International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075, Volume-8 Issue-9, July 2019

Crow Search based Scheduling Algorithm for Load Balancing in Cloud Environment

Harvinder Singh, Sanjay Tyagi, Pardeep Kumar

Abstract*: Task scheduling is a vital aspect in computer science, as it is the essence of how a computer executes various tasks and performs the related activities with accuracy and efficiency. In cloud computing, task scheduling is a typical NP-hard problem and scientists have been attempting to handle this problem for quite a long time. Although it is anything but difficult to accomplish a global optimum solution utilizing the ant colony algorithm and achieve promising outcomes. This research paper proposes a crow search based load balancing algorithm (CSLBA) for multi-objective task scheduling environment which concentrates on allocating best suitable resources for the task to be implemented with the consideration of various parameters like average makespan time (AMT), average waiting time (AWT) and average data center processing time (ADCPT). The present work provides a comparative analysis of proposed algorithm and Ant Colony Optimization based load balancing algorithm (ACOLBA). The experimentation is performed in steps and comparative analysis proved that proposed CSLBA is the optimal technique among the other scheduling technique considered in this research paper.

Index Terms*: Virtual machine, Data center, Cloud computing, Crow search algorithm, Ant colony algorithm, Load balancing.

I. INTRODUCTION

Cloud computing is “a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction” [1]. Through the web, gigantic pools of systems having variety of storage and processing assets are provided to the clients, this sort of shared framework is referred to as cloud computing. Anything as a service can be provided to the clients and this capability of the cloud can be counted as its real competency. The tasks submitted for computations by multiple users should be optimizely mapped to the limited number of available resources. This mapping problem has been placed in the category of NP-hard problems [2]. For such sort of issues, there are no algorithms which may deliver ideal arrangement inside polynomial time. Meta-heuristic based algorithms like an ant colony optimization scheduling algorithm have obtained enormous prominence in the previous years because of its proficiency and viability to tackle vast and complex problems by giving close to ideal arrangements inside reasonable time. In cloud computing, server solidification is applied at data centers to improve the productivity of the assets [3]. Many virtual machines that are running on various data centers will undoubtedly use the assets effectively. More often than not, cloud assets are under-utilized because of the poor mapping between the tasks and the resources of data center. The present research proposed CSLBA which is used for scheduling of hundreds of heterogenous tasks and compare the results produced by ACOLBA for the same number and types of tasks on certain fixed scheduling parameters. This could not be done with the real-time cloud resources and infrastructure because of time and resources limitations. This results in the usage of cloud simulator called CloudSim toolkit package to simulate, model and experiment on designing efficient and optimum cloud service infrastructures.

II. RELATED WORK

In cloud environment, users can submit multiple tasks which requires heterogeneous resources (e.g. CPU, I/O, Memory etc.) for their implementation. There are various factors that needs to be addressed while mapping of resources as per the demands of the task, like Quality of Service (QoS), energy usage and time taken to complete a job, etc. These parameters are defined by the users and documented in a Service Level Agreements (SLAs). Task to resource mapping techniques are essential to enhance the comprehensive performance of cloud system. In the recent past, the task scheduling issues has been talked by many researchers, but no one has come up with the potential solution yet. In 2017, Satpathy et al. suggested heuristic based approaches for task-resource mapping and focused on increasing energy efficiency of a data center by applying effective scheduling techniques in cloud environment [4]. The load balancing issues in cloud environment are handled by applying Stochastic Hill-Climbing algorithm. Wang et al. and Madhu et al. both recommended a scheduling policy on load balancing of resources using a genetic algorithm to strengthen the overall system performance [5][6]. To measure the performance and behavior of the system, their suggested technique considered previous experiences and the present state of work in advance. Nishant et al. and Gupta et al. introduced an ant colony based load balancing method in cloud environment to lessen makespan involved in the execution of tasks submitted by multiple users[7][8]. ABC algorithm is suggested by Karaboga et al., to work out the scheduling issues and to identify the most relevant parameters in the changing

Revised Manuscript Received on
Harvinder Singh, Department of Virtualization, School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India
Sanjay Tyagi, Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, Haryana, India
Pardeep Kumar, Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, Haryana, India.

