



Improvement for tasks allocation system in VM for cloud datacenter using modified bat algorithm

Arif Ullah¹ · Aziza Chakir²

Received: 2 August 2021 / Revised: 25 February 2022 / Accepted: 9 March 2022 /

Published online: 4 April 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Since its inception, cloud computing has greatly transformed our lives by connecting the entire world through shared computational resources over the internet. The COVID-19 pandemic has also disrupted the traditional learning and businesses and led us towards an era of cloud-based activities. Virtual machine is one of the main elements of virtualization in cloud computing that represents physical server into the virtual machine. The utilizations of these VM's are important to achieved effective task scheduling mechanism in cloud environment. This paper focuses on improvment of the task distribution system in VM for cloud computing using load balancing technique. For that reason modification took place at Bat algorithm fitness function value this section used in load balancer section. When algorithm iteration are complete then time to distribute the task among different VM therefore in this section of algorithm was modified. The second modification took place at the search process of Bat at dimension section. The proposed algorithm is known as modified Bat algorithm. Four parameter are used to check the performance of the system which are throughput, makespan, degree of imbalance and processing time. The proposed algorithm provides efficient result as compaire to other standard technique. Hence the proposed algorithm improved cloud data center accuracy and efficiency.

Keywords Cloud computing · Virtual machine · Task allocation · Datacenter

✉ Arif Ullah
Arifullahms88@gmail.com; arifullah@riphahfsd.edu.pk

Aziza Chakir
aziza1chakir@gmail.com

¹ Department of Computing, Riphah International University, Faisalabad, Punjab 44000 Faisalabad, Pakistan

² Faculty of Law, Economics and Social Sciences, Hassan II University, Casablanca, Morocco

1 Introduction

Cloud computing is new trend of technology which has changing the world direction over the last decade. The deliveries of virtualized resources are performed with the help of internet in cloud computing. These services are delivered with the rule of pay and gain base on the demand with real time services over the internet rather than having these service of local servers or person computer [26]. Cloud resource present as universal tools that any user can rent and release these services with the help of internet. It becomes twenty-first-century technology due to the mixture of high bandwidth statement and low-cost computing with storage [8, 9]. Normally four well know type of cloud computing exit which are private, hybrid, public and community cloud computing. Some of the main services offered by cloud provider are software as a service (SaaS), application as a service (AaaS), hardware as a service (HaaS), desktop as a service (DaaS), backup as a service (BaaS), and network as service (NaaS) [30]. Cloud computing is growing rapidly and is used in areas like education, geospatial sciences, engineering technology, data-intensive applications, health, life science, application programming services, different scientific and business domains. VM is a software program or operating systems that work as the real behaviour of the server or a separate computer. VM is capable of performing task like real system. Different rule and policy are mention in VM development process. With the help of these rule and policy, VM work more efficiently. Further more load balancing technique is used in improving task allocation system for VM in cloud computing [33, 37]. Figure 1 shows the structure of cloud technology.

Cloud computing facing different kind of issues and one of the big issues is task allocation system in VM. During the allocation of tasks are performing in VM there are two conditions can occur. Both conditions are known as under-loaded and over loaded where both situations can cause the network failure or delay of user request. Moreover, a proper load balancing technique is needed to overcome this situation [15]. Dynamic load balancing technique is one of the main element or heart of data center in cloud environment because it helps VM to carry out the same amount of workload and deliver the service with minimal time without delay to the end user. Therefore, well know nature inspired algorithms were selected in improving in task allocation s ystem in VM for cloud computing. This study is an alternative to improve task allocation system of VM's by proposing an improved load balancing technique for cloud computing. [2, 3]. Four parameter are used to check the performance of the system which are throughput, makespan, degree of imbalance

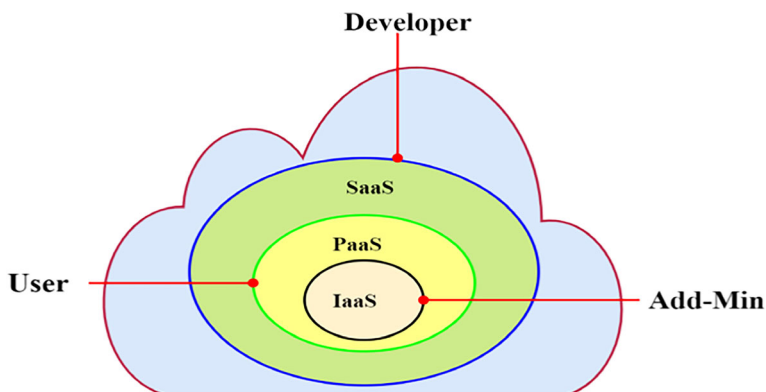


Fig. 1 Cloud computing structure

Table 1 Notation used in this paper

Symbol	Description	Symbol	Description
VM	Virtual machine	iT	Iteration Of Algorithm
LB	Load balancing	Np	Number of requests per period
CC	Cloud computing	MIppt	Million instruction per second of Processing Time
DC	Data center	DC	Data center
PTstvm	Processing time of single tasks in VM	UB	Use base
PTvm	Processing time of all tasks in VM	SCR	Second cycle range
CJmax	Total number of Tasks	SBP	Service broker policy
Ppn	Priority per node	SLA	Service level agreement
M.Bat	Modified Bat Algorithm	RP	Routing Policy
		GVM	Group of virtual machine

and processing time. The result of the paper is organized as. The related work about Bat algorithm and different parameter improvement are discussed briefly in Sect. 2; paper contribution discuss in Sect. 3; simulation environment are briefly described in Sect. 4; performance comparison is presented in Sect. 5 Section 6 discuss about conclusion and future work. Different notations are used in this paper which is mention in Table 1 which helps the reader to understand the paper.

