Indonesian Journal of Electrical Engineering and Computer Science
Vol. 29, No. 3, March 2023, pp. 1542~1550
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v29.i3.pp1542-1550

VM queuing optimal scheduling in cloud using heuristic ant colony optimal based multi-objective genetic approach

Madhina Banu Dawood Ali, Enayathullah Syed Mohamed
Department of Computer Science and Engineering, B. S. A. Crescent Institute of Science and Technology, Chennai, India

Article Info

Article history:

Received Jun 14, 2022
Revised Nov 7, 2022
Accepted Nov 19, 2022

Keywords:

Cloud computing
Datacenter
Execution time
HACOMOG approach
Optimal scheduling
Virtual machine

ABSTRACT

The usages of cloud based applications are increasing tremendously. The cloud computing task distribution is an unknown polynomial time issue that is challengeable to find the optimal solution. In solve above mentioned issue with large amount user's job requests, heuristic ant colony optimal based multi-objective genetic (HACOMOG) approach based job allocation and resource optimization is proposed. Utilization basis scheduler recognizes the task order and optimal resources to be scheduled. The primary contribution of the proposed technique is to develop several online techniques to find solution for the virtual machines (VM) Packing problem sharing-aware and for performing a comprehensive number of studies in order to assess their efficiency with online sharing algorithms. The proposed algorithm considers the utilization basis scheduler output and identified the optimzed task allocation technique based on job execution time, MakeSpan and throughput. The experimental outcomes show that the proposed HACOMOG Algorithm reduces 0.70 seconds job execution time (JET), 0.13 MakeSpan and improve 1.98 throughput on given parameters for 100, 200, and 500 tasks with conventional methodologies.

Corresponding Author:

Madhina Banu Dawood Ali
Department of Computer Science and Engineering, B. S. A Crescent Institute of Science and Technology
Chennai, Tamilnadu, India
Email: madhinabanu.phd@gmail.com

1. INTRODUCTION

Cloud computing is a significant element of cutting edge computing systems. Computing thoughts, innovation and models have created and united inside the most recent decades. Cloud computing might be a processing innovation that is rapidly consolidating itself on the grounds that the following stage inside the sending and advancement of expanding the amount of distributed application. The job of virtual machines (VMs) has risen as a significant issue in light of the fact that, by virtualization innovation, it makes the cloud computing foundations to be versatile. The job scheduling regulates a variety of jobs in the assigned systems to reduce the response time and to increase the effectiveness of calculating energy resources. Scheduling is the distribution of resources and their effectiveness. In the context of scheduling issues that have been suggested in recent years to improve the following thing namely prices, time, security, reliability, inadequacy and insufficiency of control. It also examines the issues which discussed in the scheduling and the work performed. Figure 1 shows the task scheduling in cloud environments. Figure 1 expressesdata transmission/task allocation from end user to cloud server after specifying the taskscheduling in cloud environments. It will also assign the group of task to virtual machines. Because of dynamic nature of cloud frameworks, the applications in a cloud frameworkneed to meet the various requirement of resources and scheduling is altogether different. Irregular job scheduling methodologies may build the job's execution time and minimize the throughput of the cloud framework. To take up this challenge the system tend to review the quantity of expeditiously job scheduling algorithms.

