Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

Post-migration VM model with Task Scheduling using Swarm Optimization for Load Balancing

Authors Name: **Afreen Bari**
Department: Computer Science & Engineering
Shri Jagdishprasd Jhabarmal Tibrewala University
Rajasthan, India
afreenabdulbari@gmail.com

Guide Name: **Dr. Prasadu Peddi**
Department: Computer Science Engineering
Shri Jagdishprasd Jhabarmal Tibrewala University
Rajasthan, India
peddiprasad37@gmail.com

ABSTRACT

Cloud computing has evolved as a methodology that grazes operations by automatically assigning virtual machines. Consumers compensate for programmes, based on their demand. There are also problems with a cloud service. One of the key challenges is the post-migration VM load balancing model, which suffers from many problems, including premature convergence, reduced convergence schedules, random solutions at first, and a native optimal solution. The proposed method considered the Post-migration VM Task Plan Model utilising Swarm Optimization for Load Balancing to resolve the heuristic approach problem that has already been discussed. The suggested solution focuses on the Particle Swarm algorithm for mutation-based load balancing between data centres. In this case, an efficient load balancing algorithm is built, which reduces performance parameters such as post migration VM model time and improves cloud health.

Keywords : Cloud Computing , Load balancing , Swarm Optimization for Load Balancing , Post-migration VM model.

I. INTRODUCTION

Cloud computing integrates computer concepts that are complementary and distributed, yet individuals may use the Internet to exchange resources1. Cloud vendors, service providers and customers are three separate providers of SaaS-based applications. Cloud platforms utilise virtualization technology to provide their users with computer resources for their virtual machines (VMs). On the other hand, service providers gain from these VMs in order to offer app-level functionality to consumers. Service providers use timing techniques to assign customer tasks to VMs, reduce response time, promote service quality and maximise the assets. The work algorithm is also one of the core components of cloud technologies.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

1 Such methods may be adapted to cloud settings where they have not been previously designed for a variety of computing environments. There are, for example, many discreet, standardised cluster computers that work together as a single device, utilising local resources, while the cloud ecosystem consists of a set of heterogeneous, distributed resources that can be accessed by all consumers. Therefore, a scheduling approach that could be useful to a cluster would not be appropriate for cloud environments.

The growth in heterogeneous VMs and the diversity of activities contributes to a wide number of work arrangements. The discovery of an effective permutation with a minimum completion period between all permutations is an NP-hard problem. A variety of similar experiments have been performed utilising meta-heuristic algorithms in combination with cloud-based scheduling; however, This paper proposes a scheduling strategy based on the PSO algorithm2 strengthened by the load-balancing process, which makes the optimum solution converge more easily.

In particular, a rational load balancing approach and a meta-heuristic algorithm can be used to consider two goals for the advantage of service providers (resource use) and to increase customer loyalty (make span reduction). 1. Provide the necessary computer software allocation with an improved load balancing algorithm to ensure the compatibility of any programme. We've decided on the VM with optimal compatibility for hosting requests since our load balancing algorithm.

II. RELATED WORK

Project management is called cloud computing expression, the optimal delivery of requests to data centre remedies. Applications are assigned to various groups of services, provided the support they will require. Two classes are present: limitations identified by the customer and limitations defined by the supplier. that use heuristic/meta-heuristic algorithms, and (2) those that mix the algorithm of heuristic/meta-heuristic with non-heuristic algorithms. In each of these groups, we discuss some of the more important behaviour in some of the cloud-balancing algorithms.

Dave, B. et al[1] A PSO was submitted to further resolve the problem of cloud storage allocation and load management facilities. This work was tested on Xen Server and was a rather positive result of the proposed algorithm. Compare and balance algorithms to the success of PSO.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

S. Wang, et al[2] This paper optimises Spark's current load-balancing strategy and provides a task-dependent assignment algorithm based on the computational utility of each node in the Spark cluster. on genetic algorithm and particle swarm optimization (TENAA).

T. M. Shami et al[3] The suggested PSO approach achieves greater productivity in terms of output attainable. In addition, the suggested approach would have user equity by managing the load per BS.

H. Mohamad et al[4] The proposed load schudule approach is developed with MATLAB and tested using the Distributor System PSCAD programme. The simulation findings showed that the proposed PSO-based approach could stabilise the frequency of the system by minimising the load shedding.

