

## A hybrid approach for scheduling applications in cloud computing environment

Ahmed Subhi Abdalkafor<sup>1</sup>, Khattab M. Ali Alheeti<sup>2</sup>

<sup>1</sup>Career Development Center, University of Anbar, Anbar, Iraq

<sup>2</sup>Computer Networking Systems Department, Computer Sciences and Information Technology College, University of Anbar, Anbar, Iraq,

---

### Article Info

#### Article history:

Received Jun 28, 2019

Revised Oct 9, 2019

Accepted Oct 15, 2019

---

#### Keywords:

Cloud computing  
Task scheduling algorithm  
priority

---

### ABSTRACT

Cloud computing plays an important role in our daily life. It has direct and positive impact on share and update data, knowledge, storage and scientific resources between various regions. Cloud computing performance heavily based on job scheduling algorithms that are utilized for queue waiting in modern scientific applications. The researchers are considered cloud computing a popular platform for new enforcements. These scheduling algorithms help in design efficient queue lists in cloud as well as they play vital role in reducing waiting for processing time in cloud computing. A novel job scheduling is proposed in this paper to enhance performance of cloud computing and reduce delay time in queue waiting for jobs. The proposed algorithm tries to avoid some significant challenges that throttle from developing applications of cloud computing. However, a smart scheduling technique is proposed in our paper to improve performance processing in cloud applications. Our experimental result of the proposed job scheduling algorithm shows that the proposed schemes possess outstanding enhancing rates with a reduction in waiting time for jobs in queue list.

Copyright © 2020 Institute of Advanced Engineering and Science.  
All rights reserved.

---

### Corresponding Author:

Ahmed Subhi Abdalkafor,  
Career Development Center,  
University of Anbar, Anbar, Iraq.  
Email: [ahmed.abdalkafor@uoanbar.edu.iq](mailto:ahmed.abdalkafor@uoanbar.edu.iq)

---

## 1. INTRODUCTION

Nowadays, modern technology is witnessing a wide range of development in scientific applications and research areas. Cloud computing is considered one of the most important research fields that received a lot of attention from developers and designers. It is an innovative technology that uses central remote servers and internet networks to share information/applications and store important data to become available for users [1, 2]. Cloud computing allows consumers to share and access for daily applications/sensitive information without coordination and synchronization mode. These applications provide important data and control information to users at suitable time and without any delay [3, 4].

Cloud computing utilizes computing resources and network terminals to provide important data, information, and knowledge for users at the appropriate time. In other words, users can access the same resource at different times and place i.e. share data. The main aspect of cloud computing does not need pre-established communication for any request from resources. This aspect makes cloud computing occupy a large area of scientific research and modern applications in our daily life. The cloud computing permits access to several online services, social data, information and resources that can be utilized for exchange between computer devices on demand. In order to allocate computing resources quickly and efficiently, resource allocation tasks must be scheduled [5-8]. The basic structure of cloud computing is shown in Figure 1.

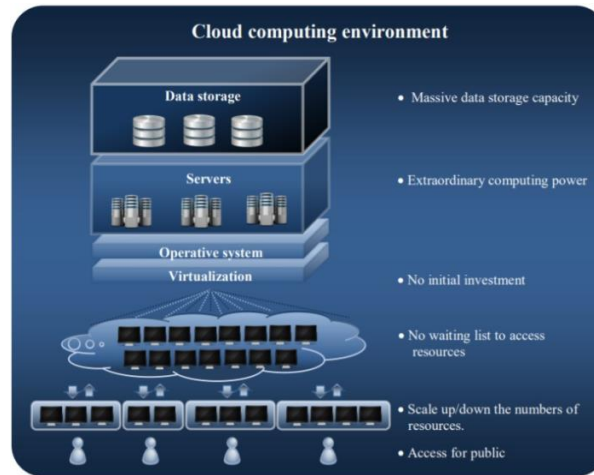


Figure 1. Basic structure of cloud computing [9]

According to Figure 1, data storage, servers, operative system, and virtualization interface are main components for any cloud computing system [9]. The difficulty of job scheduling requests is gradually increased in cloud system. In more detail, the current algorithms become paralyzed in front of the pool request of cloud web applications. As we know, scheduling in a cloud computing environment is most important issue in modern technology. The task scheduling consists of the following: workflow, static and dynamic cloud service scheduling, real-time scheduling, heuristic and opportunistic load balancing scheduling [10, 11]. The computing environment consists of a number of sources. Therefore, task scheduling is necessary technique to choose the best resources suitable for the implementation of various tasks. The efficient scheduling scheme can reduce response time, low consumption power, and provide availability for busy resources. In addition, scheduling methods have the ability to allocating computer machines with the least time to accomplish depending on their priorities, so scheduling tasks are essential and influential in this environment. For this reason, new scheduling algorithms are requested to overcome this problem.