2 Recent work

The basic inspiration behind Bat algorithm is the role of echolocation behaviour of micro-bats [20]. The micro-Bats can search their prey even in the darkness with the help of echolocation behavior these micro-bats have different sound pulses of different properties and used for different hunting strategies for his objects. In Bat algorithm makes a balance between exploration and exploitation with the help of two main parameters which are frequency and emission rate. The wavelength and pulse rate adjusted automatically the distance upon the distance and prey [14]. A lot of work has already has been published about optimization in cloud computing in the section of VM task allocation system some of them are summaries below.

Table 2 Improvement different parameter of Bat algorithm

Ref	VM	Bat modification section	Performance parameters
[23]	VM/ Load balancer	loading detection section	Response time, fault torrent
[7]	VM	SLA violation metrics	Quality of services (QoS) Migration time
[25]	VM	Probrity at initiation section	Makespan and QoS
[21]	VM/ Based learning	Generated section	Response time, Accuracy
[19]	VM	Rules	Reduce Response
[28]	VM	Initiation rule	QoS, Makespan, Task finishing time
[1]	VM/ Load balancer	Search Section	Reduce Response, QoS
[35]	VM	Initial stage	Response Time, QoS
[31]	VM	Depends on policies	Makespan
[16]	VM		Makespan, Throughput
[39]	VM	Global optimized section	QoS, Minimize the decision time
[32]	VM/Game theory	Search Section	Makespan, QoS
[5]	VM	Initiation section	QoS, Accuracy
[18]	VM	Search Rule	QoS, Accuracy
[17]	VM	Fitness function	Makespan
[27]	VM	Fitness function	Makespan

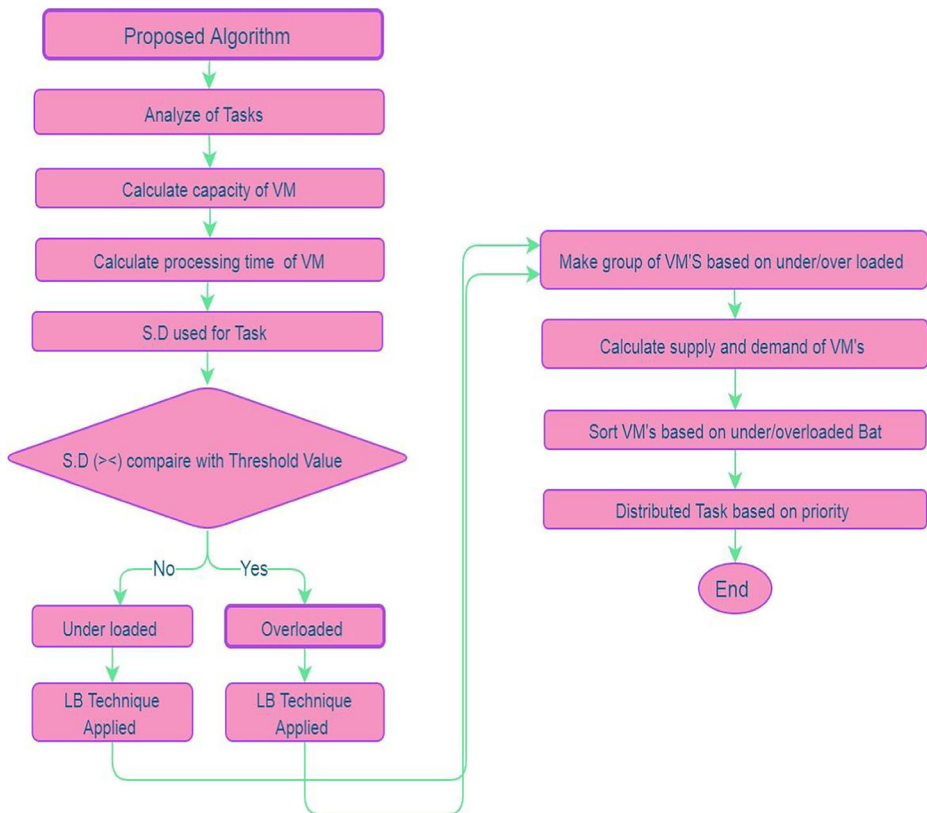


Fig. 2 Proposed algorithm working criteria

VM play important role of resource allocation system therefore the author [29], proposed Bat algorithm using directed acyclic graph with topological ordering in Bat algorithm for improvement in task allocation system of VM for cloud computing. Due to the modification of the above section improve the algorithm waiting time of task and make idle time for VM as it improves the convergence rate high.

Due to the data handing and processing capacity of cloud computing increase therefore it need to proper distributed task among different VM properly. Therefore the author proposed a modified Bat algorithm for resource allocation system optimization. The new approach the workflow system are modified by minimizing the iteration number due to this section it take less time for data execution. The proposed algorithm verified by testing and compare it result with stand algorithm based on the result it improve the makespan of them system [34].

Table 3 Algorithm step 1

Algorithm step 1	
1	$X_{ij} = X_{\min} + (X_{\max} - X_{\min}) \times \rho$ Where $X_{\max} = n$ and $X_{\min} = 0$
2	$\rho \in [0, 1]$ is random number drawn from a uniform distribution.