Dandan Zhang, et al[5] The empiric feature of defining the load balancin model was the optimization of the weighted sum of the economic cost and balance matrix, and the model was overcome by a nonlinear adaptive particle swarm optimization algorithm. The results of the simulation of a practical system demonstrate that the optimal load removal strategy can be quickly achieved.

Durgesh Patel et al[6] We concentrate on load balancing method in this article. Load balance is the distributed load mechanism of the different nodes which provides the best use of resources when tasks are overloaded.

W. Li et al[7] The optimisation algorithm for the particle swarm is then used for solving the server optimally depending on the topic mechanism. The enhanced algorithm will better improve the load balance on the power management cloud computing framework relative to the common Cloud Center Load Balancing Approach.

Swam Optimization

The PSO algorithm has been applied in order to solve optimization issues. The algorithm is based on a scanning agent for a number of particles that are searching for answers to a phenomenon in the quest area. Each particle in this swarm is focused on two parameters: self-learning ability and social experience across a series of iterations with other swarm participants. Every particle can find the best place relative to the previous position after each iteration of each algorithm, theoretically known as the best local position (local optima). Although the best place for each particle in the optimum global position is equal to the best position of all other particles in the swarm and select the correct one (global optima). The

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

cognitive parameter is taken into consideration for locale optima and the social networking parameter is used for global optima. The global optimum is the baseline for the next iteration, which the other swarm members would follow. Particle trapping in local Optima is one of the basic problems of the PSO algorithm; a state in which the particles travel at the same position with each iteration. The problem of local optim is defined. In addition, the global optimum problem is the situation in which the sum of particles in the swarm is repeated over time. In addition to the PSO algorithm, other optimization techniques, such as the Genetic and Ant Colony optimization algorithms, analyse local and international optimal problems.

Improving PSO technological performance by modelling various topology types is an active field of research. Due to the local optim dilemma, the PSO algorithm produces premature solutions to the complex high-dimensional query. He further argued that the current PSO lacks successful global search capabilities so that optimum solutions to deep and complicated optimization issues are not feasible. As a result, the optimization algorithm designers function in detail and easily converge on the fundamental improvement of the local optimum, or on discovering all the possibilities for more analysis in the quest field. The traditional strategy is to increase the diversity of the search space in order to boost the exploitation efficiency of the PSO algorithm.

III. PROPOSED METHODOLOGY

Cloud computing is an emerging region that present a lot of possible benefits to different organizations and frequent user. It is the extended appearance of distributed computing. . It is base on on-demand-service model in which in sequence, software, infrastructure and previous services are offer as per the client prerequisite at a number of instances of time. Load balancing is utilized to distribute workload amongst multiple cloud systems or nodes to acquire enhanced resource utilization. It is the important means to accomplish proficient resource contribution and exploitation. Load balancing has happen to a challenge issue currently in cloud computing system. To get together the users vast number of demands, there is a require of distributed resolution since almost it is not forever probable or cost proficient to handle one or additional inoperative services. To tackle problems by Swarm based algorithm use nature inspired Optimization technique. To develop an efficient load balancing algorithm with hybridisation of partial swam optimization technique and Genetic algorithm. It is not to function on the resolution pool except to create a convinced coding denotation.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

So primary we require doing the coding for the difficulty to be tackled. The selection of the coding system depends to an incredibly large degree on the issue and the aim to genetically optimise part of the swam and the genetic algorithm. The chromosome establishment by binary code has the classical genetic algorithm. The data representation evaluator in this paper may be ascertained as a one or more mapping relationship between physical machinery and VMs. This paper therefore prefers to mark the gene chromosome by the tree structure. In other words, any mapping solution is manifested as one tree, the device scheduling and monitoring node at primary level are the root nodes even as the instantaneous physical machines are provided with all N nodes and the third level M nodes are built on convinced physical machines for VM's. This paper specifically utilises tree spanning approaches focused on the partial swaming optimization approach and the genetic algorithm for initialization of the population.