In this paper, a novel scheduling algorithm is proposed to improve the performance of cloud computing. Moreover, it has the ability to reduce the amount of delay in queue waiting. The proposed algorithm in this research plays important role in allocating balance load between different resources. Therefore, the resource-scheduling in cloud computing is difficult tasks [10, 11].

## 2. RELATED WORK

Recently, several scheduling algorithms have been proposed to distribute cloud applications. All of these algorithms target to reduce waiting time of jobs in queue list, increase efficiency, decline number of repeat request and balance between available resources. In this paper, we proposed new hybrid scheduling algorithm to enhance performance of cloud computing. Sangwan *et al.* [12] proposed technique to improve round-robin scheduling algorithm in cloud computing environments by reducing the average of waiting time and total around time by changing the time quantum by keeping all the processes when arriving in CPU. It defines average of mean (TQ powerfully) after selecting the mean of burst time according to the number of processes. This proposed algorithm selects the first process then CPU is allocated for a temporary period according to the mean time quantum. There are many studies that have been worked on task scheduling. However, there is still a challenge for scheduling algorithms, such as minimum response time, maximum throughput, best possible resource utilization, and reducing the overload [13]. Mittal and Singh present [14] a proposed system to improve the algorithm for task scheduling in cloud computing. The authors can modify round-robin by adding a new strategy of the period for request time quantum. In more detail, the proposed algorithm obtains minimizes waiting time and maximize CPU usage. Kumar *et al.* [15] have tried to find the best slot of time quantum in round-robin algorithm to avoid the frequent context switch. Integer programming is utilized in create new CPU local scheduling named changeable time quantum (CTQ) to overcome too large/too little of the time quantum. Moreover, it has the ability to gives the best throughput rate of the system.

In [16], the authors proposed new scheme for reducing some of the drawbacks of round-robin algorithm as well as getting better performance of CPU at the same time. In other words, the modification of

algorithm plays important role in reducing average of waiting time, decline the amount of turnaround time and decreases the difficult of starvation. Nayak *et al.* [17] have improved the round-robin algorithm by organizing the processes in the ready queue. This modification can overcome the drawbacks of algorithm by set optimal time quantum and select the shortest burst time. In [18], a starvation optimizing algorithm is improved to fix one of the common problems that occur in shortest job first algorithm. This modification plays vital role in enhancing CPU and input/output resources requirements that have a direct and positive reflection on overall performance. It can increase throughput rate, reduce average waiting time and decline turnaround time in cloud environment. When resources are overloaded the algorithm has redirected all resources in virtual machines. The virtual machine in this system aims to get better response and processing time. Ghanbari *et al.* [5], have proposed a new algorithm based on the analytical hierarchy process named PJSC. It focuses on three matters which are consistency, complexity, and makes-pan. Consistency means adjusting fundamentals of comparison matrix based on scheduling of priority algorithm. When complexity is the number of jobs as well as it is calculating the priority vectors of evaluation matrixes. Agarwal *et al.* [19], have presented a general priority algorithm for effective implementation for resource allocation in cloud computing. It provides a better performance after applying in cloud sim toolkit environments and comparison with round-robin and first come first serves algorithms.

A novel hybrid scheduling algorithm is proposed in this paper to overcome common problems of the current scheduling algorithms. It plays a direct and positive impact on current methods via reducing waiting time, turnaround time and increasing performance of cloud computing system. This algorithm has the ability to establish balance for available resources. Thus, it will be providing sufficient transparency for cloud users in different regions. The objective of this paper is to propose a new algorithm to enhance performance of the round-robin algorithm via decreasing amount of waiting time and turnaround average time. This enchantment is achieved by adding priority field to arrival time of job processes whether equal or different.

### 3. CLOUD COMPUTING

The job scheduling algorithms in cloud computing are classified into two categories: batch mode heuristic algorithm (BMHA) and online mode heuristic algorithm (OMHA) [5]. The BMHA collects and queued in a set for arrive job processes at the waiting list. The time is fixed in this algorithm for jobs in cloud computing. First come first served algorithm is considered one of the applications of BMHA. In addition, round-robin, Max-Min, and Min-Min algorithms are also considered applications of BMHA. Whereas, OMHA fetches and processes jobs from queue waiting directly [8, 20, 21]. The BMHA is considered one of the most suitable algorithms for the cloud computing environment. In this approach, all arrived jobs apply one of the methods below [22]:

#### 3.1. First come first served (FCFS) 1

It is also named First in First Serves (FIFS) when the job process that arrived at queue waiting first will be served. The serves perform by order in ready queue, and they do not process short or large. The process in the queue must be waiting until all processes are complete. The processes waiting more time to terminate from ready queue, so the waiting and turnaround are quite high [23].