Table 4 Algorithm step 2

Algorithm step 2

- 1 $f_i = f_{\min} + (f_{\max} - f_{\min}) \times \beta$
 $V_{ij}^{t+1} = V_{ij}^t + \left(X_{ij}^{t-g^{best_i}} \right) \times f_i$
 $X_{ij}^{t+1} = X_{ij}^t + V_{ij}^{t+1}$
- 2 $\beta \in [0, 1]$ Is random number distributed from uniformly gbest is the task number which is placed on VM (j) for the best solution (t) is the number iteration and (fi) is the frequency of Bat. For the local search generate the uniform distribution number if the number bigger the rate of pulse emission (r_i^t) update position.
- 3 $X_{ij}^{t+1} = X_{ij}^t + \text{random}(-1, 1)A$ meant
 Where R and(-1 1) is number of uniform is the average loudness of t^{th} iteration number[6].

Task scheduling is one of the important issues which still exist in cloud computing therefore the author proposed chaos theory and Bat algorithm for task allocation for VM in cloud environment. In the proposed technique the author uses chaos theory for distribution of data to different VMs. Due to the theory it changes the fitness function value and improves the makespan and network stability. It results are compared with well-known algorithms based on the result it improves the task allocation system [24].

Due to the demand of users for better service in cloud computing, the importance of resource allocation systems has increased. Therefore, the author modified the Bat algorithm local search section for improvement in the energy management system. The proposed algorithm is compared to well-known algorithms based on the result it improved the energy utilization system [12].

The author [22] the position of each Bat is mainly updated by the current optimal solution in the early stage of searching and in the late search it also depends on the mean best position which can enhance the convergence speed of the algorithm.

To balance between the local and global search capability in the Bat algorithm, a new search equation is proposed for each Bat based on the history optimal position that it memorized; to avoid falling into a local optimal solution, an abandoned mechanism is adopted for each Bat based on the number of iterations its position has not been improved, to enhance the local search ability, a dynamic adaptive search equation is presented [36]. Table 2 presents the different modified sections of the Bat algorithm.

Table 2 presents the related work about the improvement in different parameters of the Bat algorithm. This parameter modification or improvement took place just for improvement in the resource allocation system of VM in cloud computing. Analysis of the study of these related works motivates this paper in finding some issues which still exist in the Bat algorithm. Due to the modification in the task allocation system, different parameters of time consumptions improved in the cloud datacenter. Our contribution of the paper can be summarized as follows.

Table 5 Algorithm step 3

Algorithm step 3

- 1 $V_{id}^t = v_{id}^{(t-1)} + f_i \left(x_{id}^{(t-1)} - x_d^* \right)$
- 2 In the given equation d presents the dimension of the search space V_{id}^t and $v_{id}^{(t-1)}$ denote the flight speed of the bat i and $t-1$ respectively $x_{id}^{(t-1)}$ denote the position of bat i at time $t-1$ and x_d^* represent the best position in the population up to the t iteration so far.

Table 6 Algorithm step 3

Algorithm step 4

1	$f_i = f_{new}$ $A_i^{t+1} = \alpha A_i^t$ $r_i^{t+1} = r_i^t(1 - e^{-rt})$
2	Where $\alpha = \gamma = 0.7$ Used to update the global best position.

- (b) A survey of existing improvement in different section of Bat algorithm used for load balancing purpose.
- (c) The proposed algorithm modification sections are discussed with explanation.
- (d) The proposed algorithm addresses the VM violation issue like under/overloaded task distribution in cloud datacenter.
- (e) The proposed algorithm improved different time consumptions parameters in cloud data center.
- (f) The proposed algorithm results are comparing with well-known algorithm.
- (g) Discuss the result along with future implementation of the proposed algorithm. Figure 2 presents the frame work of the proposed approach.

3 Proposed modified bat algorithm

The proposed algorithm used inload balancing technique for improvements in task allocation system of VM in cloud computing. Four main parameter throughput, makespan, degree of imbalance and processing time are used to check the performance of proposed technique in cloud computing. Cloud computing consist of a set of data centre and each data centre consist of m set of host and each host consist j of set of VM. The following step of algorithm used for implementation of task distribution. The numbers of population are uniformly distributed randomly to different VM o and n. Bat represents a solution that Bat describes by the position $X_{ij} = [o, n]$ and velocity X_{ij} where I represent the population and j present tasks [38]. The initializing the positions are present step 1 present in Table 3.

In the second step the algorithm generate the new solution which present Table 4.

Table 7 Algorithm step 5

Algorithm step 5

1	Calculate load balancer. After submitting data to the under-loaded VMj the current workload of all available VM's can be calculated by using the information that received from the data center. Thus variance (σ) is called in order to measure the variance of data on VM 's calculate variance of load can be defined as follows [13].
2	$S^2 = \frac{\sum f x_i^2}{n} - X^2$ Where $X = \frac{\sum f_i}{n}$
3	Processing time of VM $X_j = \sum_{i=1}^d \frac{\text{task-length}(i)}{\text{Capacity}}$
4	Means of processing time of all VMs $VM = \sum_{i=1}^d \frac{\text{task-length}(i)}{n-1}$