The following concept refers to the tree: this tree is a tree consisting of elements in the physical engine set and VM set. This tree's root node is the predefined source node of oversight. In this tree are integrated both actual system nodes and VM nodes. Each node of the leaf is VM. The trend of the talling tree is that the defined requirements for balance of load must be coupled or reasonably fine descent within the heritage should be established. This implies that the tree itself should also be moderately fine. Therefore, during subsequent procedures, we may create a mapping connection between physical machines and VMs. We firstly measure the possibility of preference p (p is the proportion of the load total of any VMs in a single VM consignment) for each VM according to the load in the VM collection; following this, on the basis of the possibility p all the logic discs are needed to generate the minimum number of loaded nodes inside the physical engine set to create the initial spans tree's leaf nodule. This strategy improves the opening of the VMs with an extra heat creature and chosen VMs with low heat.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

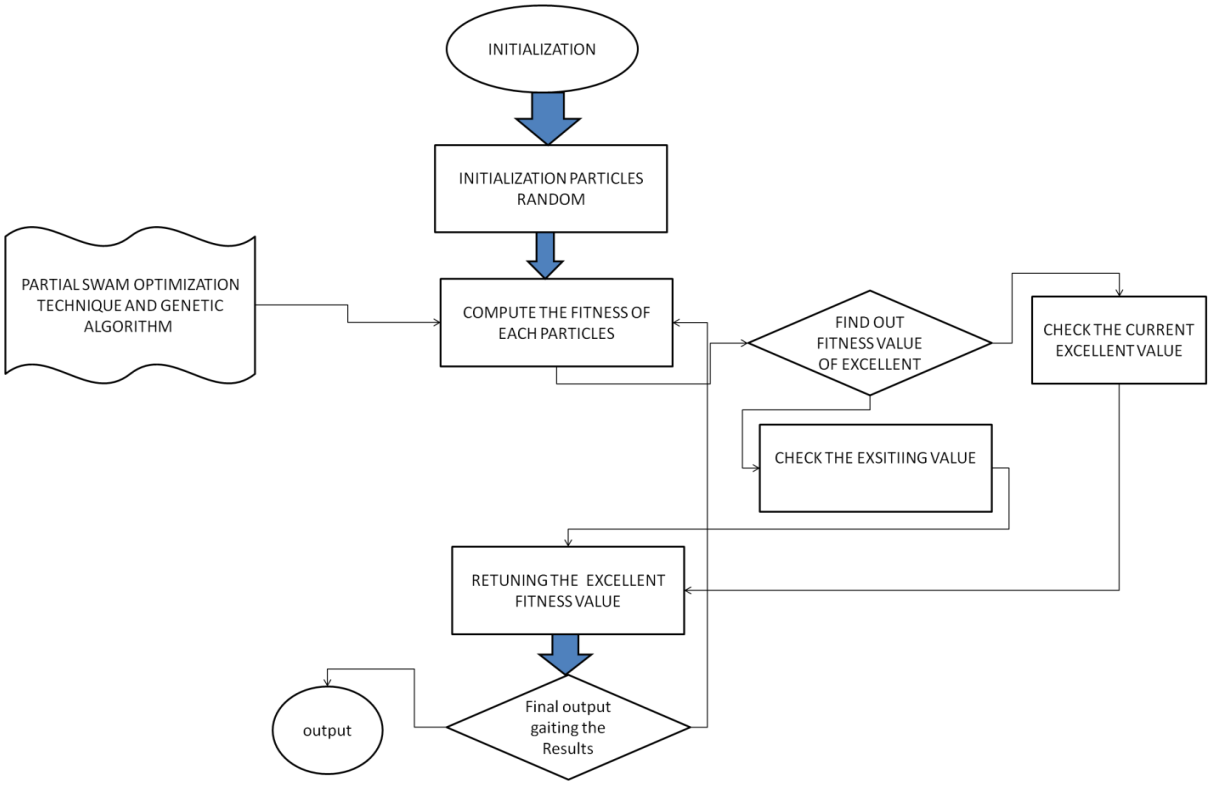

Figure 1: Flow chart for job allocation for load balancing in cloud environment based on virtual machine

PARTICLE SWARM ALGORITHM (PSO)

PSO is an optimization strategy focused on swarm. The solution space is used to identify the optimum path. During boot-up, it pushes all virtual machines around and selects the optimal computer for the production of the resource load. The ideal VM, where less load is accessible and task diagram, is defined by one of the mechanisms. This will reduce the relative resources and time consumed processing the node.