#### 3.2. Min-min algorithm

This algorithm determined all tasks and arranged it about large and small, the small tasks will be executed, firstly while the large task delays for a long time. The drawback of this algorithm is to assign the smaller processes to the resources with comparatively higher computational power [24].

#### 3.3. Max-min algorithm

This algorithm also depends on small and large tasks. However, it changes to the Min-Min algorithm whenever the task was too large. It will be executed as soon as and delay a small task for long-time [25].

#### 3.4. Round-robin algorithm

It is suitable for cloud computing applications. In addition, the most effective time-sharing systems, it's simplest and most widely used proportional share scheduling algorithm [11]. The average wait time (WT) and average turned around time (TAT) are head points on system, so round-robin designed to give some better responsive but the worst average output parameters (WT, TAT), it's not efficient enough to be employed in real-time systems [5].

### 3.5. Priority scheduling algorithm

In this algorithm, the FCFS processes scheduled is applied when jobs have equal-priority. It has a very important effect on performance of cloud computing in current applications.

### 3.6. The Shortest-Job-First (SJF) algorithm

It is a special case of the general priority scheduling algorithm [6], in computer science, queue priority as the kind of abstract data, which is like a queue regularly or structure of the data stacked but where additionally each element has a "priority" associated with it. This algorithm's work when each process has high priority is executed before a low priority process. If two processes have the same priority, they are executed according to their order in the queue [26]. The common types of scheduling algorithms that utilized in cloud computing is shown in Figure 2.

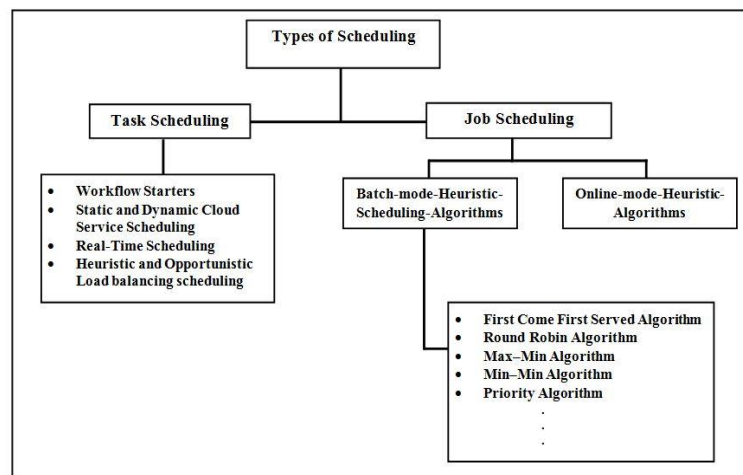


Figure 2. Categories of scheduling algorithm

Workflows technique is utilized in this paper to determine the optimal distribution of their job processes to available resources whether computational network. Application structures in workflows stage are analyzed to select appropriate computational resources for tasks. In more detail, this step based on resource availability and application characteristics for each task. Generate a satisfactory solution in slot time for tasks in cloud computing is mainly based on workflow schedulers. The workflow technique in cloud computing is shown in Figure 3.

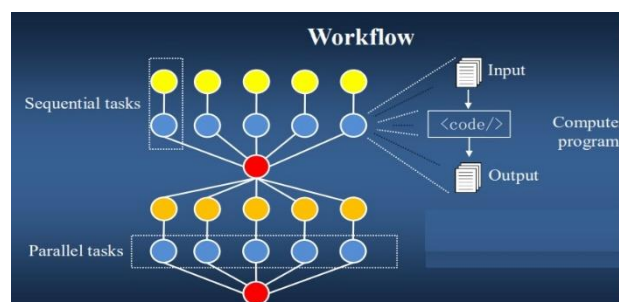


Figure 3. Workflow scheme in cloud computing [9]

The input and output data files are associated with each computer program represented by nodes. The role of workflow schedulers easily notices with cloud computing in modern applications. In more detail, Figure 4 demonstrates interaction process of scheduler with workflow and cloud resources. The interaction process of scheduling engine is represented in three layers as shown in Figure 4 which are workflow, scheduler, and cloud resources. Three steps are accomplished in layer two which are task analysis, objectives optimization, and task distribution. Allocated of cloud resource in layer three heavily depend on layer two.