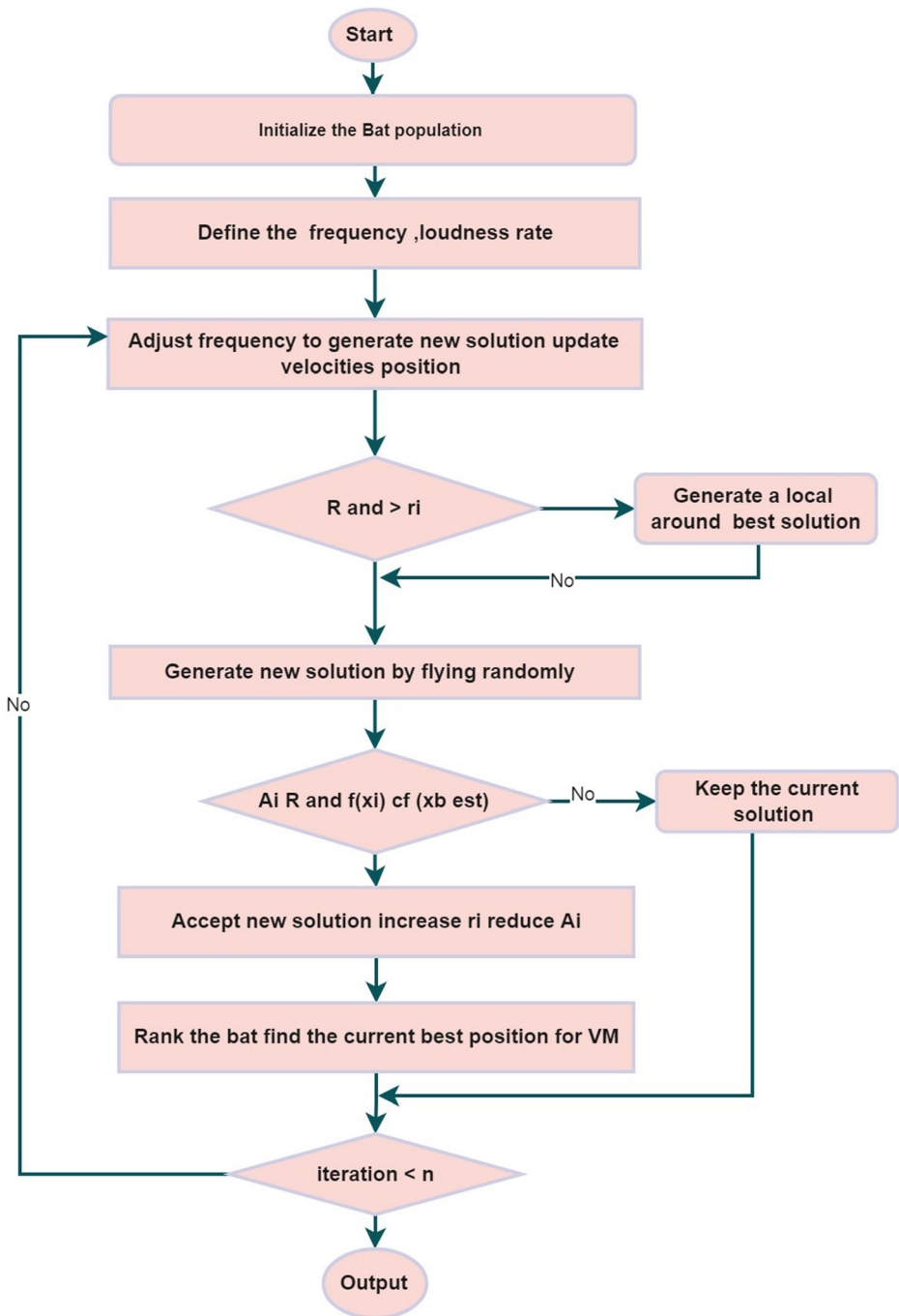


Fig. 3 Flow chat of proposed algorithm

Table 8 Simulation parameters

Type	Parameter	Value	Type	Parameter	Value
Region	From 1 to 4	5	Task/Data	Number of task	100/1000
Data center	Number of data center	5		Length of task	100/200/400 byte
	Number of hosts	100/1000		Number of processor per requirement	250 KB
	Type of Manager	Time and space	Memory	Type of manager	Time and space
Virtual machine	Bandwidth	1000		Total memory	204,800 Mb
	Total number of VM	30/20/20		Number of processors	4 per VM
	Number of processor per virtual machine	4		Total processor	120
	Virtual machine memory	512		Storage Memory	100,000 Mb
	Bandwidth in bit	1000		Viable memory	10,000
Used algorithm	VM image size	1000	Cloudlets CPU	Total number of task	400/600
	Maximum number of Iteration	100		MIPS OF PE/Number of PFs VM	2500/1
			Proposed algo-rithm	Maximum number of Iteration	100
Dataset	Data center	No/VMs	No/VMs D1=5	DC/Policy	12 GBRM
SWF				Random	Windows 2010
Size 400 K/200 k	D1,D2,D3	D1=5	64-bit version	12 GBRM	
PC	D4,D5	D*5=30/50			

After the generate the new solution now it time to speed at which Bat searches for prey in the dimension present as dimension. Table 5 present the equation.

Fourth step form the update loudness and rate of pulse emission are producing a random number form uniform distribution. If the value small smaller then loudness A_i and f new generated by using (Eq 1 in step 3) is smaller then pervious frequenting and update the following equation Tables 6.

The in the algorithm step 5 (eq. 4) are the modified section where they are used for calculation of load on load balancer section (Table 7). When this section calculated the load and get information from data centre where any VM available or not. When some of the VM found overloaded with multiple task then they need to migrate the data to overloaded VM. The data selection method used for Data migration is priority based if any low case priority is selected from migration then other remains as candidate for migration. The above process done on the equation base, line 3,4 of the algorithm step 5 and it compare the which the threshold value. Figure 3 show the flow chat of the proposed technique.