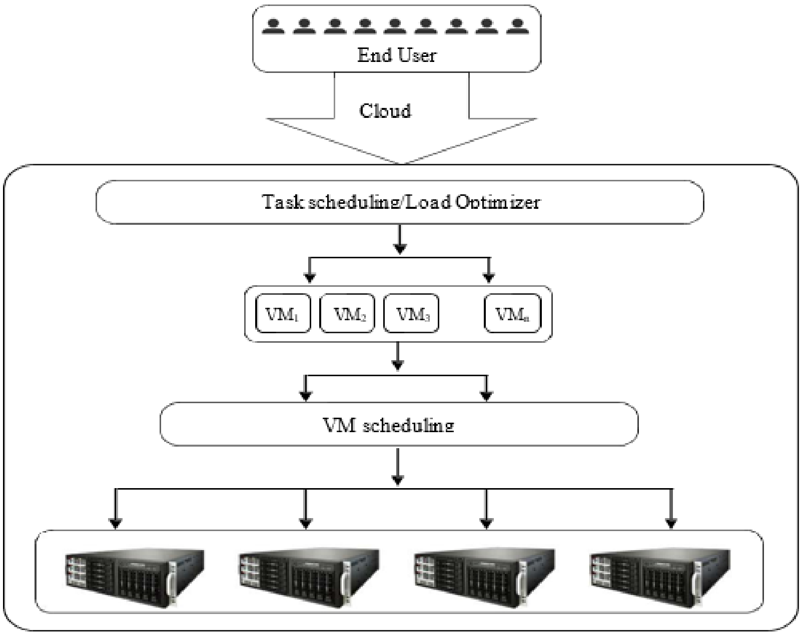

Figure 1. Task scheduling cloud environments

Various elementary algorithms are utilized for task scheduling for a cloud domain like ant colony optimization (ACO) calculation, genetic calculation (GA), simulated annealing (SA) and practical swarm optimization (PSO) reource models is performed with the help of different conventional optimization methods. Because of poor search capability, low convergence rate and failure to work in a unique environment. The current procedures can not achieve ideal optimum task allocation in terms of scheduled quality of service (QoS) parameters. These developments on optimal scheduling of VM's are a significant issue. The method was characterised by another chromosomal structure, pre-arranged beginning population dependant on the size of the task and the resource's processing speed, and adjustment of all objective estimations in resultant ages. When allocating resources, the resource costs' ambiguity and their vitality's use, however, are not taken into account. With the help of population evolutionary algorithms and natural evolution, the optimal set for multi-target optimization problems advanced in the direction of a non-dominated arrangement solution. However, using the resource with large-scale applications is more computationally expensive. A difficult challenge in cloud computing is the scheduling of jobs. Because of dynamic nature of cloud systems, the applications in a cloud framework need to meet the various reuirements of resources and scheduling is altogether different. Inappropriate assignment scheduling techniques can build the task's execution time and minimal throughput in cloud environment. To respond to this issues, the proposed system reuired to review the number of efficient job scheduling algorithms.

The current VM resources scheduling in cloud computing for most parts considers the present state of system but seldom considers historical data and system variation, whichmost of the time leads to load imbalance of system. According to the current situations of system and historical data through genetic algorithm, the technique figures ahead impact. It will have on the framework after communication of the required VM resources and after that selectss the least-effective solution. It perceives the best load balancing and reduces or avoids dynamic migration cost and load balancing by traditional algorithm after scheduling [1]. In cloud computing, an allocation system means to improve the usage of assets for fulfilling the requests of cloud users. Physical computing assets are preoccupied through virtualization as VMs, each involving processor, memory and backing for communication. The resource determination and optimization apportion the applicant assets to clients [2]. The physical assets are shared among numerous clients through virtualization and provisioning. The relationship between asset allocation, cloud suppliers and cloud clients is evaluated. The cloud is equal and distributed system containing virtual machine and inter connected machine are progressively apportioned. The cloud market is utilizing cloud services such as infrastructure as a service (IaaS) and the market request the QoS. The various consumer and supplier considered for the promoting to diminish the quality of services. The issue illuminated by utilizing combinational auction, value forecast and winner assurance. Use of information technology standards to build up the design and model of the market-oriented cloud [3].

The work considers asset allocation in a cloud computing market through the auctioning of VM occasions. It generalizes the current study by presenting combinational auctions of heterogeneous VMs, and models dynamic VM provisioning. Social government welfare expansion under powerful asset provisioning is demonstrated NP-hard, and displayed with a straight number program [4]. It handled the asset question: what is the job of uncertainty in cloud computing services and asset provisioning? Those procedures are for programming in cloud computing endeavours underneath uncertainty, and address ways for moderating occupation execution time uncertainty inside the asset provisioning [5], [6]. Load balanced min-min algorithm using genetic algorithm so as to reduce the make range and increment the use of asset [7]. The technique competes to current technique on same target [8]. The technical obstacles to these changes and put forward architecture for succeeding them in the design; a client uses VM technology to rapidly embody modified service software on a nearby cloudlet and afterward utilize assistance for a wireless area network [9]. Thedevices normally work as machine or group of machines that is a lot of associated with the web and accessible for use by close by devices [10]. The framework accepted that the VM packaging issue is increasingly summed up, called sharing-aware VM packing, which focuses on the equivalent VM packing issue, yet empowers examples of the VM on the equivalent physical server to share memory pages, lessening the measure of cloud-based resources expected to satisfy user need [11]. An improved load balanced min-min calculation utilizing hereditary calculation (GA) so as to limit the make length and increment the usage of asset [12].

The method explained a stochastic (arbitrary) model of a distributed computing cluster, where jobs show up as per a stochastic procedure and solicitation VMs, which are determined as far as assets like memory, CPU and storage space [13]. While, there depends a few plan issues identified with such structures. Where, its solitary focus on the rssource designation issues, for example, the development of algorithms for the load balancing within servers, and algorithms for planning or programming VM arrangements [14]. A PSO-COGENT method was implemented that optimizes the assignment of the allocated resources [15]. It optimizes parameters using autonomous discrete assignments, depending on the specified fitness function while taking into account the limiting time frame. A disaggregated datacenter network framework expressed [16]. The first scheduling algorithm specially implemented for disaggregated computing. The system explains the advantages that disaggregation will get to operators [17]. The current existing transport protocols (with TCP included), given about their Internet based beginnings, are skeptic for such stream flow deadlines [18]. The framework depicted the arranging and execution of D3, a deadline aware management protocol that is modified for the data centre environment [19]. D3 utilizes explicit rate the management to apportion data transmission reliable with flow deadlines [20], [21].

