edge Cloud Computing proposed by Weiwei Chen, Dong Wang et.al., [6] in the paper, the authors considered computation offloading at the mobile edge cloud with the set of Wireless Devices (WDs) and

Manuscript received January 5, 2022
Manuscript revised January 20, 2022
https://doi.org/10.22937/IJCSNS.2022.22.1.99

calculated how much energy harvested from Wireless Devices and how to schedule the task into servers.

A Deadline- Constrained Multi-Objective Task Scheduling Algorithm in Mobile Cloud Environments written by Li Liu, Raj Kumar Buyya et.al., [7] in this paper the authors concentrated with both unconstrained and time deadline constrained cases, the task scheduling is modeled as a multi-objective optimization problem. Completion Time Minimization in Multi- User Task Scheduling with Heterogeneous Processors and Budget Constraints by Sowndarya Sundar, Jaya Prakash Champati et.al., [8] the aim of this paper is to identified a task scheduling decision that minimizes the weighted sum completion time of all tasks.

Multi- Objective Tasks Scheduling Algorithm for Cloud Computing Throughput Optimization proposed by Atul Vikas lakra and Dharmendra Kumar yadav, [9] in this paper, they proposed a multi- objective task scheduling algorithm form mapping tasks to a virtual machines in order to improve the throughput of the datacenter and reduce the cost without violating the SLA (Service Level Agreement) for an application in cloud SaaS environment. Age Based Task Scheduling and Computation Offloading in Mobile- Edge Computing Systems proposed by Xianxin Song, Xiaoqi Qin et.al., [10] in this paper, the authors workout with single MEC server and one mobile device running several applications and proposed a light- weight task scheduling and computation offloading algorithm. Efficient multi- tasks scheduling algorithm in mobile cloud computing with time constraints proposed by Tongxiang Wang, Xianglin Wei et.al, [11] in this paper, the authors investigated the MCC- assisted execution of multi- tasks scheduling problem in hybrid MCC- architecture. First they formulated an optimization problem. Second generated the Cooperative Multi- tasks Scheduling based on Ant Colony Optimization algorithm (CMSACO) for tackle the problem.

Multi- Task Scheduling Based on Classification in Mobile Edge Computing was proposed by Xiao Zheng, Yuanfang Chen et.al., [12] in this paper, a dynamic multi task scheduling prototype was proposed to improve the limited resource utilization in the vehicular networks (VNET) assisted by Mobile Edge Computing (MEC).

## 3. Proposed Algorithm

The most issue is to get a task from different workflows, at that time task recognizable proof takes a longer time to verify the errand and the execution time doesn't come to the optimized esteem. In the priority scheduling algorithm most of the time high priority tasks continuously get the opportunity for execution. Whereas a long time, frequently low priority tasks have a chance to execute, indeed in the event that high need tasks keep coming at that point low need task is at that point skipped and the CPU moved high priority task to execute and this contributes execution time get increment and throughput get diminish.

In this proposed algorithm MTSA, to begin with, task recognizable proof has been made which distinguishes all approaching tasks from the versatile clients and recognizes what are the resources that need to process the task. At that point calculate the inexact time utilization to require the execution of the assignment. And recognizes the attributes of the offloaded errand based on the properties the server has been chosen to execute the task.

At that point settle the time limitations to execute the assignment in the event that the time gets crossed the constraint consequently task go back and frame the line. At last, calculate the waiting time of the assignment within the queue. Based on the task holding up within the line need has been calculated based on the Age of the task. After that most elevated need task once more go to execution. In this strategy, low need task moreover gets a chance to execute.

**Input:** Receives number of task from mobile user n
Queue length l
Priority of the task r
Number of Servers s
**Output:**
Execution of all incoming task
1. Task from mobile user
2. Queue formed by the incoming task
3. Request send to Data Center
4. Data Center assigns task to server
5. Schedule the task based on priority
    r=0;
    while r<n do
    r++;
    end while
    while r++<n do
6. if server is available to process task then
    Server ← n (assign the task to server)
7. else if server= busy then
    a. Task wait in the queue for processing
    b. Calculate the waiting time of the task in queue
        i. Identify the length of the queue l
        ii. Repeat step5 for identify the priority of the task
8. Stop the process

## 4. Experimentation and Analysis

The Experimental Analysis for the proposed algorithm is done utilizing the Workflow Test system that progresses the current CloudSim test system by offering the next level of workflow management. The proposed MTSA planning algorithm is compared with the existing SESA algorithm. The execution measurements are Add up to Completion Time, Execution Time, and Throughput.

