LOAD BALANCING IN CLOUD USING META-HEURISTIC BEES SWARM SCHEDULING

M. Ramesh
Department of Computer Science, Bharathiar University, India

Abstract

For virtualized file sharing, cloud storage uses scripting and load balancing in the cloud architecture. The two must be designed for optimum file sharing in the cloud computing environment. Latest advances in cloud data centre have been made in flexible traffic control for traffic balancing and service efficiency. But the delay during multidimensional resource distribution remains a problem. There is also a need for effective resource planning to guarantee cloud load optimization. In this post, we create an optimised algorithm for the preparation of resources and load balancing to provide efficient cloud services. The methodology provides a multidimensional resource planning paradigm for effective resource planning of cloud networks based on Bees Swarm Optimisation (BSO). A stable and balanced load-equilibrium makes the dynamically chosen request in a class with a multidimensional queuing load-optimization algorithm. A load balancing algorithm is then introduced to ensure resource understatement and overuse, which raises the latency period for each application form. Simulations have been undertaken for performance evaluation in the cloud data centre using Cloudsim simulators, and findings suggest that the approach suggested contributes to improved success rates, resource planning efficiency and response time.

Keywords:

Task Scheduling, Bees Swarm Optimisation, Resource Allocation, Scalable Traffic Management

1. INTRODUCTION

In the field of information management cloud computing plays a central role (IT). Cloud computing is given greater importance in the context of academic groups because of its distributed existence. The internet-based cloud computing model [1] aims to effectively distribute computing services (hardware and software). The modern file sharing architecture allows a successful load balance and resource planning to be established. Cloud-specified tools and directories are accessed by cloud users virtualizing. In this case, effective resource planning and load balancing remain the key priority for efficient file sharing.

Much analysis has been developed in order to schedule capital and balance cloud loads. For example, a cloud solution has been addressed to Scalable Traffic Management (STM) [2]. The overall link fee was minimised by means of STM and the load balance was thus assured between users in the network. However, this approach was not ideal for multidimensional resource planning. The aim of a scalable workload partitioning system [4] is to increase response time and performance for distributed transactions to deal with this problem. The cloud setting is also used for mission preparation with the actions of Honey Bee [3]. To optimum use of the computer, the action of sweet bees has been used. Honey bees' action is used to manage the load efficiently in a cloud world.

Cloud storage is widely respected and is a supreme means for handling and optimising data use and resources, as well as delivering various computing and IT services. In [6] explores new perspectives into data centre optimization, performance, and energy-reducing delivery methods. Resource utilisation remains unaddressed, however. To resolve this, the performance appraisal of resource management [5] to ensure quality of operation has been performed.

2. BACKGROUND

A heuristic task planning was studied in [8] to maximise the resource efficiency. Game theory processes are implemented in [7] unified model. However, the key requirements known as job preparation have not been followed. The new EDF, HLF and LSTF algorithms in [9] schedule different activities in one distributed system in real time. The key purpose of the algorithm was good quality (precise) performance and all workers joining the machine were assured.

Researchers have under varying conditions indicated resource planning for load balancing algorithms. Agent-based load balancing algorithm [10] uses an autonomous model to provide a complex load balance for the cloud world. Optimisation, although load balancing was ensured, has not been accomplished. The budget-based programming algorithms [11] are optimised for maximising performance through the global greedy budget and progressive refinement method to solve the problem.

A movement towards globally dispersed data centres was rising with the formation of cloud computing. Thus, energy-efficient cloud servers and costs are very important to reduce the cost of power. An Online Scheduling Algorithm was created [14] to increase energy costs and equity between various organisations. Qi Zhang et al. [12] has modified dynamics to minimise energy consumption and latency in a few devices by presenting the K-means cluster algorithm. A Hyper Heuristic Scheduling algorithm was developed to increase planning efficiency [13].

Load balance can be one way to maximise capacity. Load balance. The optimum power allocation and load distribution is discussed for multiple cloud servers using a queuing system [15]. Dynamic multi-server load balancing was developed to consign the data centre using an efficient load balancing algorithm. An object-oriented approach built on a modern, rather decentralised reporting framework [16] that enhances efficiency of scheduling for effective data exchanged. The heterogeneous allocation of capital using selfish, cost-effective heuristic approaches [14].

The BSO approach at the necessary cloud level did not achieve the load balancing. The way out needs longer to allocate money. Performance preparation, reaction time and charge balance were the unaddressed issues. To boost cloud efficiency for solving problems in current methods, the BSO planning approach is proposed.