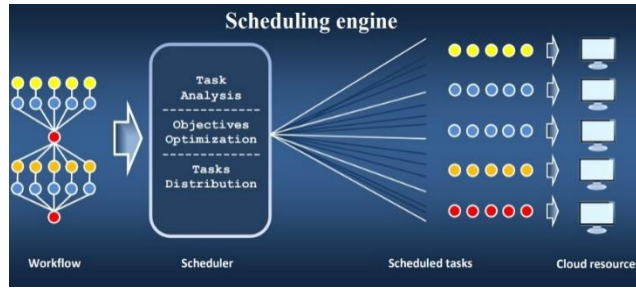


Figure 4. Scheduler interaction with cloud resources and workflow [9]

**4. THE PROPOSED ALGORITHM**

A novel scheduling algorithm is proposed in this paper to fix common problems of the traditional round-robin algorithm employed in cloud computing. In more detail, it uses first come first serve algorithm to schedule all arrived processes in queue waiting with fixed time quantum. However, waiting time and turnaround time are criteria utilized in core work of the proposed algorithm. This algorithm is enhanced in this paper by re-arranging processes via slapped priority filed. When processes arrive at ready queue, the highest priority is allocated to the lowest process value of burst time. In this case, all arrived processes at ready queue will yield to re-arranging technique. However, the new algorithm plays important role in enhancing overall performance and throughput rate of cloud computing. Thus, it reduces the amount of waiting time and turnaround time. The flowchart of the proposed algorithm is shown in Figure 5.

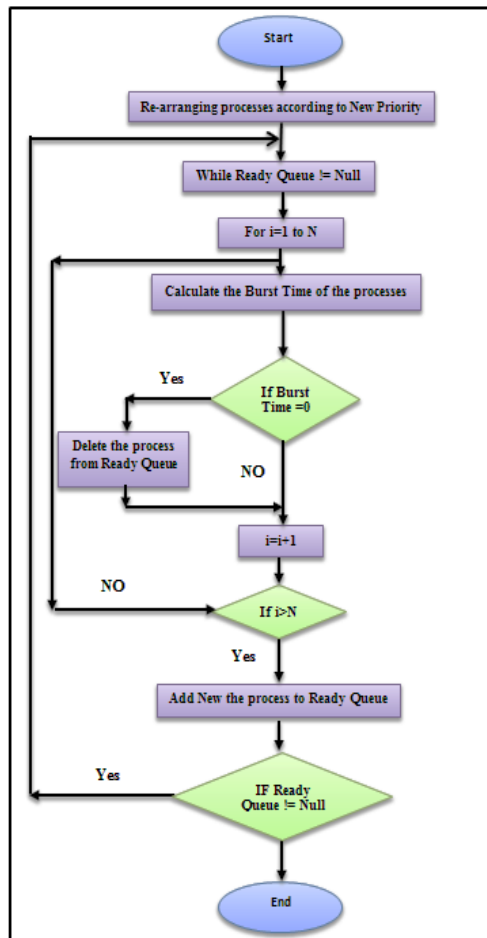


Figure 5. Block diagram of proposed algorithm

The principle work of the proposed scheduling technique is shown in the algorithm.

---

Algorithm 1 the Proposed Scheme

---

Input: N, BT, AT, Priority, TQ

Output: Gantt Chart, AWT, ATAT

- 1: Arrange all the processes in RQ in order of their highest priority to the lowest value of BT.
  - 2: While RQ is not empty
  - 3: For I=1 to N
  - 4: Calculate the BT of the processes;
  - 5: Delete the process from RQ if the BT=0;
  - 6: If the new process arrived RQ go to Step 2;
  - 7: Draw Gantt Chart and Calculate AWT, ATAT;
  - 8: End while.
- 

The steps of the proposed algorithm are listed in Algorithm1. The algorithm starts with the random initialization of tasks:

**Step 1:** Allocate CPU to every process in the round-robin approach, new priorities are assigned to all arrived processed according to give the highest priority to the lowest value of burst time with the same time quantum time. Then each process gets the control of the CPU according to the new priorities based on the remaining CPU bursts in case the ready queue is not reached to the null

**Step 2:** After finishing the first step, calculate the burst time of all processes and re-arranged the remaining burst time with the new priorities until the value of a burst time reached zero.

## 5. SIMULATION RESULT AND DISCUSSION

To evaluate the performance of the proposed system, various job processes are applied to the traditional round-robin algorithm and the proposed algorithm. All these scenarios are employed under environment of virtual box of net framework. In more detail, five of jobs processes which are P1, P2, P3, P4, p5 are integrated with CPU burst time. These values of burst time are 25, 30, 87, 13, 20 respectively. In this paper, various priority values are utilized which are 2, 3, 4, 0, 1 respectively. The fixed time quantum in this proposal is 20ms with equal and different arrival time in ready queue as shown in Table 1.