Load balancing assumes fundamental job in cloud execution and its strength, also it discussed about many loads balancing method which helped with distributing the load among the nodes and obtained which suited the most which fit the most [22]. A fuzzy logic approachtodeal with limit the energy consumption by deciding the fitting flexibly speed of the processor accommodated timing requirements that are guaranteed [23]. Enhanced minimal resource optimization-based scheduling algorithm to reduce the assets and keep up the QOS? The method improves service brokering policy to minimize delay response in cloud [24]-[26]. Multipath delay commutator fast fourier transform has been proposed for enhancing the throughput and speed [27]. Cooperative routing using the fresher encounter algorithm to improve energy-efficiency and solves the node dead issues [28].

Objectives: heuristic ant colony optimal based multi-objective genetic (HACOMOG) approach is designed to offer effective task allocation and concentrate on request time, response time and space availability. The method upgrades HACOMOG Approach that acknowledges the utilization basis scheduler output as its info. In GA the best solution is assessed by applying the fitness function over the chromosomes and the n best elites are found. The nth fitted arrangement of GA is converted into the initial pheromone tehnique. The crossover operation is applied to consolidate GA's best arrangement and ACO's final solution. A heterogeneous asset allocation approach, known as skewness-avoidance multi-asset assignment, to apportion asset steady with shifted needs on contrasting sorts of assets is proposed. Proposed solution incorporates a VM allocation algorithmic to ensure heterogeneous remaining tasks at hand are distributed fitly to avoid skew resource utilization in physical machines (PMs), and a model-based appropriate to deal with active the proper number of dynamic PMs to work skewness-avoidance multi-asset allotment. The main contributions of article are given bellows:

- To propose HACOMOG Approach based task allocation technique to handle the large volume of cloud user’s request-response.
- To provide an efficient utilization basis scheduler that prioritize the job based on the execution time and memory details.
- To allocate resources based on datacenter load predictions in order to efficiently optimise resource usage.
- To improve load optimization and keep data communication between the user base and the reliability and stability.
- To reduce job execution time (JET), MakeSpan and improve throughput of proposed HACOMOG Approach rather than conventional methodologies.

2. HEURISTIC ANT COLONY OPTIMAL BASED MULTI-OBJECTIVE GENETIC APPROACH

In this section, The module describes HACOMOG Approach based job allocation method for productive utility-based scheduler that positions the client assignments dependent on the execution time and memory details. It additionally distinguishes the suitable VMs based on the capable models to execute the jobs in the queue. Figure 2 expresses the system architecture, executions steps with diagrammatic representations. Process carried in implementing task scheduling with ACO. Proposed HACOMOG Approach considers the utilization basis scheduler yield output as its input. In GA the best solution is assessed by applying the fitness function over the chromosomes and the n best elites are found. The nth fitted arrangement of GA is changed into the underlying pheromone update techniques. The hybrid activity is applied to consolidate GA's best arrangement and ACO's final solution.