4 Simulation environment

The simulations consist of different experiments which consist of different data centers, different number of VM's running on the physical host and different numbers of tasks are

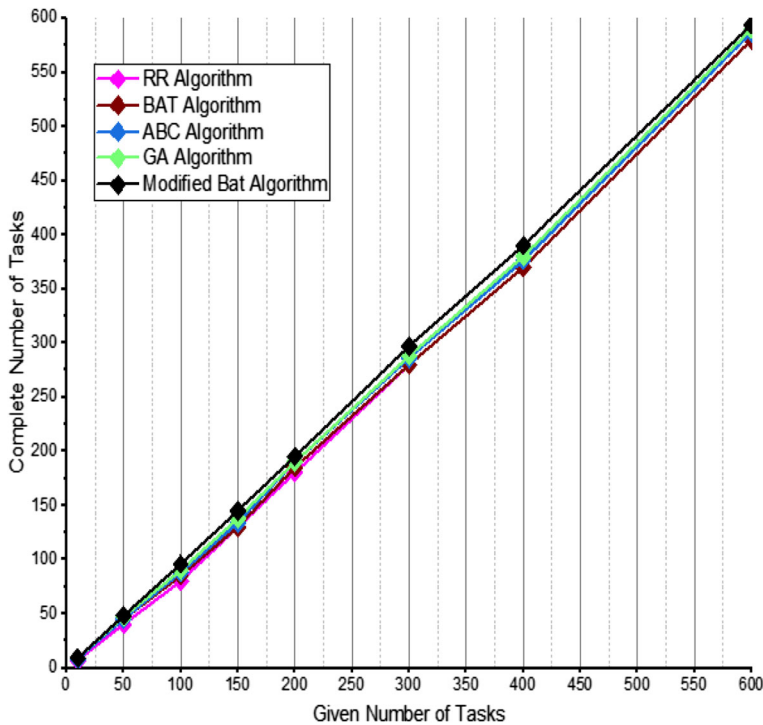


Fig. 4 Makespan result of 600 task

given. Table 8 show different parameters of simulation and these parameters changes at each experiment.

Table 8 present the different parameters some of these parameters are change during each experiments like task number, VM number and data size.

5 Performance comparison

In this section we present all the result of different parameters which are used to measure the performance of the proposed algorithm like throughput, makespan, degree of imbalance and processing time. These parameters are checked with different simulation steps their result are explain below.

5.1 Performance based on makespan

In this sub-section the performance of proposed algorithm measured based on makespan time. For the paper makespan measured as total time taken by resource to complete the executing of all tasks in a cloud datacenter [10]. Here the analysis has been done taking 7 different scenarios taking 200 to 1200 tasks and the makespan time is calculated as.

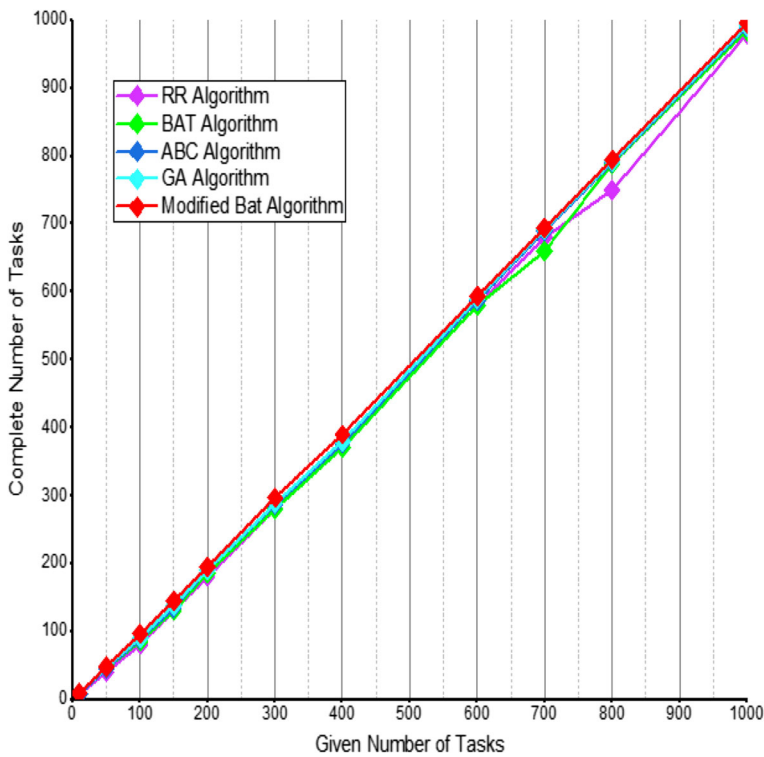


Fig. 5 Makespan result of 1000 task

$$\text{Makespan} = \frac{\text{Given time to the network}}{\text{Complite time of network}} * 100$$

Here five different algorithm like Round-Robin, Bat, artificial bee colony, genetic algorithm, and proposed modified Bat algorithm are evaluated by different number of tasks from 200 to 1200. The result of proposed algorithm makespan are, 187/200, 293/400, 338/600, 396/800, 443/1000 and 478/1200 which are the most less number of time taken by any algorithm for completing number of takes in the experiments. Figures 4, 5 and 6 presents the makespan.

5.2 Performance based on processing time

In this sub section the evaluation performance of the proposed algorithm is calculated based on processing time and the result are measured with other algorithms. The equation of total processing time is given below.

Proces sin $g_{Time} = \text{Cloudlet}_{length} / \text{VM}_{Mips} \times \text{PE}_{Number}$ [11]. Based on Table 9 the result of proposed algorithm are 1089.56, 1123.67, 1234.78, 1234.78, 1876.78 and 2078.89 which are the most lowest total processing time in the simulation Table 9 and Fig. 7 present the overall result.