Basic Steps for PSO:

1. Initialize random-position and velocity particle population.

2. Calculate each particle's fitness feature value.

3. Compare the present value of the particle to the fitness value of each particle, and find Pbest.

Various algorithms are built to load balancing to various activities. If the literature review is done we should assume that most of the suggested load equilibrium algorithms are complicated and cannot be applied. The virtual machine would be better allocated to the PSO

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

algorithm role [11]. i.e. task scans all of the VM and assigns the right VM task to the lowest waste memory. The consumer sends his work request to the cloud service and determines that the VM can save the job.

VM based on PSO algorithm is chosen in the cloud server. Our task is to manage the load if VM overload happens. The first step is to submit and the request is approved by the Cloud service and the request is passed to VM. The user control begins the loop and provides the VM Scheduler with control. VM Scheduler key feature executes a PSO algorithm for load balancing. Only the overloaded VM is located on the basis of the threshold value. The next stage is to switch from the overloaded virtual machine to the loaded virtual machine after discovering an overloaded VM..

Improved PSO Algorithm:

1. Intialize pBest, gBest and p,S with 0s

2. Call intialize() [for particle generation]

3. Repeat while(false)

4. if pBest < gBest

5. gBest = pBest

6. do for i □ 0 to S

particle.get (i)

if testProblem (i) =Target

set condition true for while

7. pBest= minimum() [minimum in the particle]

8. Particle.get(gBest)

9. If Target_testProblem(pBest)< Target_testProblem(p)

p = pBest

10. a) getVelocity(gBest) [retrieve

velocity of gBest particle]

b) Updateparticle (gBest) [update

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

particle with effect to gBest]

c) S++ [make an increment in S's

current value]

11. else set condition true for while

12. end of function

IV. PROPOSED ALGORITHM

Algorithm: VM-Assign Load Balancer Input:

No of incoming jobs j], J2..... jn Available VM k], K2••• ..... kn

Output: All incoming jobs j], J2 ..... Jn are allocated smallest amount loaded virtual machine between the accessible Kh k2•••••••• .kn 1:

originally all the VM's have 0 allocation.

2: VM allocate load balancer preserve the index / allocate table of VMs which has no. of requests at present allocated to every VM.

3: When requests appear at the data canter it pass to the load balancer.

4: Index table is parsed and smallest amount loaded VM is particular for execution.

Case I: if establish a. ensure whether the selected least loaded VM is used instantly in the last iteration.

if YES goto step

4 to discover subsequently least VM if NO Least loaded VM is chosen

5: VM-assign load balancer proceeds the VM id to the data center.

6: Request is allocate to the VM. Data center alert the VM-assign load balancer about the allowance.

7: VM allocate load balancer modernize the requirements hold by each VM.

The heuristic searching algorithm aims to find the optimized results based on the "survival of the fittest individual" theory. The objective is to maintain the balance load over the virtual

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

machine and also to reduce the make span time. Following are the basic steps of Genetic Algorithm.

The representation of the individual in genetic form is termed as the genotype. This representation is shown in the

matrix form i.e., m X n matrix where m is considered as the number of tasks and n is considered as the number of

processors or the nodes for processing. Considering 4 tasks and two available resources, the chromosome will be like

Tasks	T_1	T_4	T_2	T_3
Resources	R_2	R_1	R_1	R_2

• Initial Population Genetic algorithm aims to search the large search space for finding the best solution (here nodes). So, instead of randomly generating the initial population, we tend to apply Dijkstra's algorithm [15] which reduces the search space along with time.

Selection

According to the fitness value, the individuals are categorized as less fit individuals and the most-fit individuals.

The most-fit individuals are selected for the creation of the offspring. The selection can be done by various strategies like

Roulette wheel method or the fitness proportion selection, stochastic universal sampling, tournament selection, reward

based selection etc. We have opted for the tournament selection strategy because of its property of working well in parallel

architecture. Moreover, it has efficient code.

• Crossover

Crossover is the operation in genetic algorithm which is used to produce child solution from the best fit parent individual that were selected. Two parents are selected and single point crossover is performed for generating child solution.

• Mutation

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

Mutation is another operation applied to the less fit individuals. The one or two genes in the chromosomes change their values i.e., from 0 to 1 or from 1 to 0 in order to make the individual fit for creating offspring that are fit for the next generation.

Termination

The algorithm will end when the termination condition occurs i.e., when the mutation rate tends to stagnate. This means that for certain generation the fitness value comes out to be same. As soon as this stage is reached the algorithm stops returning the best solution.