Table 1. Processes specifications

Process Name	Arrival Time	Burst Time	Priority
P <sub>1</sub>	0	25	2
P <sub>2</sub>	0	30	3
P <sub>3</sub>	0	87	4
P <sub>4</sub>	0	13	0
P <sub>5</sub>	0	20	1

The performance metrics are considered a very important factor in evaluating the proposed system. These metrics, such as turnaround time and waiting time [14]. Average of turnaround time (TAT) can be calculated as:

$$\text{Average ATA} = \sum (T_i - \text{AT of } P_i) / N \quad (1)$$

Average of waiting time (WT) can be calculated as:

$$\text{Average WT} = \sum (\text{TAT of } P_i - \text{BT of } P_i) / N \quad (2)$$

In this paper, two scenarios are applied to test the efficiency of the proposed system.

On one hand, we assume all problem arrival at the same time in scenario I. In this case, the traditional round-robin is built on distribution of the time quantum among the current processes i.e. the algorithm handles all processes without feature of the priority. The processes distribution of traditional round-robin algorithm is shown in Figure 6. Average of WT: 74.4ms; Average of TAT: 109.4ms.

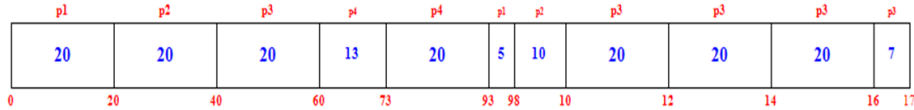


Figure 6. Gantt chart of traditional Round-robin algorithm

On the other hand, to evaluate the performance of the proposed algorithm, we applied to the same scenario of the previous jobs. In addition, the highest priority value attached to the lowest value of burst time respectively. Adding process heavily depends on arrival time of jobs at ready queue that have the same TQ. The Gantt chart of the proposed algorithm is shown Figure 7. Average of WT: 58.4ms; Average of TAT: 93.4ms.

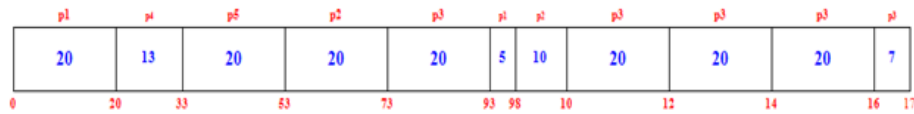


Figure 7. Gantt chart of proposed algorithm

However, Figure 8 shows the performance comparison of jobs at ready queues that have the same arrival time. The unites of saved time for the average of WT and TAT after applying the proposed algorithm that is shown in Table 2. Whereas, the process will have repeated in scenario II again to confirm efficiently of the proposed scheduling algorithm. In this case, we assume that jobs at a ready queue with various arrival times. Table 3 shows process specifications for five jobs in the queue.

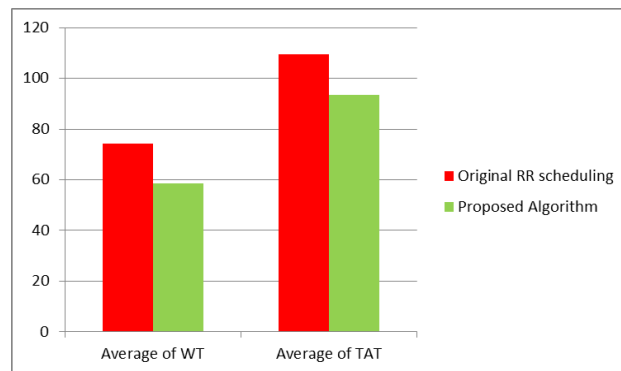


Figure 8. Performance comparison for scenario

Table 2. Results for Scenario I

Performance Attribute	Traditional Round-robin Scheduling	Proposed Algorithm	Remark
Average Waiting Time	74.4ms	58.4ms	16 Unites of Saved Time
Average Turnaround Time	109.4ms	93.3ms	16.1 Unites Saved Time

Table 3. Processes Specifications

Process Name	Arrival Time	Burst Time	Priority
P <sub>1</sub>	1	25	2
P <sub>2</sub>	3	30	3
P <sub>3</sub>	5	87	4
P <sub>4</sub>	2	13	0
P <sub>5</sub>	0	20	1

The number of jobs with their slot time for traditional round-robin is shown in Figure 9. Average of WT: 53.6ms; Average of TAT: 88.6ms. The number of jobs with their slot time for the proposed algorithm is shown in Figure 10. Whereas, Average of WT: 49.6ms; Average of TAT: 84.6ms. Performance comparison in various arrival times is shown in Figure 11. To discuss the proposed algorithm, we need to compare our algorithm with traditional to calculate the efficient rate of the algorithm. According to the result, we can easily notice that the proposed algorithm more efficient from the traditional algorithm. Our proposal is applied with the previous algorithm [12, 14-17] to measure its efficiency.