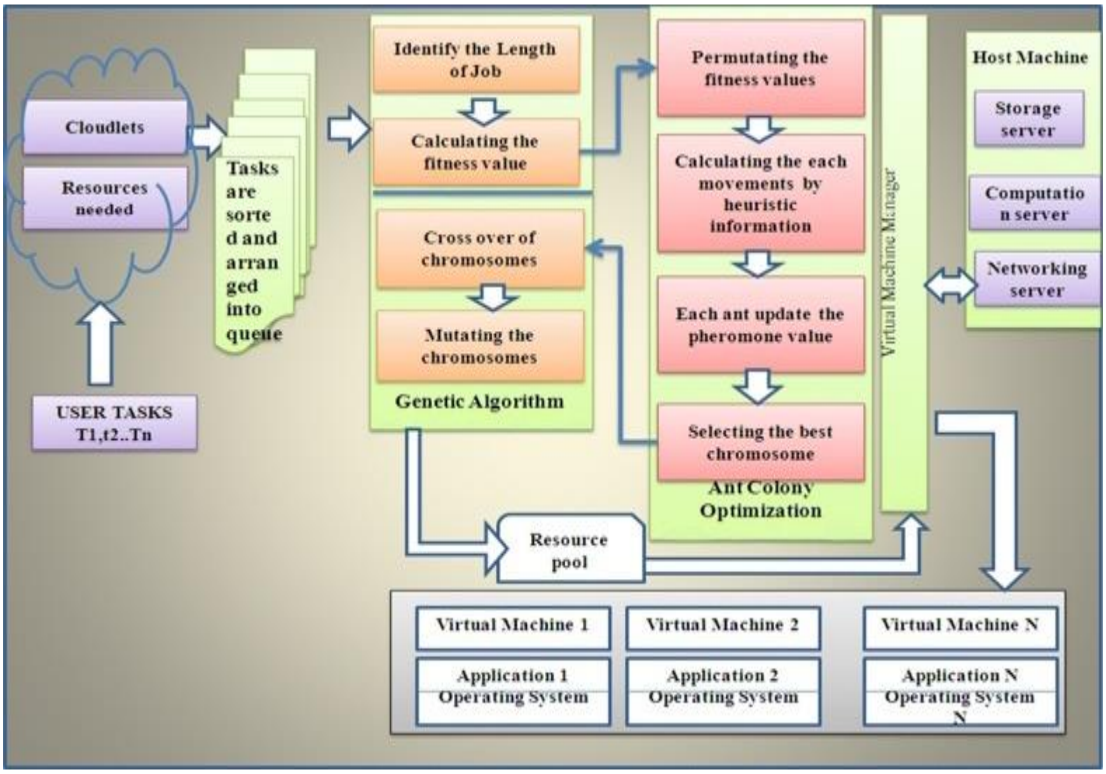

Figure 2. System architecture

Proposed priority technique is developed for proficient execution of job and examination with genetic and ACO method. A heterogeneous asset allocation approach, known as skewness-avoidance multi- rssource allocation, to distribute rssource reliable with varied needs on varying sorts of rssource is proposed. Proposed technique arrangement incorporates a VM assignment algorithmic standard to ensure heterogeneous workloads are designated fitly to maintain a strategic distance from active rssource usage in PMs, and a model-based method to deal with evaluation of the suitable number of dynamic PMs to work skewness-evasion multi-asset allocation. Proposed technique utilize the algorithmic program as a structure block for developing of a randomized combinational auction approach that is computationally skilful, honest in desire, and expectation a similar social welfare approximation factor. It offers on-request; web pay-based pay-as-you-go facilities. The real time conduct of immense scope uses of web in accessible datacentres, with combination of their distributed workflow, prompts deadline with associated datacenter application traffic. A network stream is helpul, and gives to application throughput and service provider revenue if, and just in the event that, it finishes within its deadline. The proposed algorithm execution steps are shown below in details:

- Construct ant solutions: The technique for building choices might be considered as a stage on the structure chart G=(V, E).
- Apply local search: The objective of the pheromone test is to raise and minimize the pheromone scores associated with excellent alternatives or effective individuals.
- Update pheromones: The objective of the pheromone update is to expand the pheromone esteems related with great or promising arrangements, and to minimize those that are related with terrible ones.

- Ant colony system: Ant System is the first ACO algorithm proposed in the survey work. The principle characteristic is that, at each iteration procedure, pheromone esteems are denied by all the m ants that have developed an answer inside the cycle itself. In ACO System a neighbourhood pheromone update is acquainted furthermore with the pheromone update performed toward the completion of the improvement technique. Local pheromone update is performed by all the ants when each development step. Every insect applies it just to the last edge navigated.
- Initialization: During the initialization steps, a solution memory of the ideal population size is designed by a many people individually randomly producing from a uniform distribution.
- Fitness function: The issue is characterized as a multi-objective issue to increase less makespan that is, the time when last task is practiced and stream time that includes minimizing the sum of conclusion period times all tasks.
- Selection: Selection operators are utilized to choose great people and design a mating pool to which the crossover operators will be applied. The Selection is finished by making two subjective assigned people groups to participate inside the competition and decide on most effective among them.
- Crossover: The objective of any transformative algorithm is to acquire relatives of better quality that will take care of next generation and empower the search to investigate new areas of solutions space not investigated at this point. Crossover accomplishes this by choosing the people from the parental generation and interchanging their genes, to get new individuals.
- Mutation: Mutation transforms at least one more gene characteristics in a people from its initial state, which prompts new people looking to locate a stronger alternative than previously

The primary contribution of the proposed technique is to develop several online methods to solve the VM Packing issue sharing-aware and to perform an comprehensive number of studies in order to assess their efficiency with several current online sharing algorithms. Here, proposed algorithms take more jobs and attempt to diminish the execution time. It brought technique deploy in cloud environment. Where, execution time difference will be observed in cloud and grid. The proposed algorithm pseudo code is shown below.

```
Input : Cloudlet task, resource (VM, RAM, Memory, Bandwidth, DC OS, architectures)
Output:     Optimized task with Job Execution Time, MakeSpan, Throughput
Procedure:
Steps 1:    Initiate Genemin-impro-ratio, T1, T2, T3, T4;
Steps 2:    Describe objective function and fitness function;
Steps 3:    While Genedie< Genemin-impro-ratio do
Steps 4:    Select randomly generated group of real numbers;
Steps 5:    Apply Selection, Crossover and Mutation process;
Steps 6:    End while
Steps 7:    Create various optimal solution sets;
Steps 8:    Initiate n, m, α, β, NC-max;
Steps 9:    Consign m ants on n nodes;
Steps 10:   While NC<NC-max do
Steps 11:   Initiate TabU;
Steps 12:   For all ants in ANT, antk € ANT
Steps 13:   For all tasks in meta-task T, tj €T
Steps 14:   Choose next task;
Steps 15:   Update TabU, add the task tj into Tabu;
Steps 16:   Update local pheromone tj;
Steps 17:   End for
Steps 18:   End for
Steps 19:   Estimate globally optimal scheduling method through iteration;
Steps 20:   Revised global pheromone with best one only;
Steps 21:   End while
Steps 22:   Display optimized solution with Job Execution Time, MakeSpan, Throughput
```

3. RESULTS AND DISUSSIONS

The experiment is excuted on Intel core i^{7th} processor, 08 GB RAM and 500 GB Memory. The execution is done in Java programming environments utilizing JDK 1.8, NetBeans 8.0.2. The usage is utilized Cloudsim library to assess execution of proposed technique. The CloudSim toolkit gives the model and features of Cloud framework components, for example, datacenter, VMs and asset provisioning arrangements.