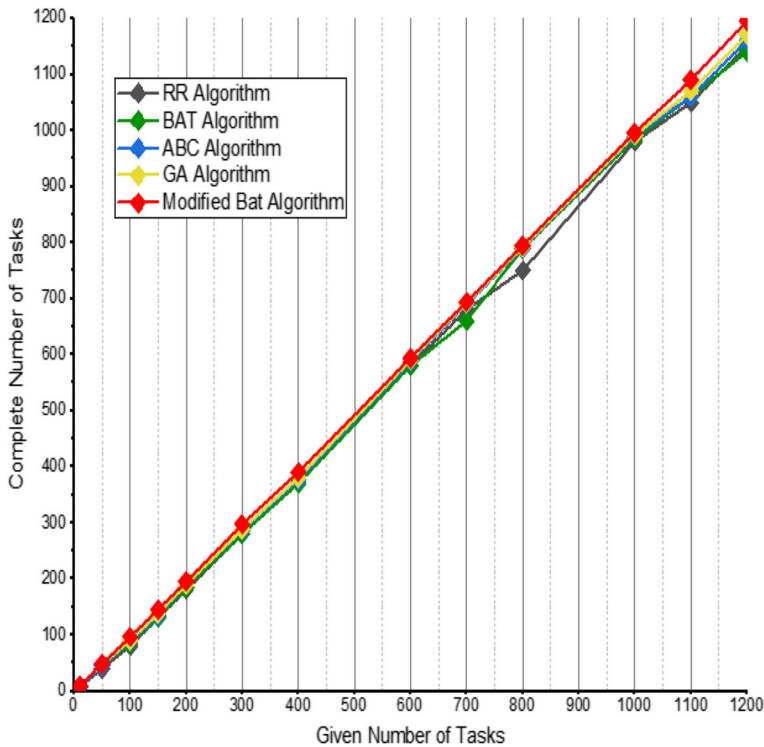


Fig. 6 Makespan result of 1200 task

5.3 Performance based on degree of imbalance

In this sub-section the proposed algorithm is tested based on the degree of imbalance and compared with other present algorithms the given equation presents degree of imbalance.

$$D_i = (T_{\max} - T_{\min}) / T_{\text{avg}}$$

Where $T_{\max}$ and $T_{\min}$ are minimum and maximum T_i present the all VM's T_{avg} is present the average T_i VM [4]. Based on Table 10 the proposed algorithm provide most efficient result as compare to the standard algorithm Table 10 and Fig. 8 present the overall result.

Table 9 Total processing time

Total processing time					
Task Number	RR	Bat	ABC	GA	Modified Bat
200	1300.89	1245.67	1223.67	1156.78	1089.56
400	1500.67	1478.89	1389.89	1245.78	1123.67
600	1765.65	1645.89	1567.78	1356.89	1234.78
800	1800.78	1769.89	1543.67	1456.89	1289.67
1000	2234.89	2145.89	2089.89	1987.89	1876.78
1200	2489.89	2356.78	2245.78	2167.98	2078.89

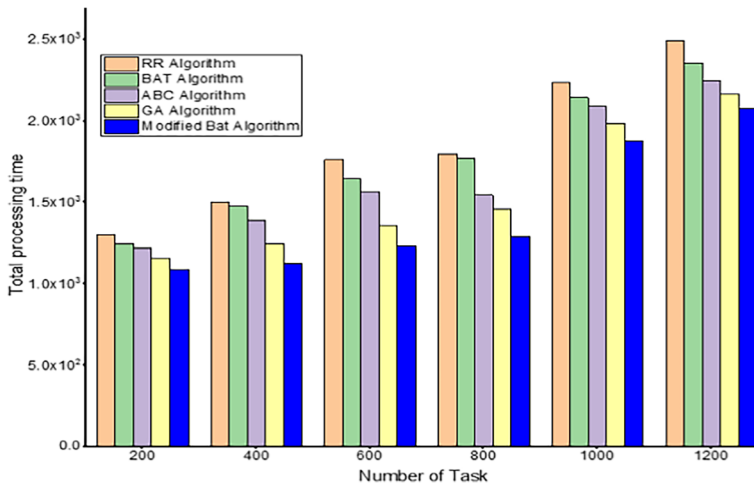


Fig. 7 Total processing time

Table 10 Result analysis of different workflows for degree of imbalance

Degree of imbalance

Task Number	RR	Bat	ABC	GA	Modified Bat
200	56.90	47.45	45.34	43.78	40.89
400	89.90	87.67	80.34	77.45	70.34
600	110.56	100.89	89.78	86.67	80.56
800	170.67	167.67	162.67	160.56	156.45
1000	198.90	190.45	188.67	180.67	176.78
1200	210.67	200.78	190.56	189.67	180.67