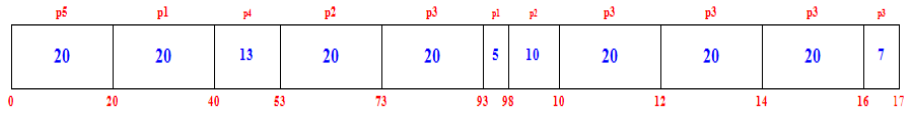


Figure 9. Gantt chart for round-robin algorithm

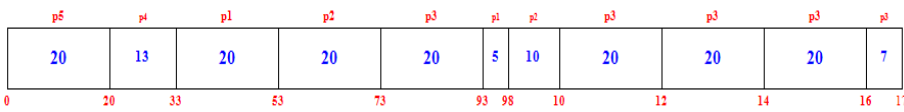


Figure 10. Gantt chart of the proposed algorithm

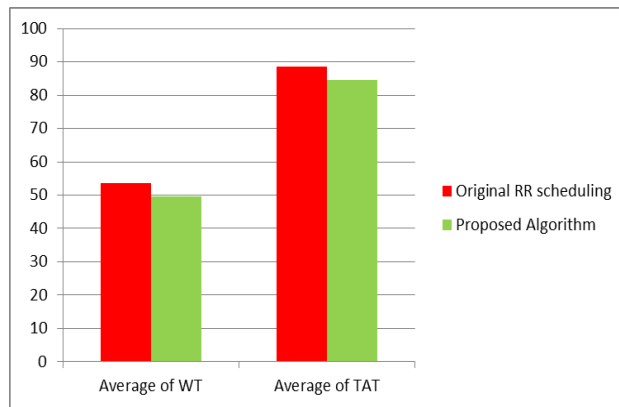


Figure 11. Performance comparison for scenario II

a. Improved round-robin scheduling in cloud computing [12]

Some performance metrics are utilised to evaluate efficacy of new scheduling method for clouding computing that proposed [12]. Thus, Figure 12 shows improve rate of round-robin. Average of WT: 42.4ms; Average of TAT: 62.4ms.

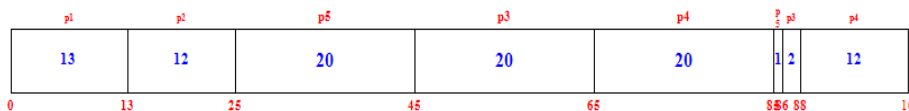


Figure 12. Gantt chart of improved round-robin scheduling

b. Enhanced round-robin [14]

In Figure 13, average of WT and average of TAT are calculated to measure improving rate of the round-robin [14]. Average of WT: 64.2ms; Average of TAT: 112.4ms.



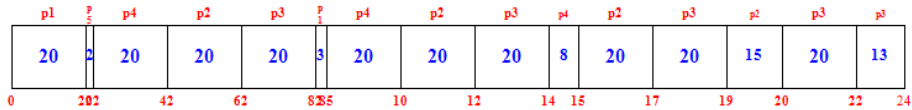


Figure13. Gantt chart of enhanced round-robin

c. Integer programming [15]

In addition, average of WT and average of TAT are required to test integer programming for the scheduling method that shown in Figure 14 [15]. Average of WT: 36ms; Average of TAT: 84.6ms.

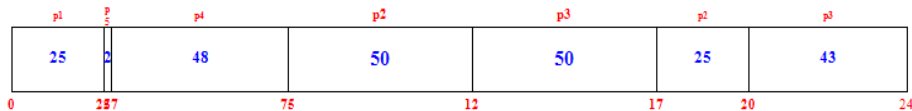


Figure 14. Gantt chart of integer programming robin algorithm

d. Improved round-robin algorithm [17]

Nayak et al, have proposed scheduling algorithm to improve performance of round-robin method. However, figure 15 shows efficiency of this algorithm. Average of WT: 84ms; Average of TAT: 141.6ms.

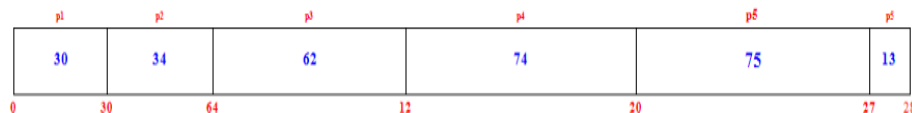


Figure 15. Gantt chart of improved round-robin algorithm

e. A priority-based round-robin [16]

Rajput et al [16], have presented some improvement process on round-robin algorithm as shown in Figure 16. Average of WT: 24.4ms; Average of TAT: 35ms.

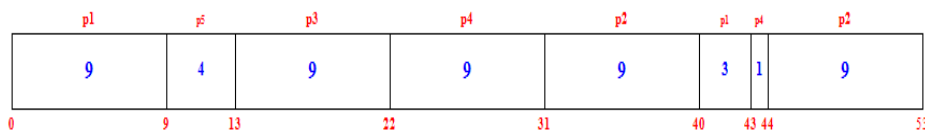


Figure 16. Gantt chart of priority-based round-robin algorithm

To distinguish our work from other, we need comparison study with the recent papers at the same research area. Table 4 shows comparison study between various proposed algorithm on round-robin method. The summary of the comparison between the previous scheduling algorithms with our proposal scheduling algorithm is shown in Figure 17.