Proposed HACOMOG Approach has limit measure of VM and variable measure of cloudlets. The experiment setup configurations are apportioned to deploy the examination for assessing the productivity of proposed technique whose input parameters are described in Table 1.

Table 1. Cloud experimental details

Parameters	Value
Number of Jobs	50-1000
Virtual Machine	15
MIPS	280-1000
RAM	512-2048
Bandwidth	1000 MBPS
DC VM	Xen
No. of Process Machine	4
No. of Running Deployment	50
DC VM Policy	Time Shared
DC OS	Linux
VM Memory	1-10 GB
DC Architecture	X86

3.1. Simulation results

This module presents the exploratory information, the outcomes and results analysis. The presentation of the proposed HACOMOG Approach is assessed utilizing the parameters like response time, completion time and throughput. CloudSim is an extendible simulation toolkit that engages design and reproduction of the Cloud processing frameworks and application provisioning environments. Proposed HACOMOG Approach describes the mathematical expression of JET, MakeSpan and Throughput.

3.1.1. Job execution time

Job Execution Time are included in the job execution time. It is an average value for each user, but it is not standardised based on the amount of work. The JET runs the whole programme that user from multiple datacenter requests in datacenter processing. It reduces VM load and speeds up task execution time. It allows the recovery of the request from the cloud datacenter. It demonstrates how quickly consumer applications are responded. The cross product of work time and price on resource j is known as JETi, j. The task selects the minimal JET asset during the auction, which is stated as (1).

$$JET_{tp} = \left(TS_{ip} + \frac{taskload_t}{Ability_p}\right) \times \left(price_p \times \frac{taskload_t}{Ability_p}\right) \quad (1)$$

Only after its parent tasks are completed by the partial ordering relationship of task begins for each assignment. ti, j relates to the moment the resource starts and the last time I run on resource j. When his parent's task has been completed and the resource j is free. Workload I refer to job capacity load j, cost j is computing capacity and cost.

3.1.2. MakeSpan

A virtual machine determines the makespan as the amount of time needed to do the full task at hand by the set deadline. After all jobs have been executed, Makespan chooses the resources for the entire system (VM, RAM, bandwidth, and memory). In (2) and represents MakeSpan (3).

$$MakeSpan = max\{MS_{rj}\} \quad (2)$$

$$MS_{rj} = \sum_{T_i \in \theta_{rj}} EET_{T_i r_j} \quad (3)$$

3.1.3. Throughput (TRP)

The throughput is depicted as each task is effectively responded by the complete amount of work requests in the cloud. The throughput formula is calculated in (4).

$$TRP = \frac{1}{N_{Task}} \sum_{i=1}^{N_{Task}} \left[\frac{Size_{VM} - Size_{Task}}{Size_{VM}}\right] \times 100 \quad (4)$$

Where, N_{Task} is the complete amount of tasks that customers ask by users, $Size_{VM}$stores is the VM storage, and $Size_{Task}$is the overall task size.

Table 2 shows JET, Makespanand TRP, 100, 200, and 500 tasks with conventional methods. Proposed HACOMOG Approach is evaluated Min-Min, HoneyBee, PSO, and PSO-COGENT conventional methods. Based on the performance of Proposed HACOMOG Approach, it's observed thatPSO-COGENT scheduling algorithm is closest competitors. Where, PSO-COGENT assigns a virtual machine to perform its tasks and optimizes the parameters necessary for understanding the QoS parameter within deadline. However, during simultaneous cloud assignment handling, the technique fails to decrease the execution time and optimize the resources. PSO is an optimizing algorithm centered on the population that imitates the behavior of animal swarm that is called a particle each participant in a swarm. PSO is a meta-heuristic algorithm, which offers the almost ideal alternative in brief cycles. PSO provides the best performance and execution time. PSO algorithm is used for both continuous and separate problem solving. To fix discrete optimization problems, binary PSO is used. However, PSO method cannot optimize the asset and decrease costs during multiple tasks. HoneyBee method expressed to balance load across VMs and minimize the makespan in cloud environment. Here, the VMs are grouped into three groups based on their loads; over-loaded VMs, under-loaded VMs, and balanced VMs. The technique switches jobs from over-loaded VMs, and takes the decision of submitting them to one of the under-loaded VMs. A job is considered as a honeyBee and the under-loaded VMs are considered as the destination of the honeybees. But, HoneyBee is failed to optimize the VM queue task scheduling. Min-Min scheduling algorithms are used to minimize the MakeSpan, cost, and optimize resource in cloud and grid computing environments. In the cloudlist, it selects a cloud with the lowest running time and assigns to the virtual machine that generates the minimum runtime. The algorithm prioritized assignments with minimal finishing time. However, this means that clouds boost the system's total response time when clouds are much more numerous with minimum finishing period.