### 4.1. Total Completion Time

Within the existing SESA algorithm [13], the task can be sent to the server for execution without any time imperatives at that time single errand planning calculation confront a few issues in the execution and completion stage it is not able to reach optimize arrangement. In the MTSA algorithm partition, the assignments into a few workloads based on the server efficiency and settles the time limitations for execution and completion of the errand and similarly separate the task into a virtual machine. Sometime recently doing out the task check the accessibility of the assets for way better completion. As appeared in figure1, the proposed MTSA algorithm accomplishes the superior result when compared with the existing SESA algorithm.

**Table 1:** Performance of Total Completion Time

| Number of Tasks | Total Completion Time (sec) | |
|---|---|---|
| | SESA | MTSA |
| 50 | 52 | 51 |
| 100 | 69 | 58 |
| 150 | 62 | 59 |
| 200 | 77 | 74 |

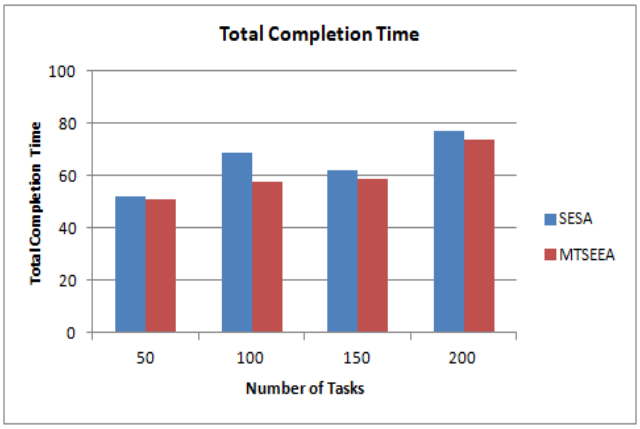


**Fig 1:** Total Completion Time

### 4.2 Execution Time

The proposed MTSA algorithm gives better Execution Time compare with the existing SESA calculation. The MTSA algorithm diminishes the task holding up inside the line which leads to prevalent execution time. Figure 2, shows up the Execution time of the proposed MTSA algorithm.

**Table 2:** Performance of Execution Time

| Number of Tasks | Total Completion Time (sec) | |
|---|---|---|
| | SESA | MTSA |
| 50 | 42 | 37 |
| 100 | 54 | 51 |
| 150 | 60 | 59 |
| 200 | 67 | 65 |

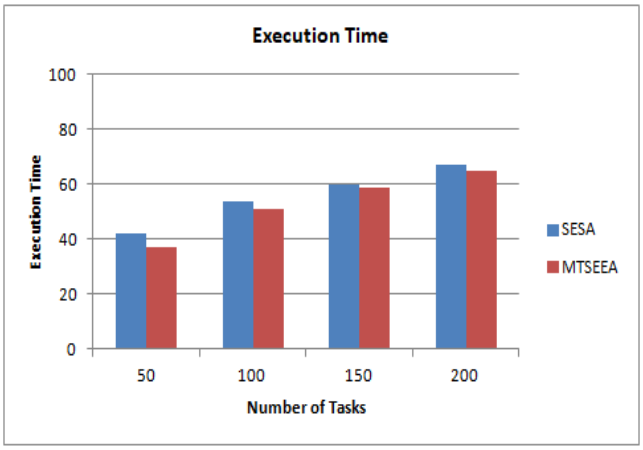


**Fig 2:** Execution Time

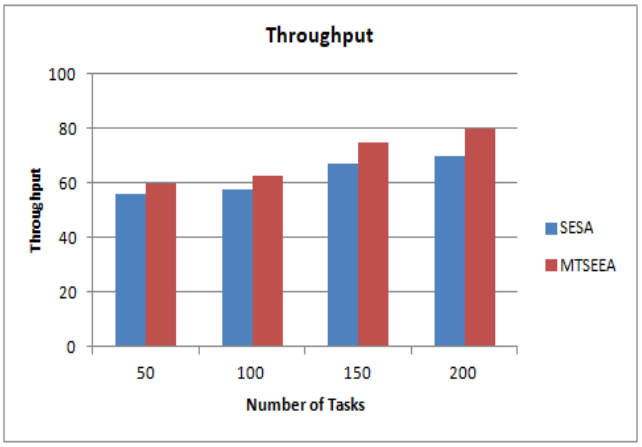


**Fig 3:** Throughput

### 4.3 Throughput

In Execution and completion, the task is essentially isolated into servers based on the required levels. All high priority and low priority tasks are essentially shared with the available resources which subsequently lead to the expanded Throughput