Table 4 comparison between various improve on round-robin.

Performance Attribute	AWT	AWT After Applying the proposed algorithm	ATAT	ATAT After Applying the proposed algorithm
Sangwan <i>et al.</i> [12]	44.8ms	42.4ms	64.8ms	62.4ms
Mittal <i>et al.</i> [14]	68.8ms	64.2ms	117ms	112.4ms
Kumar <i>et al.</i> [15]	71.6ms	36ms	119.8ms	84.6ms
Rajput <i>et al.</i> [16]	28ms	24.4ms	40ms	35ms
Nayak <i>et al.</i> [17]	85.6ms	84ms	143.2ms	141.6ms

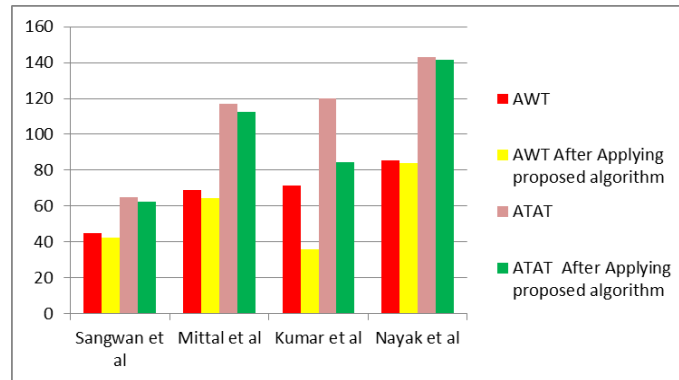


Figure 17. Comparison between previous algorithms with new algorithm

## 6. CONCLUSION

A novel scheduling algorithm for cloud computing is proposed in this paper to overcome common problems of round-robin algorithm. It has the ability to create balance load between job processes in cloud computing at ready queue. The propriety value attached to burst time plays a vital role in enhancing performance of cloud computing scheduling. When compare our proposal with the previous algorithm, we can easily notice that the proposed algorithm has a significant effect on performance of cloud computing. Our experimental result of the proposed jobs scheduling algorithm shows that the proposed schemes possess outstanding enhancing rates with a reduction in waiting time for jobs in queue list. In future work, this idea will apply on another scheduling algorithm for cloud computing that targeted to reduce rate of waiting time and turnaround time at ready queue with.

## REFERENCES

- [1] N. Kulkarni, S. V. N. L. Lalitha, and S. A. Deokar, "Real Time Control and Monitoring of Grid Power Systems using Cloud Computing," *International Journal of Electrical and Computer Engineering*, vol. 9, No. 2, pp. 941-949, 2019.
- [2] K. Bhagchandani and D. P. Augustine, "IoT Based Heart Monitoring and Alerting System with Cloud Computing and Managing The Traffic for an Ambulance in India," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 6, pp. 5068-5074, 2019.
- [3] H. G. Tani, and C. El Amrani, "Cloud Computing CPU Allocation and Scheduling Algorithms using CloudSim Simulator," *International Journal of Electrical and Computer Engineering*, vol. 6, no. 4, pp. 1866-1879, 2016.
- [4] W. Hussain, F. K. Hussain, O. K. Hussain, E. Damiani, and E. Chang, "Formulating and Managing Viable SLAs in Cloud Computing from a Small to Medium Service Provider's Viewpoint: A State-of-the-Art Review," *Information Systems*, vol. 71, pp. 240-259, 2017.
- [5] S. Ghanbari and M. Othman, "A Priority Based Job Scheduling Algorithm in Cloud Computing," *Procedia Engineering*, vol. 50, pp. 778-785, 2012.
- [6] R. Nallakumar, "A Survey on Deadline Constrained Workflow Scheduling Algorithms in Cloud Environment," *International Journal of Computer Science Trends and Technology (IJCTST)*, vol. 2, no. 5, pp. 44-50, 2014. arXiv preprint arXiv:1409.7916.
- [7] T. Dillon, C. Wu, and E. Chang, "Cloud Computing: Issues and Challenges," *2010 24th IEEE International Conference on Advanced Information Networking and Applications*, Australia, pp. 27-33, 2010.
- [8] W. C. Wu and S. C. Yang, "Enhancing Image Security and Privacy in Cloud System using Steganography," *IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, pp. 321-322, June 2017.
- [9] I. C. Lopez, "Scientific Workflow Scheduling for Cloud Computing Environments," *Doctoral Thesis, The University of Sydney*, 2016.
- [10] Y. H. Lee, S. Leu, and R. S. Chang, "Improving Job Scheduling Algorithms in a Grid Environment," *Future generation computer systems*, vol. 27, no. 8, pp. 991-998, 2011.
- [11] S. K. Panda and S. K. Bhoi, "An Effective Round-Robin Algorithm using Min-Max Dispersion Measure," *International Journal on Computer Science and Engineering*, 2014. arXiv preprint arXiv:1404.5869.
- [12] P. Sangwan, M. Sharma, and A. Kumar, "Improved Round-robin Scheduling in Cloud Computing," *Advances in Computational Sciences and Technology*, vol. 10, no. 4, pp. 639-644, 2017.
- [13] B. Maniyar, and B. Kanani, "Review on Round-Robin Algorithm for Task Scheduling in Cloud Computing," *Journal of Emerging Technologies and Innovative Research*, vol. 2, no. 3, 2015.
- [14] S. Mittal, S. Singh, and R. Kaur, "Enhanced Round-robin Technique for Task Scheduling in Cloud Computing Environment," *International Journal of Engineering Research & Technology*, pp. 525-529, 2016.