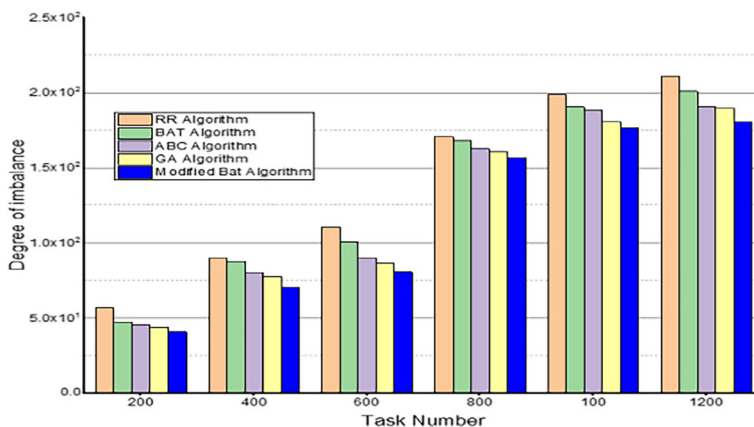


Fig. 8 Degree of imbalance

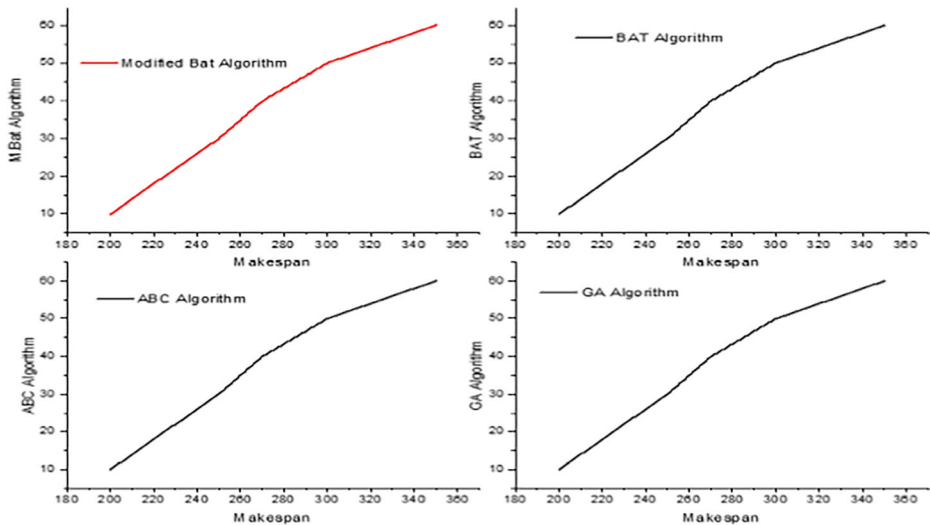


Fig. 9 Result of makespan

Based on the Table 10 most efficient result are, 40.89, 70.34, 80.56, 156.45, 176.78 and 180.67 these are the result of the proposed algorithm as compare to other algorithm. Figure 9 present the result of makespan where it shows it present the different algorithm makespan result.

The inspiration driving this work is to diminish makespan and improve task allocation system in VM for cloud computing in a dynamic environment. Therefore, this study proposed modified Bat algorithm design for improvement in tasks allocation system in VM for cloud computing. The proposed algorithm can greatly reduce the makespan time, increase the utilization of resources and balance the load on each virtual machine. The performance of the proposed algorithm on 120 trial simulations were performed in which tasks number, size of task, VMs number and DC number in different zones are dynamically changed. CloudSim simulation tool was used for the simulation purpose. Based on the results the proposed took less amount of time in each data center during execution.

6 Conclusion

In this paper the proposed algorithm was design for improvement of four parameter which are throughput, makespan, degree of imbalance and processing time in cloud computing. During experiments process different parameter of simulation change like task number, VM and data size to check the proposed algorithm performance. The proposed model results are compared with Bat algorithm ABC algorithm RR algorithm and GA algorithm. From the calculation new technique improves 1.78% improvement in reduction of time consumption of cloud datacenter using load balancing technique. The simulation result shows that purposed algorithm improve in time consumption of datacenter for cloud computing. The proposed load balancing algorithm shows better performance for a large number of assigned tasks and data size as compare to the other standard algorithms. In upcoming paper we implement the proposed algorithm for predication of host in cloud datacenters.

Author contribution All authors are equality contribution for making paper.

Data availability No specific Data are used because it is review papers all those paper which are study and used in this paper are cited in the paper.

Declarations

Competing interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Abdhullah SS, Jyoti K, Sharma S, Pandey US (2016, March) Review of recent load balancing techniques in cloud computing and BAT algorithm variants. In: 2016 3rd international conference on computing for sustainable global development (INDIACom). IEEE, p 2428–2431
2. Adhikari M, Nandy S, Amgoth T (2019) Meta heuristic-based task deployment mechanism for load balancing in IaaS cloud. *J Netw Comput Appl* 128:64–77
3. Afzal S, Kavitha G (2019) Load balancing in cloud computing—a hierarchical taxonomical classification. *J Cloud Comput* 8(1):1–24
4. Alam T, Ullah A, Benaïda M (2022) Deep reinforcement learning approach for computation offloading in blockchain-enabled communications systems. *J Ambient Intell Human Comput* <https://doi.org/10.1007/s12652-021-03663-2>
5. Alihodzic A, Tuba M (2014) Improved bat algorithm applied to multilevel image thresholding. *Sci World J* 2014:1–16
6. Cai X, Gao XZ, Xue Y (2016) Improved bat algorithm with optimal forage strategy and random disturbance strategy. *Int J Bio-Inspired Comput* 8(4):205–214
7. Chaudhury KS (2021) A particle swarm and ant Colony optimization based load balancing and virtual machine scheduling algorithm for cloud computing environment. *Turk J Comput Math Educ (TURCOMAT)* 12(11):3885–3898
8. Cheng TCE, Kovalyov MY (2001) Single machine batch scheduling with sequential job processing. *IIE Trans* 33(5):413–420
9. Cho KM, Tsai PW, Tsai CW, Yang CS (2015) A hybrid meta-heuristic algorithm for VM scheduling with load balancing in cloud computing. *Neural Comput & Applic* 26(6):1297–1309
10. Cui Y, Zhao Z, Ma Y, Dong S (2019) Resource allocation algorithm design of high quality of service based on chaotic neural network in wireless communication technology. *Clust Comput* 22(5):11005–11017
11. Dong S, Jain R (2020) Energy-efficient scheme based on user task characteristic in virtual cloud platform. *Clust Comput* 23(2):1125–1135
12. Mondal B, Dasgupta K, Dutta P (2012) Load balancing in cloud computing using stochastic hill climbing—a soft computing approach. *Procedia Technology* 4:783–789
13. Huang HC (2016) Fusion of modified bat algorithm soft computing and dynamic model hard computing to online self-adaptive fuzzy control of autonomous mobile robots. *IEEE Trans Ind Inform* 12(3):972–979
14. Jian C, Chen J, Ping J, Zhang M (2019) An improved chaotic bat swarm scheduling learning model on edge computing. *IEEE Access* 7:58602–58610
15. Kaur A, Kaur B, Singh D (2019) Meta-heuristic based framework for workflow load balancing in cloud environment. *Int J Inf Technol* 11(1):119–125
16. Kaur T, Saini BS, Gupta S (2019) An adaptive fuzzy K-nearest neighbor approach for MR brain tumor image classification using parameter free bat optimization algorithm. *Multimed Tools Appl* 78(15):21853–21890
17. Kavousi-Fard A, Niknam T, Fotuhi-Firuzabad M (2015) A novel stochastic framework based on cloud theory and S θ S-modified bat algorithm to solve the distribution feeder reconfiguration. *IEEE Trans Smart Grid* 7(2):740–750
18. Khooban MH, Niknam T (2015) A new intelligent online fuzzy tuning approach for multi-area load frequency control: self adaptive modified bat algorithm. *Int J Electr Power Energy Syst* 71:254–261
19. Kintsakis AM, Psomopoulos FE, Symeonidis AL, Mitkas PA (2017) Hermes: seamless delivery of containerized bioinformatics workflows in hybrid cloud (HTC) environments. *SoftwareX* 6:217–224