environment [9]. Kiyarazm et al. and Aslanzadeh et al. addressed the job scheduling issue and suggested changed PSO algorithm, which received an affirmative response from the cloud consumers [10][11]. In cloud environment, various simulation toolkits are available for mimicking and computing the performance of scheduling and load balancing algorithms. Calheiros et al. introduced the CloudSim simulation tool for design and simulation of cloud-based data center environment, which enables to test cloud system and identify the behavior of applications submitted by the consumers on various parameters [12]. The mapping of multiple tasks on the limited number of resources can be done by reflecting the costs for every task by considering the difference in needs of the tasks for the requested resources. To do so, a resource cost model is recommended along with a multi-objective optimization-scheduling design that deals with both the performance and the budget parameters.From the above literature, it is found that so far the research could not provide a promising mapping of the demands of multiple tasks submitted by the cloud consumers, to the limited number of resources available i.e. one task can require higher computational and processing power from the CPU whereas another task might require higher storage instead. So it is significant to address these kind of scheduling issues and recommend an optimum scheduling strategy for the same.

A. *Challenges*

1. In cloud environment, many researchers have suggested numerous meta-heuristic algorithms to deal with task scheduling issues. But somehow an optimum task scheduling algorithm that should revise its scheduling approach corresponding to the changes of the tasks submitted by the cloud consumers, is the need of the hour.
2. In the recent past, researchers have concentrated on optimizing a specific objective (i.e. mimizing execution time), but few of them optimize over two objectives (i.e. minimizing execution time, minimizing cost, load balancing etc) at a point of time. Thus, it is an excellent thought to test the effect of two or more objectives on scheduling issue in cloud environment.
3. In cloud environment, the parameters like scalability and trustworthiness should be taken care off while designing scheduling optimization algorithms.
4. The major issue is to map a batch of tasks submitted by the cloud consumers to the VMs from the cloud providers, so that execution time of workload is decreased to minimal optimized time.

B. *Problem Statement*

Cloud computing is nowadays providing various services but still there are several issues which requires to be focused on as far as the optimal management of the resources is involved. Among those issues, task to resource mapping is one of the main issue. Task resource mapping problems, "which are concerned with searching for optimal (or near-optimal) real-time and predictive schedules subject to several constraints". The effectiveness of scheduling method in cloud environment depends on how efficiently various tasks are projected to the finite number of resources available. A new multi-objective scheduling approach has to be introduced which can give an optimum result to task scheduling issue and enhance the overall performance of cloud environment by analyzing multiple criteria like execution time, makespan, cost, power utilization, load available and resource usage and so on.

C. *Objective of the Research Paper*

To resolve the problems related to the scheduling issues in cloud environment, stated in the problem statement section, the present research paper has the following goals:

1. To study and explore various existing meta-heuristic scheduling optimization methods.
2. To introduce a meta-heuristic scheduling optimization method which address the task scheduling issue in cloud environment.
3. To compare and analyse quantitatively, the recommended meta-heuristic scheduling optimization algorithm with the existing algorithms based on performance metrics and validate the results by using cloud simulation tool.

III. OPTIMIZATION PROBLEM FORMULATION

Load balancing is defined as the technique of distributing the task submitted by the users from overloaded VM of a particular datacenter to an available underloaded VM of datacenter. The objective function and the performance constraints are presented in the further sections.

A. *Objective Function*

The objective of this work is to divide the load equally among all the VMs of hosts contained in a particular data center. The objective function should be minimized or maximized depending upon the performance contraints taken into consideration. In present research, the aim is to minimize the objective function comprises of parameters like ADCPT, AMT and AWT.

$$OF = v_1 w_1 + v_2 w_2 + v_3 w_3 \tag{1}$$

$$where, \sum_{i=1}^{3} v_i = 1 \tag{2}$$

v_i represents the scaling factor, w_1 is the ADCPT, w_2 is the AMT and w_3 is the AWT of the data center.

B. *Performance Constraints*

An optimum load balancer should map the computation load to resources in such a manner that the comprehensive performance of the system must be stepped up. The present research paper test the suggested CSLBA based on following metrics [13]:

Makespan Time

Makespan time refers to the total time that elapses from the start to finish of the task. Makespan time depends on the execution time of different VMs. The execution time (ET_n) of n^{th} virtual machine (vm) is based on the decision variable VR_{mn}, where

$$VR_{mn} = \begin{cases} 1 & \text{if} vm_n \leftarrow t_m \\ 0 & \text{if} vm_n \nleftarrow t_m \end{cases} \tag{3}$$

And the $ET_n, 1 \le j \le k$ is calculated as follows.