Results analysis : Implementation of our proposed Java programming strategy. The code is compiled by JDK version 6 following the coding of algorithm, and the class file is new to the CloudAnalyst tool to enforce. Operating system is Ubuntu 16,04, CPU is Intel® CoreTM 2 Duo 3,0GHz with a memory of 2,0GB and 320GB of disc space in Table 1 display the evaluation through dissimilar parameters, identical to sprite, throughput and time waiting, of the Load Balancing (LB) algorithms.

The difference between these algorithms is strong and bad, and this is described as high or low in duration. There are various results as discussed by historically different algorithms. This makes it fair to assign loads to static algorithms. It doesn't tolerate blame, because it's a smaller amount of multifaceted. There is no matching or fault-tolerant algorithm. In the case with limited operations, it shows the biggest result. Needs are specified in the current algorithm. It has enhanced its functions and offers high performance. Dynamic load balancing, along with this, only includes the current state of the system and has additional overhead and fault tolerance. Our proposed approach has strong productivity and low response time. It has low overhead and reliability since the VM device cannot allow high priority attempts.

Algorithm	load	time	cost
Traditional approach	8.4121	31.1427	40.1428
Proposed Approach	0.3261	14.7390	29.5175

Table 1: The performance comparison traditional approach and proposed

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

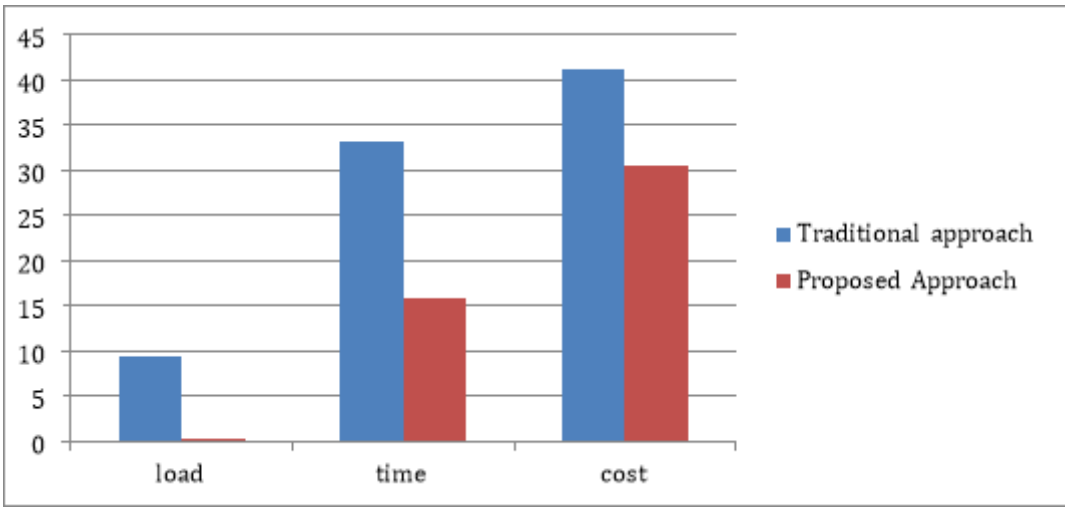

Figure 1: The performance comparison traditional approach and proposed

Compare varying algorithms with various parameters such as justice, concert, pace and difficulty in depth. OTBGA believes it to be reasonable to disperse the load, has strong performance, decent response time and less complicated than before. Our proposed algorithm is additional persuasive according to subsequent evidence. A time limit to utilising equivalent time for any job is OTBGA's primary benefit. Identifying the OTBGA is improved in terms of overall runtime and expense than current algorithms during the trial. OTBGA also has excellent performance in load balancing.

V. Conclusion and Future Scope

Task schedules play a critical function for cloud services, since both cloud providers and their customers need to be mutually beneficial. In this paper, we proposed a new PSO-inspired document approach to quantify the load between VMs. The assignments of roles are similar to the location vectors of the particle. Place vectors are first generated randomly, followed by a load balancing technique for each iteration. It will decrease the marksmanship by taking advantage of load-balancing technologies. It also gains from the right exercise feature by increasing resource usage and efficiency.