Table 2. JET, MakeSpan, and TRP for 100, 200, and 500 tasks

Existing algorithms	100			200			500		
	ET	MakeSpan	RP	ET	MakeSpan	RP	ET	Makespan	RP
Min-Min	0.10	0.863	0.10	1.33	1.45	0.84	07.13	3.47	0.66
HoneyBee	9.93	0.898	0.20	1.07	1.48	0.87	07.31	3.42	0.66
PSO	9.42	0.848	0.15	0.70	1.41	0.91	05.36	3.20	0.74
PSO-COGENT	9.15	0.810	0.22	0.30	1.39	0.96	04.96	3.10	0.76
HACOMOG	8.90	0.795	0.29	9.90	1.31	0.98	03.50	2.97	0.80

Figure 3 shows comparison between compositions and generations, which gives a result for max/average fitness value per generation. Figure 4 shows comparison between fitness value and generations, which gives a result for max/average fitness value per generation. Figure 5 represents the Utility value Vs iterations and it shows evaluation between utility worth and iterations.

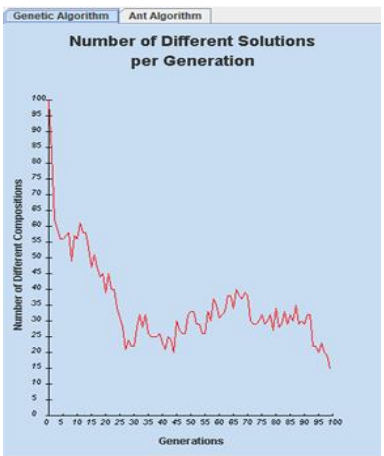

Figure 3. Compositions Vs generations

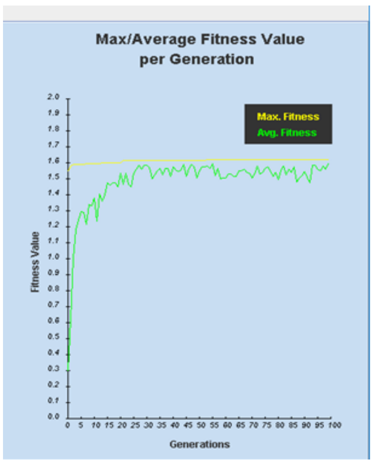

Figure 4. Fitness Vs generations

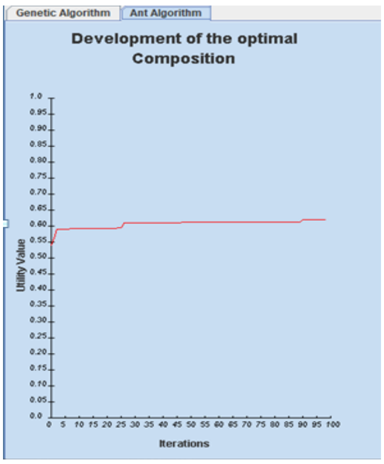

Figure 5. Utility value Vs iterations

Iterations value ranges from 0 to 100 and utility value reaches from 0.0 to 1.0. Proposed HACOMOG Approach expressed for efficient job executions. The quality of services parameters determined by the cloud user, which contain work execution time, MakeSpan and throughput. It also offers fairness resource allocation, which play vital role in task scheduling. Here, cloud users are examined and requested in the queue dependent on utility threshold esteems like memory and execution time. The utility function is utilized to evaluate the real memory and execution time necessities of the jobs. In view of the utility values of the jobs, the assignments are requested in a queue. VMs are selected dependent on the capacity criteria in which they can produce the user-defined task in the queue. Hence, proposed method performs best on overall parameters in respective domains. Finally, it can be said that proposed HACOMOG Approach reduces 0.70 seconds JET, 0.13 MakeSpan and improve 1.98 Throughput on given parameters for 100,200 and 500 tasks with conventional methodologies.

4. CONCLUSION

The article presents HACOMOG Approach for effective utilization basis scheduler that prioritizes the user task dependent on the execution time and memory details. It additionally recognizes the fitting VMs based on the capacity criteria to execute the tasks in the queue. In GA-ACO algorithm, ACO helps GA from falling into neighbourhood optimal solution and ACO is utilized to upgrade the GA solutions. By combining ACO algorithm with GA, the performance of the job allocation is improved. In GA the best solution is assessed by applying the fitness function over the chromosomes and the n best elites are found. The nth fitted arrangement of GA is changed into the underlying pheromone update techniques. In this work, it's demonstrated that the HACOMOG Approach minimizes the response time and completion time while expands the throughput of the system. Scheduling refers to an assortment of policies to deal with the request for work that will be performed by a system. VMs are chosen dependent on capacity criteria in which they can perform the user-defined task in the queue. HACOMOG Approach considers the utilization basis scheduler yield output as its input. Hence, it can be said that proposed method performs best on overall parameters in respective domains. Finally, it can be said that HACOMOG Approach reduces 0.70 seconds JET, 0.13 MakeSpan and improve 1.98 Throughput on given parameters for 100, 200, and 500 tasks with conventional methodologies. In future, paper can be extended with security related issues in load balancing which will be implemented in the real-world application in cloud environment. Cloud demand is increasing day-by-day, so it is required to adopt the privacy with load balancing and resource optimization in cloud.