$ET_n = \sum_{m=1}^{k} VR_{mn} \times$

ETC_{mn} (4)

The makespan (MS)[14] is the maximum of ET_n i.e.,

$$MS = Max.[ET_n]_{n=1}^{k} \quad (5)$$

Waiting Time

Waiting time is the distinction between the execution start time and submission time of a particular task.The waiting time (WT)[15] of m_{th} task is the completion time (CT) of $(m-1)_{th}$ task waiting in the queue of n_{th} virtual machine i.e.

$$WT_{nm} = CT_{n(m-1)} \quad (6)$$

Data Center Processing Time

It is the time taken by the data center to execute the request submitted by the users. The data center processing time $((DCPT)$[15] of a m_{th} task at n_{th} virtual machine is the ratio of size of the task to the capacity of the n_{th} virtual machine i.e.

$$DCPT_{mn} = \frac{size_m}{capacity_n} \quad (7)$$

IV. PROPOSED SYSTEM MODEL

In cloud computing environment, for providing the optimal services to the customers from different geographical locations of the world, nowadays, the large number of data centers are deployed and available all over the world and are accessible using internet. Every data center in cloud environment comprises many servers of varied capacities of virtual machines. Each virtual machine can execute multiple tasks depending upon the mapping of its capacity parameters and the needs of the requested tasks. However, existing system does not provide the effective scheduling mechanism of the multiple tasks submitted by the customers among the VMs in data centers to achieve reasonable QoS levels. In this research paper, a CSLBA is introduced, which is capable for productive allotment of user tasks onto various available resources with a focus to optimize data center load balancing, minimizing ADCPT, minimizing the AMT, minimizing energy consumption, minimizing cost of data center and minimizing AWT.

A. *Proposed Crow Search Based Load Balancing Algorithm*

Crows are seen being among the world's most wise birds. As a group, crows present outstanding illustrations of intellect and score high on knowledge assessments. Crows interact in subtle forms, can shield and fetch food across seasons. A CSLBA is introduced in this research paper and the capabilities of crows are projected to solve the real world multimodal scheduling optimization issue in terms of its representation, operators, control parameters, evolutionary mechanism, the performance metric, areas of applications as presented below [16].

- **Algorithm:** CSLBA Algorithm
- **Representation:** D dimensional search space where d represents the count of decision variables.
- **Operators:** Maximum number of iterations $iter_{max}$, awareness probability (AP), flock size (N) & flight length (fl).
- **Control Parameters:** Awareness probability (AP) & flight length (fl).
- **Evolutionary Mechanism:**

The location update equations for CSLBA algorithm is given in equation 8 and 9

$$LOC[i,t+1] = \begin{cases} LOC[i,t] + k[i] * b[i,t] * (HIDE[i,t] - LOC[i,t]) & k[j] \geq P[i,t] \\ random\ location & otherwise \end{cases} \quad (8)$$

$$HIDE[c,t+1] = \begin{cases} LOC[c,t+1] & \Phi(LOC[c,t+1]) is\ better\ than\ \Phi(HIDE[c,t]) \\ HIDE[c,t] & otherwise \end{cases} \quad (9)$$

where, C is the set of all crows, LOC is the table of all crows locations, HIDE represents the table of hiding places, LOC $[i,t]$ represents location of i^{th} crow at t^{th} instance, b[i,t] is the balance factor of i^{th} crow at t^{th} instance, HIDE $[i,t]$ represents the best known hiding location of i^{th} crow at t^{th} instance, P[i,t] is the awareness probability of i^{th} crow at t^{th} instance, V represents the victim crow.$k[j]$represents random number for crow j, Φ represents the objective function and t_{max} represents the maximum number of iterations.

- **Performance Metric:** Data center processing time, makespan time, power consumption, cost of data center, waiting time and data center loading.
- **Areas of Application:** Solving complex engineering design problem, multi-objective scheduling optimization issues.