**Table 3:** Throughput

| Number of Tasks | Total Completion Time (sec) | |
|---|---|---|
| | SESA | MTSA |
| 50 | 56 | 60 |
| 100 | 58 | 63 |
| 150 | 67 | 75 |
| 200 | 70 | 80 |

## References

[1] GE Junwei, YUAN Yongsheng., "*Research of cloud computing task scheduling algorithm based on improved genetic algorithm*", In: Applied Mechanics and Materials Volume: 347-350 pp 2426-2429 Trans Tech Publications, Switzerland (2013).

[2] Tiago Gama Rodrigues, katsuya Suto, Hiroki Nishiyama and Nei Kato., "*Hybrid Method for Minimizing Service Delay in Edge Cloud Computing through VM Migration and Transmission Power Control*", In: IEEE Transactions On Computers DOI: 10.1109/TC.2016.2620469 (2016).

[3] Dileep Kumar Sajnani, Abdul Rasheed Mahesar, Abdullah lakhan,Irfan Ali Jamali., "*Latency Aware and Service Delay With Task Scheduling in Mobile Edge Computing*", In: Communications and Network, 10, 127-141, http://doi.org/10.4236/cn.2018.104011 (2018).

[4] Pouriya Paymard, Nader Mokari, Mehdi Orooji., "*Task Scheduling Based on priority and Resource Allocation in Multi- User Multi- Task Mobile Edge Computing System*", In: 2019 IEEE 30th Annual International Symposium on personal, Indoor and Mobile Radio Communications (PIMRC): Track 3: Mobile and Wireless networks, DOI: 978-1-5386-8110-7/19 (2019).

[5] Ahmed A. AI- habob, Octavia A. Dobre and Garcia Armada., "*Sequential Task Scheduling for Mobile Edge Computing Using Genetic Algorithm*", In: IEEE DOI: 978-7281-0960-2/19 (2019).

[6] Weiwei Chen, Dong Wang and Keqin Li., "*Multi- user Multi- task Computation offloading in Green Mobile Edge Cloud Computing*", In: IEEE Transactions on Services Computing, DOI: 10.1109/TSC.2018.2826544 (2018).

[7] Li Liu, Qi Fan and Rajkumar Buyya., "*A Deadline-Constrained Multi- objective Task Scheduling Algorithm in Mobile Cloud Environments*", In: IEEE Access Volume 6, DOI: 10.1109/Access. 2018. 287015 (2018).

[8] Sowndarya Sundar, Jaya Prakash Champati and Ben Liang., "*Completion Time Minimization in Multi- user Task Scheduling with Heterogeneous processors and Budget Constraints"*, In: IEEE DOI: 978-1-5386-2542-2/18. (2018).

[9] Atul Vikas Lakra, Dharmendra Kumar yadav., "*Multi-Objective Tasks Scheduling Algorithm for Cloud Computing Throughput Optimization",* In: Elesvier, International Conference On Intelligent Computing, Communication & Convergence (ICCC- 2015), page no: 107-113, DOI: 10.1016/j.procs.2015.04.158. (2015).

[10] Xianxin Song, Xiaoqi Qin, yunzheng Tao, Baoling Liu and Ping Zhang., "*Age Based Task Scheduling and Computation Offloading in Mobile- Edge Computing Systems"*, In: IEEE 2nd Workshop on Intelligent Computing and Caching at the Network Edge, DOI: 978-1-7281-0922-0/19. (2019).

[11] 11. Tongxiang wang, Xianglin Wei, Chaogang Tang, Jianhua Fan., "*Efficient multi- tasks scheduling algorithm in mobile cloud computing with time constraints"*, In: Springer DOI: 10.1007/s12083-017-0561-9. (2017).

[12] 12. Xiao Zheng, Yuanfang Chen, Muhammad Alam and Jun Guo., "*Multi- Task Scheduling Based on Classification in Mobile Edge Computing"*, In: MDPI Electronics, DOI: 10.3390/electronics8090938. (2019).

[13] S.M.Muthukumari and Dr. E. George Dharma Prakash Raj., "*Server Efficiency Based Scheduling Algorithm for Mobile Edge Computing"*, In: International Journal of Advanced Science and Technology, Volume 29. No. 5s, (2020), pp. 446-453.