Our process certainly decreases make-up by 33 per cent and raises the device consumption by 22 per cent when converging in half-time to the global minimum in contrast with the PSO Job Scheduling method. Our approach is generic and flexible as, by growing the task-resource array dimension, it can be used in data centres with multiple tasks and resources. In addition, for all cloud systems with autonomous and non-preventional activities our approach is appropriate. We expect in the future to broaden our working flow method to take other QoS requirements into consideration such as fault tolerance and cost reduction.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3056 - 3068
Received 16 February 2021; Accepted 08 March 2021.

REFERENCES

[1]. Dave, B. Patel, G. Bhatt and Y. Vora, "Load balancing in cloud computing using particle swarm optimization on Xen Server," 2017 Nirma University International Conference on Engineering (NUiCONE), Ahmedabad, 2017, pp. 1-6, doi: 10.1109/NUICONE.2017.8325618.

[2]. S. Wang, L. Zhang, Y. Zhang and N. Cao, "Spark Load Balancing Strategy Optimization Based on Internet of Things," 2018 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Zhengzhou, China, 2018, pp. 76-763, doi: 10.1109/CyberC.2018.00025.

[3]. T. M. Shami, D. Grace and A. Burr, "Load Balancing and Control Using Particle Swarm Optimisation in 5G Heterogeneous Networks," 2018 European Conference on Networks and Communications (EuCNC), Ljubljana, Slovenia, 2018, pp. 1-9, doi: 10.1109/EuCNC.2018.8442519.

[4]. H. Mohamad, A. I. M. Isa, Z. M. Yasin, N. A. Salim and N. N. A. Mohd Rahim, "Optimal load shedding technique for an islanding distribution system by using Particle Swarm Optimization," 2017 3rd International Conference on Power Generation Systems and Renewable Energy Technologies (PGSRET), Johor Bahru, 2017, pp. 154-158, doi: 10.1109/PGSRET.2017.8251819.

[5]. Dandan Zhang, Wenbo Li, Zhe Jiang, Naihu Wu and Yanwen Hou, "Load shedding strategy coordinate optimization based on sensitivity analysis," 2016 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC), Xi'an, 2016, pp. 1893-1897, doi: 10.1109/APPEEC.2016.7779818.

[6]. Durgesh Patel, Anand S Rajawat, Efficient Throttled Load Balancing Algorithm in Cloud Environment,
International Journal of Modern Trends in Engineering and Research (IJMTER) Volume 02, Issue 03, [March - 2015] e-ISSN: 2349-9745, p-ISSN: 2393-8161

[7]. W. Li, T. Jian, Y. Wang and X. Ma, "Research on Virtual Machine Load Balancing Based on Improved Particle Swarm Optimization," 2019 IEEE Symposium Series on Computational Intelligence (SSCI), Xiamen, China, 2019, pp. 2846-2852, doi: 10.1109/SSCI44817.2019.9002730.

[8]. M. Agnihotri and S. Sharma, "Execution analysis of load balancing particle swarm optimization algorithm in cloud data center," 2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC), Waknaghat, 2016, pp. 668-672, doi: 10.1109/PDGC.2016.7913206.

[9]. A. S. Rajawat and A. R. Upadhyay, "Web Personalization Model Using Modified S3VM Algorithm For developing Recommendation Process," 2nd International Conference on Data, Engineering and Applications (IDEA), Bhopal, India, 2020, pp. 1-6, doi: 10.1109/IDEA49133.2020.9170701.

[10]. H. P. Kuribayashi et al., "Particle Swarm-Based Cell Range Expansion for Heterogeneous Mobile Networks," in IEEE Access, vol. 8, pp. 37021-37034, 2020, doi: 10.1109/ACCESS.2020.2975981.

[11]. M. Masood, M. M. Fouad, R. Kamal, I. Glesk and I. U. Khan, "An Improved Particle Swarm Algorithm for Multi-Objectives Based Optimization in MPLS/GMPLS Networks," in IEEE Access, vol. 7, pp. 137147-137162, 2019, doi: 10.1109/ACCESS.2019.2934946.

[12]. A. Singh Rajawat and S. Jain, "Fusion Deep Learning Based on Back Propagation Neural Network for Personalization," 2nd International Conference on Data, Engineering and Applications (IDEA), Bhopal, India, 2020, pp. 1-7, doi: 10.1109/IDEA49133.2020.9170693.