The pseudo code for CSLBA is presented in Algorithm 1. The suggested approach endeavors to recreate the intelligent strategy of the crows to determine the solution of scheduling optimization issues. The behaviour of the flocks of crow has many similitudes with an optimization process. As illustrated by this act, crows conceal their plethora of food in concealing spots of the environment and recover the put-away food when it is required.

Algorithm 1 CSLBA Algorithm

1: Fill the first row of the LOC table by random locations for each crow
2: Initialize HIDE table by HIDE ← LOC.
3: **if** $(t < t_{max})$ **then**
4: $\forall c \in C$
5: $V \leftarrow rand(|C|)$
6: Set awareness propbability, P
7: Update LOC according to equation 8
8: Check feasibility of the new location
9: Update the new row of HIDE according to the equation 9
10: $t \leftarrow t+1$
11: Repeat from Step 3
12: End

For finding better sources of food, there is a behaviour of the crow, that it will tail other crows in the flock. It is a very tedious task for a crow to find out the source of food covered up by other crow as if a particular crow finds another is tailing it, the crow endeavors to deceive that crow by heading off to some another location of the environment. From an optimization viewpoint, the following mapping of the crowsbehaviour is done.

- searchers→crow

- seeking space→environment
- feasible solution→each position of the environment
- objective function→quality of the food source
- global solution of the issue→best food source of the environment

In the further sections of this paper, the proposed CSLBA is analyzed and compared quantitatively with ACOLBA based on various performance optimization metrics.

V. PERFORMANCE EVALUATION

This section of the research paper presents the results and discussion of the quantitative analysis of proposed CSLBA and existing algorithm. Here, the existing ACOLBA [17][18] in cloud environment used for comparison with the proposed algorithm for analysing its effectiveness over parameters like ADCPT, AMT and AWT.

A. *Experimental Setup*

The quantitative analysis of the suggested CSLBA in the cloud is done using cloudsim simulator and cloud analyst tool [19]. For experimentation, the user base configuration comprises 07 different user bases distributed over 6 regions namely Region 0 to Region 5. Each of the user base has 60 requests\slash user\slash hour with peak hour's start and end time is 3 and 9 GMT, respectively. The data size\slash requests (in bytes) is 100 with average peak users and average off-peak users as 1000 and 100 respectively for each of the user bases. The application deployment configuration comprises 07 different data centers namely DC1 to DC7 each having 5 virtual machines (VMs) with an image size of 10000. The memory and bandwidth (BW) of each data center is 512 and 1000, respectively. The data center configuration comprises 07 different data centers distributed over 6 regions namely Region 0 to Region 5. Each of the data center has x86 architecture (Arch) and Linux as its operating system (OS) with 5 physical hardware units. The cost per VM and data transfer cost is 0.1 dollar per hour and 0.1 dollar per gigabit, respectively. Meanwhile, the cost of memory and storage is 0.05 dollar per second and 0.1 dollar per second, respectively. For the experimentation, it is assumed that the tasks are mutually independent and are not preemptive.

B. *Comparative Analysis*

In cloud environment, the experiments on the suggested CSLBA, and the ACOLBA are carried out using three distinct service broker policies i.e. closest data center policy (CDCP), optimize response time policy (ORTP) and the reconfigure dynamically with load policy (RDLP) based on parameters like ADCPT, AMT and AWT. This segment of a research paper portrays the results of the experimentation performed.

Makespan Time

The aim of the simulation is to compute the average makespan time and test the proposed CSLBA against ACOLBA algorithm. The algorithm that results minimum AMT under each of the service broker policy is considered to be the optimal meta-heuristic scheduling algorithm.

Table IAMT of ACOLBA and CSLBA for CDCP, ORTP and RDLP policies

	CDCP	ORTP	RDLP
ACOLBA	156.68	78.19	84.92
CSLBA	154.71	71.46	82.69

The values of AMT for both meta-heuristic algorithm under test has been calculated and tabularized in Table I with the corresponding graphical representation shown in Fig. 1. The average of the makespan time is obtained after 100 independent executions. The graphical representation depicts the variations of AMT with the changes in the service broker policies.