REFERENCES

[1] H. Dinh. C. Lee, P. Wang, and D. Niyato, "Survey based on the cloud computing style, approaches, and application," *Wireless Communications and Mobile Computing*, vol. 13, no. 18, pp. 1-6, 2013, doi: 10.1002/wcm.1203.

[2] M. Eisa, E. I. Esedimy, and A. Halawani, "Improving scalability of cloud system based on routing algorithms," *International Journal of Computer Applications*, vol. 101, no. 11, pp. 40-46, 2014.

[3] S. S. Rajput, and V. S. Kushwah, "A genetic based mostly impermanent load-balanced min-min task scheduling programme rule for the load reconciliation of cloud computing," *Eighth International Conference on the machine Intelligence and therefore the Communication Networks*, 2016, pp. 677–681, doi: 10.1109/CICN.2016.139.

[4] K. M. Cho, P. W. Tsai, C. W. Tsai, and C. S. Yang, "A hybrid meta-heuristic algorithm for VM scheduling with load balancing in cloud computing," *Neural Computing Applications*, vol. 26, no. 6, pp. 1297–1309, 2015, doi: 10.1007/s00521-014-1804-9.

[5] S. Abolfazli, Z. Sanaei, E. Ahmed, A. Gani, and R. Buyya, "Cloud-based augmentation for mobile devices: motivation, taxonomies, and open challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 337-368, 2013, doi: 10.1109/SURV.2013.070813.00285.

[6] J. Ma, W. Li, T. FU, L. Yan, and G. Hu, "A novel dynamic task scheduling algorithm based on improved genetic algorithm in cloud computing," *Wireless Communications, Networking and Applications, Lecture Notes in Electrical Engineering*, Springer, vol. 348, pp. 829-835, 2016, doi: 10.1007/978-81-322-2580-5_75.

[7] F. Liu, *et al.*, "Gearing resource-poor mobile devices with powerful clouds: architectures, challenges, and applications," *IEEE Wireless communications*, vol. 20, no. 3, pp. 14-22, 2013, doi: 10.1109/MWC.2013.6549279.

[8] S. H. H. Madni, M. S. A. Latiff, Y. Coulibaly, and S. M. Abdulhamid, "Resource scheduling for infrastructure as a service (IAAS) in cloud computing: Challenges and opportunities," *Journal of Network and Computer Applications*, vol. 68, pp. 173-200, 2016, doi: 10.1016/j.jnca.2016.04.016.

[9] M. Satyanarayanan, V. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," *IEEE pervasive Computing*, vol. 8, pp. 14-23, 2009, doi: 10.1145/2043164.2018443.

[10] C. Wilson, H. Ballani, T. Karagiannis, and A. Rowtron, "Better never than late: Meeting deadlines in datacenter networks," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, pp. 50-61, 2011.

[11] S. Rampersaud and D. Grosu, "Sharing-aware internet virtual machine (VM) packing in the heterogeneous resource clouds," *IEEE Transactions on Distributed and Parallel Systems*, vol. 28, no. 7, pp. 2046–2059, 2017, doi: 10.1109/TPDS.2016.2641937.

[12] L. M. Ausubel, "An efficient auction for heterogeneous commodities," *American Economic Review*, vol. 96, no. 3, pp. 602-629, 2006, doi: 10.1257/aer.96.3.602.

[13] F. Ramezani, J. Lu, and F. Hussain, "Task scheduling optimization in cloud computing applying multi-objective particle swarm optimization," *Service-Oriented Computing. ICSOC 2013*. Lecture Notes in Computer Science, Springer, vol. 8274, pp. 237-251, 2013, doi: 10.1007/978-3-642-45005-1_17.

[14] A. M. S. Kumar and M. Venkatesan, "Multi-objective task scheduling using hybrid genetic-ant colony optimization algorithm in cloud environmen," *Wireless Personal Communications*, vol. 107, pp. 1835-1848, 2019, doi: 10.1007/s11277-019-06360-8.

[15] M. Kumar and S. C. Sharma, "PSO-COGENT: Cost and energy efficient scheduling in cloud environment with deadline constraint," *Sustainable Computing: Informatics and Systems,* vol. 19, September 2018, pp. 147-164, 2018, 10.1016/j.suscom.2018.06.002.

[16] V. R. T. Kanakala and V. K. Reddy, "Performance analysis of load balancing techniques in cloud computing environment," *Indonesian Journal of Electrical Engineering*, vol. 13, no. 3, pp. 568 -573, 2015, doi:10.11591/telkomnika.v13i3.7237.

[17] M. H. A. Awadalla, "Heuristic approach for scheduling dependent real-time tasks," *Bulletin of Electrical Engineering and Informatics*, vol. 4, no. 3, pp. 217-230, 2015, doi: doi.org/10.11591/eei.v4i3.505.

[18] R. Mahadevan and N. Anbazhagan, "An efficient framework to improve QoS of CSP using enhanced minimal resource optimization based scheduling algorithm," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 12, no. 3, pp. 1179-1186, 2018, doi: 10.11591/ijeecs.v12.i3.pp1179-1186.