20. Lin W, Peng G, Bian X, Xu S, Chang V, Li Y (2019) Scheduling algorithms for heterogeneous cloud environment: main resource load balancing algorithm and time balancing algorithm. *J Grid Comput* 17(4): 699–726
21. Meijer C, Grootes MW, Koma Z, Dzigan Y, Gonçalves R, Andela B, van den Oord G, Ranguelova E, Renaud N, Kissling WD (2020) Laserchicken—a tool for distributed feature calculation from massive LiDAR point cloud datasets. *SoftwareX* 12:100626
22. Mishra SK, Sahoo B, Parida PP (2020) Load balancing in cloud computing: a big picture. *J King Saud Univ-Comput Inf Sci* 32(2):149–158
23. Negi S, Rauthan MMS, Vaisla KS, Panwar N (2021) CMODLB: an efficient load balancing approach in cloud computing environment. *J Supercomput* 77:1–53
24. Ouhamme S, Hadi Y, Arifullah A (2020) A hybrid grey wolf optimizer and artificial bee colony algorithm used for improvement in resource allocation system for cloud technology
25. Ouhamme S, Hadi Y, Ullah A (2021) An efficient forecasting approach for resource utilization in cloud data center using CNN-LSTM model. *Neural Comput Appl* 33:1–13
26. Pérez J, Valdez F, Castillo O (2015, May) Modification of the bat algorithm using fuzzy logic for dynamical parameter adaptation. In 2015 IEEE congress on evolutionary computation (CEC). IEEE p 464–471
27. Pérez J, Valdez F, Castillo O (2017) Modification of the bat algorithm using type-2 fuzzy logic for dynamical parameter adaptation. In: *Nature-inspired design of hybrid intelligent systems*. Springer, Cham, pp 343–355
28. Powers C, Bassman L, Linker TM, Nomura KI, Gulania S, Kalia RK, Nakano A, Vashishta P (2021) MISTIQS: an open-source software for performing quantum dynamics simulations on quantum computers. *SoftwareX* 14:100696
29. Priya V, Kumar CS, Kannan R (2019) Resource scheduling algorithm with load balancing for cloud service provisioning. *Appl Soft Comput* 76:416–424
30. Sharma S, Luhach AK, Sinha SA (2016) An optimal load balancing technique for cloud computing environment using bat algorithm. *Indian J Sci Technol* 9(28):1–4
31. Srivastava P, Shukla A, Bansal A (2021) A comprehensive review on soil classification using deep learning and computer vision techniques. *Multimed Tools Appl* 80:1–28
32. Tripathi D, Edla DR, Kuppli V, Dharavath R (2020) Binary BAT algorithm and RBFN based hybrid credit scoring model. *Multimed Tools Appl* 79(43):31889–31912
33. Ullah A (2019) Artificial bee colony algorithm used for load balancing in cloud computing. *IAES Int J Artif Intell* 8(2):156
34. Ullah A, Nawi NM (2020) Enhancing the dynamic load balancing technique for cloud computing using HBATAABC algorithm. *Int J Model Simul Sci Comput* 11(05):2050041
35. Ullah A, Nawi NM (2021) An improved in tasks allocation system for virtual machines in cloud computing using HBAC algorithm. *J Ambient Intell Human Comput* <https://doi.org/10.1007/s12652-021-03496-z>
36. Ullah A, Nawi NM, Khan MH (2020) BAT algorithm used for load balancing purpose in cloud computing: an overview. *Int J High Performa Comput Netw* 16(1):43–54
37. Xu M, Tian W, Buyya R (2017) A survey on load balancing algorithms for virtual machines placement in cloud computing. *Concurr Comput Pract Exp* 29(12):e4123
38. Yue X, Zhang H (2020) Modified hybrid bat algorithm with genetic crossover operation and smart inertia weight for multilevel image segmentation. *Appl Soft Comput* 90:106157
39. Yue S, Zhang H (2021) A hybrid grasshopper optimization algorithm with bat algorithm for global optimization. *Multimed Tools Appl* 80(3):3863–3884