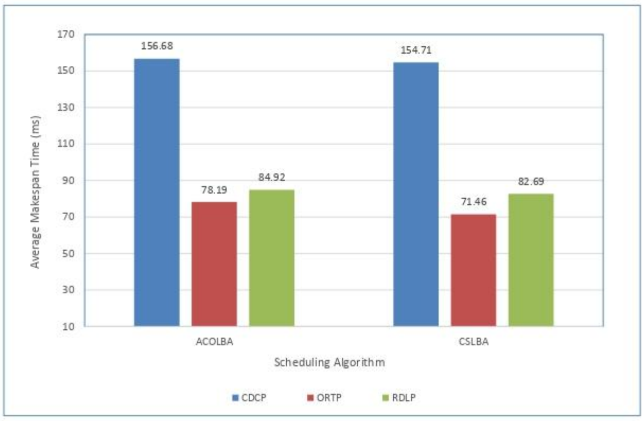

Fig. 1Graphical representation of AMT of ACOLBA and CSLBA for CDCP, ORTP and RDLP policies

When implemented for CDCP, the value of AMT for ACOLBA is 156.68 which is further reduced by 1.26% that is 154.71 in case of proposed CSLBA. ACOLBA exhibits the AMT of 78.19 as compared to 71.46 of proposed CSLBA, which shows a gradual decrease of 8.61%, when implemented using ORTP. However, for RDLP, ACOLBA has AMT of 84.92 as compared to 82.69 of proposed CSLBA, which exhibits a decrease of 2.63%.

o**Waiting Time of Data Center**

The aim of the simulation is to compute the average waiting time (AWT) of the data center and test the proposed CSLBA against ACOLBA. The algorithm that possess minimum AWT of the data center under each of the service broker policy is considered to be the optimal meta-heuristic scheduling algorithm. The values of AWT of the data center for both meta-heuristic algorithm under test has been calculated and tabularized in Table II. The AWT of the data center is obtained after 100 independent executions.

Table II AWT of Data Center in case of CDCP, ORTP and RDLP

	CDCP		ORTP		RDLP	
	ACOlBA	CSLBA	ACOLBA	CSLBA	ACOLBA	CSLBA
DC1	0	0	0.444	0.391	0.545	0.52
DC2	0.223	0.201	0.484	0.3	0.664	0.59
DC3	1	0.9	0.551	0.501	0.598	0.54
DC4	0.479	0.463	0.474	0.408	0.567	0.523
DC5	0.492	0.48	0.512	0.502	2	0.862
DC6	0	0	0	0	0	0
DC7	0	0	1	0.378	1.486	1.392

The graphical representation depicts the variations of AWT of the data center with the changes in the service broker policies.

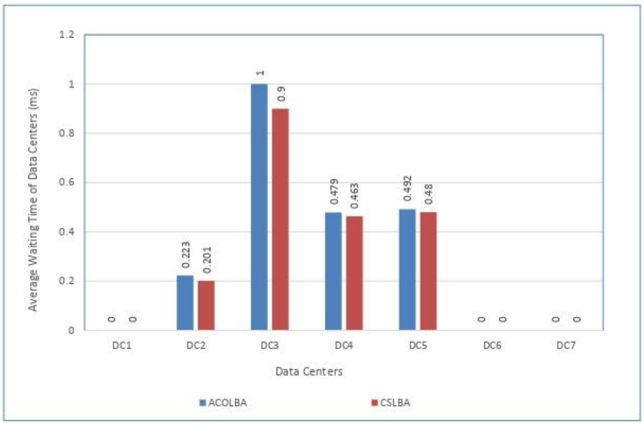

Fig. 2 Graphical representation of AWT of Data Centers for CDCP

The graphical representation of AWT of the data center in case of CDCP is shown in Fig. 2. When implemented for CDCP, the value of AWT of the data center DC1 to DC7 for ACOLBA is 0, 0.223, 1, 0.479, 0.492, 0, 0 which is further reduced by 0%, 9.87%, 10%, 3.34%, 2.44%, 0%, 0% that is 0, 0.201, 0.9, 0.463, 0.48, 0, 0 in case of proposed CSLBA.