[19] N. Sabarinathan, "An effective method for qos-constrained work flow scheduling of cloud services," *International Journal of MC Square Scientific Research*, vol. 3, no. 1, pp. 7-17, 2011, doi: 10.4304/jsw.9.4.926-930.

[20] Z. Benlalia, K. Abouelmehdi, A. Beni-hssane, and A. Ezzati, "Comparing load balancing algorithms for web application in cloud environment," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 17, no. 2, pp. 1104-1108, 2020, doi: 10.11591/ijeecs.v17.i2.pp1104-1108.

[21] A. Joshi and S. D. Munisamy, "Enhancement of cloud performance metrics using dynamic degree memory balanced allocation algorithm," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22, no. 3, pp. 1697-1707, 2021, doi: 10.11591/ijeecs.v22.i3.pp1697-1707.

[22] T. Rao, "Optimized scheduling techniques focused on powerful heuristics leveraging cloud services soft computing," *International Journal of Pervasive Computing and Communications.* vol. 18, no. 1, pp. 5-17, 2022, doi: 10.1108/IJPCC-04-2021-0087.

[23] Z. Salmanian, H. Izadkhah, and A. Isazadeh, "Auto-scale resource provisioning in IaaS Clouds," *The Computer Journal*, vol. 65, no. 2, pp. 297-309, 2022, doi: 10.1093/comjnl/bxaa030.

[24] A. Moori, B. Barekatain, and M. Akbari, "LATOC: an enhanced load balancing algorithm based on hybrid AHP-TOPSIS and OPSO algorithms in cloud computing," *The Journal of Supercomputing*, vol. 78, no. 4, pp. 4882-910, 2022, doi: 10.1007/s11227-021-04042-6.

[25] A. Chhabra, K. C. Huang, N. Bacanin, and T. A. Rashid, "Optimizing bag-of-tasks scheduling on cloud data centers using hybrid swarm-intelligence meta-heuristic," *The Journal of Supercomputing*, pp. 1-63, 2022, doi: 10.1007/s11227-021-04199-0.

[26] S. Murugan, T. R. G. Babu, and C. Srinivasan, "Underwater object recognition using KNN classifier," *International Journal of MC Square Scientific Research*, vol. 9, no. 3, pp. 48-52, 2017, doi: 10.20894/IJMSR.117.009.003.007.

[27] M. M. Ismail, M. Subbiah, and S. Chelliah, "Design of Pipelined Radix-2, 4 and 8 based multipath delay commutator (MDC) FFt," *Indian Journal of Public Health Research & Development*, vol. 9, no. 3, p. 765, 2018, doi: 10.5958/0976-5506.2018.00380.7.

[28] A. Unnikrishnan and V. Das, "Cooperative routing for improving the lifetime of wireless Ad-Hoc networks," *International Journal of Advances in Signal and Image Sciences*, vol. 8, no. 1, pp. 17–24, Jan. 2022, doi: 10.29284/ijasis.8.1.2022.17-24.

BIOGRAPHIES OF AUTHORS

Madhina Banu Dawood Ali is assistant Professor (Senior Grade) at Computer Science and Engineering department, B. S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, TamilNadu, India. She is completed her Master degree in computer Science and Engineering fromAnna University, Chennai in 2008 and Master of Computer Application at Barathidasan University, Trichy, TamilNadu, India in 2006. She has 13+Year of teaching experience in various institutions in Tamil Nadu, India. She published 13 (4+ articles in Scopus Indexed International journal, 4+ National Journal and 5+ international Conferences) articles in the domain of cloud computing and Network security. Currently She is pursuing her PhD in Computer Science and Engineering from B. S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, Tamil Nadu, India. She participated many international/national conferences, workshop, Faculty development programme and symposium in various institutions and organizations in India. Her Interested areas are cloud computing, internet of things, mobile and pervasive computing, software testing, artificial intelligence, agile software methodology, and usability engineering. She can be contacted at email: madina@crescent.education.

Enayathullah Syed Mohamed is working as Professor and Head of Department at Department of Computer Science & Engineering, B. S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, India. He obtained his PhD in Computer Science from B. S. Abdur Rahman Crescent University in 2015, Master Degree in Computer Science & Engineering from the University of Madras in 2003 and Batcholer degree in Electrical and Electronics Engineering from the University of Madras in 1997. He has 17+ Years of teaching experience in various institutions in Tamil Nadu, India. He published 41 (20+ articles in Scopus Indexed International journal, 4+ National Conferences and 17+ international Conferences) articles and 3 book chapters in the domain of Soft Computing, Information Security, Web Technologies and Big data and Data Visualization. He received 19 awards, published 5+ patents and completed 4+ funding projects from public and private consultancy. He guided 11 PhD Candidates, 12 Post graduate Students and 36 under graduate students. He participated 50+ international/national conferences, workshops, Faculty development programmes and symposiums in various institutions and organizations in India. He delivered 19+ lectures, faculty development programmes and workshops in various institutions in India. His interested areas are soft computing, information security, web technologies, big data and data visualization, mathematical modeling and graphics and multimedia. He can be contacted at email: syedmohamed@crescent.education.