- [15] D. P. Kumar, T. S. Reddy, and A. Y. Reddy, "Finding Best Time Quantum for Round-robin Scheduling Algorithm to Avoid Frequent Context Switch," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 5, 2014.
- [16] I. S. Rajput and D. Gupta, "A Priority-Based Round-Robin CPU Scheduling Algorithm for real-Time Systems," *International Journal of Innovations in Engineering and Technology*, vol. 1, no. 3, pp. 1-11, 2012.
- [17] D. Nayak, S. K. Malla, and D. Debadarshini, "Improved Round-Robin Scheduling using Dynamic Time Quantum," *International Journal of Computer Applications*, vol. 38, no. 5, pp. 34-38, 2012.
- [18] E. S. Ahmad, E. I. Ahmad, and E. S. Mirdha, "A Novel Dynamic Priority Based Job Scheduling Approach for Cloud Environment," *International Research Journal of Engineering and Technology (IRJET)*, vol. 04, pp. 518-522, 2017.
- [19] D. Agarwal and S. Jain, "Efficient Optimal Algorithm of Task Scheduling in Cloud Computing Environment," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 9, no. 7, 2014. *arXiv preprint arXiv:1404.2076*.
- [20] D. D. H. Miriam, B. Prabha, and J. F. Lilian, "An Efficient Job Scheduling in Isometric HPCLOUD using ZBLA Optimization," *Procedia Computer Science*, vol. 50, pp. 307-315, 2015.
- [21] P. Thakur and M. Mahajan, "Different Scheduling Algorithm in Cloud Computing: A Survey," *International Journal of modern computer science*, vol. 5(1), 2017.
- [22] R. S. Chang, C. Y. Lin, and C. F. Lin, "An Adaptive Scoring Job Scheduling Algorithm for Grid Computing," *Information Sciences*, vol. 207, pp. 79-89, 2012.
- [23] R. Kamal, M. A. Shah, N. Hafeez, and A. Hanif, "Enhanced User Preference Based Intelligent Scheduling Algorithm (E-UPISA)," *23rd International Conference on Automation and Computing (ICAC)*, pp. 1-6, September 2017.
- [24] Monika and A. Jindal, "Optimized Task Scheduling Algorithm for Cloud Computing," *In Information and Communication Technology for Sustainable Development, Springer Nature Singapore*, vol. 9, pp. 431-439, 2018.
- [25] K. Kaur, N. Kaur, and K. Kaur, "A Novel Context and Load-Aware Family Genetic Algorithm Based Task Scheduling in Cloud Computing," *Data Engineering and Intelligent Computing. Advanced in Intelligent Systems and Computing, Springer Nature Singapore*, vol. 542, pp. 521-531, 2018.
- [26] H. G. Tani, and C. El Amrani, "Smarter Round-Robin Scheduling Algorithm for Cloud Computing and Big Data," *Journal of Data Mining and Digital Humanities*, 2018.

## BIOGRAPHIES OF AUTHORS



**Ahmed Subhi Abdalkafor** was born in Anbar-Iraq in 1988. He received his MSc. degree from Computer Science Department in Middle East University, Jordan in 2016. The B.Sc. from Faculty of Computer Science at Anbar University, Iraq. His research interests are, Image Processing, Pattern Classification, Neural Network, Cloud Computing,



**Khattab M. Ali Alheeti** (S'14) received the M.Sc. (Hons.) Computer Networking and Information in 2009, from the University of Al al-Bayt. Currently, Jordan. He is a Ph.D. researcher in Computer Science and Electronic Engineering School, University of Essex, Colchester, UK. His research interests are; security, self-driving vehicles, computer networking, telecommunication.