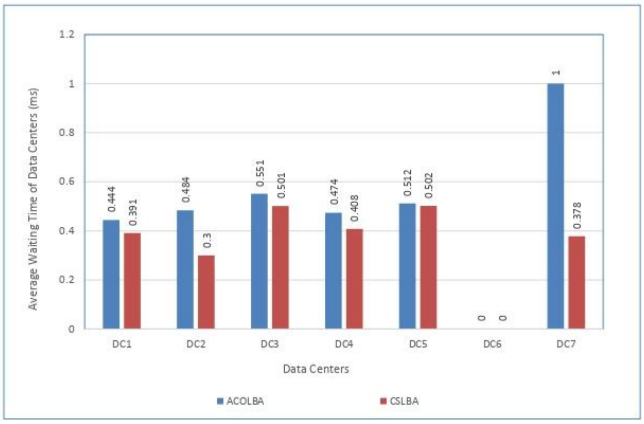

Fig. 3 Graphical representation of AWT of Data Centers for ORTP

The graphical representation of AWT of the data center in case of ORTP is shown in Fig. 3. ACOLBA exhibits the AWT of data center DC1 to DC7 is 0.444, 0.484, 0.551, 0.474, 0.512, 0, 1 as compared to 0.391, 0.3, 0.501, 0.408, 0.502, 0, 0.378 of proposed CSLBA, which shows a decrease of 11.94%, 38.02%, 9.07%, 13.92%, 1.95%, 0%, 62.20% when implemented using ORTP.

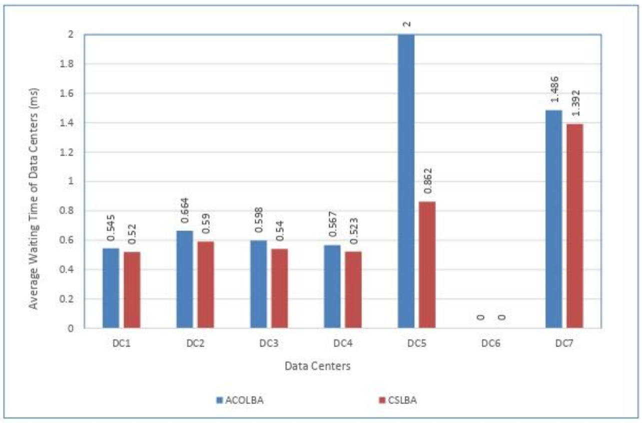

Fig. 4 Graphical representation of AWT of Data Centers for RDLP

The graphical representation of AWT of the data center in case of RDLP is shown in Fig. 4. However, for RDLP, ACOLBA has AWT of the data center DC1 to DC7 of 0.545, 0.664, 0.598, 0.567, 2, 0, 1.486 as compared to 0.52, 0.59, 0.54, 0.523, 0.862, 0, 1.392 of proposed CSLBA, which exhibits a decrease of 4.59%, 11.14%, 9.70%, 7.72%, 56.90%, 0%, 6.33%.

Data Center Processing Time

The aim of this simulation is to compute the average data center processing time of the algorithms under test. The algorithm that results minimum data-center processing time under each of the service broker policy, is considered to be the optimal meta-heuristic scheduling algorithm.

Table III ADCPT of ACOLBA and CSLBA for CDCP, ORTP and RDLP policies

	CDCP	ORTP	RDLP
ACOLBA	14.18	7.25	13.88
CSLBA	14.05	0.47	13.22

The values of ADCPT for both meta-heuristic algorithm under test has been calculated and tabularized in Table III with the corresponding graphical representation shown in Fig. 5. The average of the data center processing time is obtained after 100 independent executions. The graphical representation depicts the variations of ADCPT with the changes in the service broker policies.When implemented for CDCP, the value of ADCPT for ACOLBA is 14.18 which is further reduced by 0.92% that is 14.05 in case of proposed CSLBA. ACOLBA exhibits the ADCPT of 7.25 as compared to 0.47 of proposed CSLBA, which shows a gradual decrease of 93.52% when implemented using ORTP.

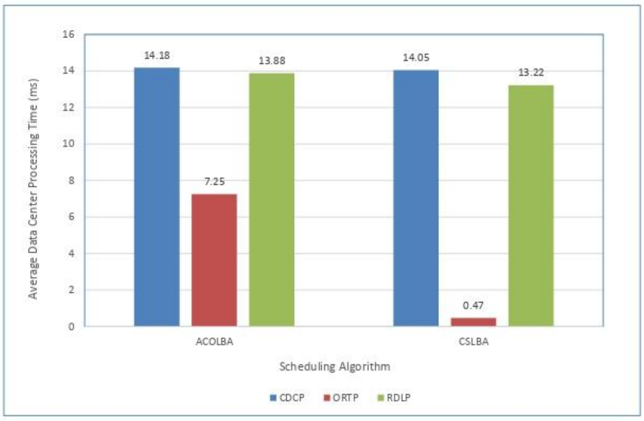

Fig. 5 Graphical representation of ADCPT of ACOLBA and CSLBA for CDCP, ORTP and RDLP policies

However, for RDLP, ACOLBA has ADCPT of 13.88 as compared to 13.22 of proposed CSLBA, which exhibits a decrease of 4.76%.

C. *Comparative Discussion*

For finding an optimal scheduling algorithm among ACOLBA, and the proposed CSLBA, the parameters like ADCPT, AMT and AWT is calculated. The ADCPT under each of the service broker policy should be the minimum to find the optimal meta-heuristic scheduling algorithm. During experimentation, it has been proved that the proposed CSLBA has reduced the ADCPT for each of the service broker policy to a great extent. Hence, it results as the optimum during comparison. The AMT under each of the service broker policy should be the minimum to find the optimal meta-heuristic scheduling algorithm. During experimentation, it has been proved that the proposed CSLBA has significantly decreased the AMT for each of the service broker policy. Thus, the comparison reveals that proposed CSLBA is optimum. The AWT of data center under each of the service broker policy should be the minimum to find the meta-heuristic scheduling algorithm. During experimentation, it has been proved that the proposed CSLBA has reduced the AWT of data center for each of the service broker policy to certain limits. Hence, it results as the optimum during comparison. Thus, the CSLBA results as the optimum meta-heuristic scheduling algorithm.

VI. CONCLUSION & FUTURE SCOPE

This research paper is focussed to have a deep insight to various existing task-resource mapping algorithms in cloud environment. For this purpose, a CSLBA is proposed for solving the task scheduling problem under the computing environment, where the number of data centers and user jobs changes dynamically. The CSLBA is simulated using CloudSim simulator and then the comparative analysis among existing load balancing algorithm is done, based on different parameters like ADCPT, AMT and AWT. The experimental results illustrated that the proposed CSLBA out-performed the ACOLBA for all the parameters considered for comparison and thus it is figured out to be the optimal load balancing scheduling algorithm. In the future, the proposed CSLBA can be hybridized with other load balancing meta-heuristic scheduling algorithm for solving the multi-objective scheduling optimization problem.

REFERENCES

1. [1] P. Mell and T. Grance, “On-demand self-service.,” *Nist*, vol. 15, pp. 10–15, 2009.
2. [2] F. Chang, J. Ren, and R. Viswanathan, “Optimal Resource Allocation in Clouds,” *2010 IEEE 3rd International Conference on Cloud Computing*, pp. 418–425, 2010.
3. [3] M. Kumar and S. C. Sharma, “Dynamic load balancing algorithm for balancing the workload among virtual machine in cloud computing,” *Procedia Computer Science*, vol. 115, pp. 322–329, 2017.
4. [4] A. Satpathy, S. K. Addya, A. K. Turuk, B. Majhi, and G. Sahoo, “A resource aware VM placement strategy in cloud data centers based on crow search algorithm,” *2017 4th International Conference on Advanced Computing and Communication Systems, ICACCS 2017*, 2017.
5. [5] T. Wang, Z. Liu, Y. Chen, Y. Xu, and X. Dai, “Load balancing task scheduling based on genetic algorithm in cloud computing,” *Proceedings - 2014 World Ubiquitous Science Congress: 2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing, DASC 2014*, pp. 146–152, 2014.
6. [6] B. R. Madhu and P. Chandra, “Minimizing Execution Time of Cloudlets through Optimal Allocation of Virtual Machines Using Genetic Algorithm,” pp. 213–217, 2016.
7. [7] E. Gupta and V. Deshpande, “A technique based on ant colony optimization for load balancing in cloud data center,” *Proceedings - 2014 13th International Conference on Information Technology, ICIT 2014*, pp. 12–17, 2014.
8. [8] P. Xu, G. He, Z. Li, and Z. Zhang, “An efficient load balancing algorithm for virtual machine allocation based on ant colony optimization,” *International Journal of Distributed Sensor Networks*, vol. 14, no. 12, 2018.
9. [9] D. Karaboga and B. Gorkemli, “Solving Traveling Salesman Problem by Using Combinatorial Artificial Bee Colony Algorithms,” *International Journal on Artificial Intelligence Tools*, vol. 28, no. 01, p. 1950004, 2019.
10. [10] O. Kiyarazm, M.-H. Moeinzadeh, and S. Sharifian-R, “A New Method for Scheduling Load Balancing in Multi-processor Systems Based on PSO,” *2011 Second International Conference on Intelligent Systems, Modelling and Simulation*, pp. 71–76, 2011.
11. [11] S. Aslanzadeh and Z. Chaczko, “Load balancing optimization in cloud computing: Applying Endocrine-particale swarm optimization,” *IEEE International Conference on Electro Information Technology*, vol. 2015-June, pp. 165–169, 2015.
12. [12] R. N. Calheiros, R. Ranjan, C. A. F. De Rose, and R. Buyya, “CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services,” pp. 1–9, 2009.
13. [13] M. Kalra and S. Singh, “A review of metaheuristic scheduling techniques in cloud computing,” *Egyptian informatics journal*, vol. 16, no. 3, pp. 275–295, 2015.
14. [14] S. K. Mishra, B. Sahoo, and P. P. Parida, “Load balancing in cloud computing: A big picture,” *Journal of King Saud University - Computer and Information Sciences*, 2018.
15. [15] M. Adhikari and T. Amgoth, “Heuristic-based load-balancing algorithm for IaaS cloud,” *Future Generation Computer Systems*, vol. 81, pp. 156–165, 2018.
16. [16] A. Askarzadeh, “A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm,” *Computers and Structures*, vol. 169, no. May, pp. 1–12, 2016.
17. [17] K. Li, G. Xu, G. Zhao, Y. Dong, and D. Wang, “Cloud Task Scheduling Based on Load Balancing Ant Colony Optimization,” *2011 Sixth Annual Chinagrid Conference*, pp. 3–9, 2011.
18. [18] K. Nishant *et al.*, “Load Balancing of Nodes in Cloud Using Ant Colony Optimization,” *2012 UKSim 14th International Conference on Computer Modelling and Simulation*, pp. 3–8, 2012.
19. [19] V. N. Volkova, L. V Chernenkaya, M. Hajali, E. N. Desyatirikova, A. Khodar, and A. Osama, “Load Balancing in Cloud Computing,” pp. 387–390, 2018.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

AUTHORS PROFILE

Harvinder Singh received his M.Tech degree in 2009 and B.Tech degree in 2005 from Kurukshetra University, Kurukshetra Presently he is working as Assistant Professor in Department of Virtualization, School of Computer Science, University of Petroleum and Energy Studies, Dehradun. He has also served as Assistant Professor in Department of Computer Science & Engineering, HCTM Technical Campus, Kaithal, Haryana India. His research interest lies in Cloud Computing, Software Project Management, Internet of Things. He has published 5 research papers in Referred Journals and International Conferences. He has also served as TPC member of International Conferences.

Dr. Sanjay Tyagi completed his Masters degree and Ph.D. degree from Kurukshetra University. The author is currently working with Kurukshetra University, India for 27 years. His area of research includes Software Testing, Cloud Computing, Information Systems, MANETs. He has more than 70 research publications in Scopus Indexed / referred Journals / International conferences to his credit. He has attended more than 65 International and National Conferences / Seminars / Workshops.He is on the review panel of various International and National Journals &Conferences.He has been the member of technical programme committees / organizing committees of various International and National Conferences.

Dr. Pardeep Kumar received his PhD in Computer Science & Applications from Kurukshetra University, Kurukshetra in 2014. He received his M.Sc.(Computer Science) degree from Kurukshetra University, Kurukshetra in 1997. Also he received his M.Sc.(Statistics) degree from Kurukshetra University, Kurukshetra in 1995. Presently he is working as Associate Professor in Department of Computer Science and Applications, Kurukshetra University, Kurukshetra. His research interest lies in Optimisation, Cloud Computing and Network Routing. He has published more than 30 research papers in SCI Journals, Referred Journals and International Conferences. He has served as TPC member of many International Conferences. He has also chaired many sessions at various National & International Conferences.