In this paper, we create an optimised algorithm for the preparation of resources and load balancing to provide efficient cloud services. The methodology creates a BSO (BSO)-based multidimensional resource planning paradigm to achieve reliability in cloud computing capital planning. A powerful and balanced load balancing can lead to increased usage of virtual machines with a dynamically chosen class application with a Queuing Load Optimization algorithm. A load balancing algorithm is then introduced to ensure resource understatement and overuse, which raises the latency period for each application form.

3. PROPOSED METHOD

The cloud architecture focuses on optimum resource usage and minimal time for transmission. In cloud computing, BSO approaches are used to solve certain issues. In the suggested approach, the BSO programmer reduces the response time of job applications. Multidimensional BSO-based resource planning increases the performance of resource planning. The goal of a load balance is to improve the effectiveness of resource planning and the overall performance rate for jobs. The resources preparation based on BSO is structured to optimise the usage of resources and reduce working time.

The BSO approach consists of three steps in the cloud world between users and servers; (1) incoming requests are submitted to users; and (2) resource preparation using Online Resource Management.

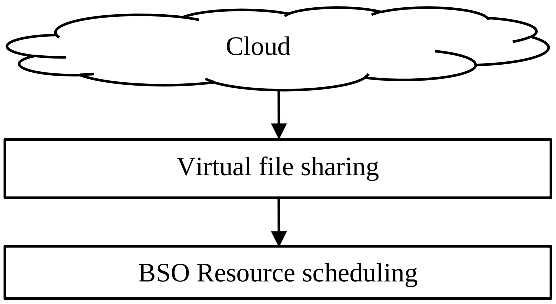

Fig.1. Process Flow of the proposed BSO model

3.1 RESOURCE SCHEDULING USING BSO

The bee algorithm is a bio-inspired metaheuristic method for honeybees and how they look for plants to get the pollen required for the production of honey. In a large territory a colony of bees is searching for new food sources and starts the production of new sources of food. Where there is much pollution in these sources, the amount of bees gathered from these sources reduces and significant numbers of bees decreases. The colony sent too many bees during the pollen collection season, known as scout bees, to accept the territory at random, to remind the colony of the best opportunities for food.

The colony retains a certain proportion of its scout bees until the harvest season begins to detect stronger and newer sources of food. When scout bees come back to the colony and have discovered a better opportunity than the colony at the moment, Dance transmits the exact location of the food source and the colony then begins sending more bees to the resource.

The algorithm for resource scheduling is given below:

Step 1: Require: population size, neighborhood

Step 2: Initialize population with random solutions.

Step 3: Evaluate fitness of the population.

Step 4: While stopping criterion not met do

Step 5: Select the sites for neighborhood search

Step 6: Recruit bees for n selected sites and evaluate their fitness

Step 7: Select the fittest bee from each site assign remaining bees to search randomly

Step 8: Evaluate their fitness end while

The first step that the BSO does is change the velocity vector location of the particles and pick a defined number of particles, in this instance it is recommended that the new food source found and searched the region surrounding the radius searches should be chosen best than the same particles around them.

To do this correctly you need a way to Fig.out what the next and last particle location is, if we cannot find this then this metaheuristic cannot be implemented so it wouldn't be understood what the next and last particle will look like in this field.

The next elements such as binary operations have been specified and one number has been applied to the solution vector subtracted. If we have a 5-element knapsack problem, for example, the solution vector would have five spaces and can be 0 or 1, in each space, we will find the following and the next vector example above.

4. RESULTS

Experiments are conducted in three metrics, including average success rate, resource planning efficiency and response time in order to show the effective performance of BSO on tasks and resource allocations.

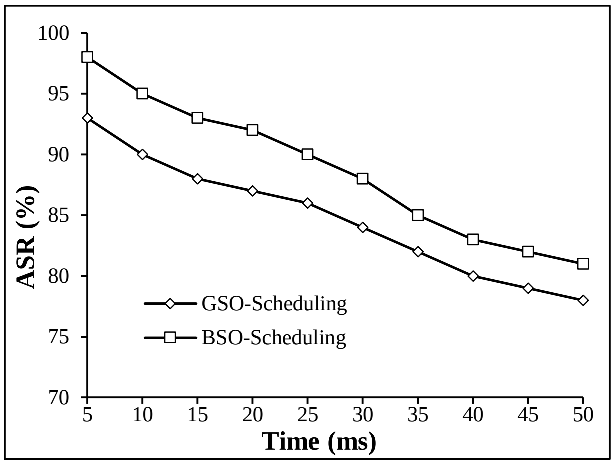

Fig.2. Average Success Rate (%)

The Fig.2 illustrates the percentage average success rate (ASR) between the BSO proposed and the current STM model. The result shows that for all user requests, the ASR of the proposed BSO is higher than the current STM model.

The Fig.3 indicates the percentage of resource planning effectiveness (RSE) between the BSO proposed and the STM model currently in use. The result demonstrates that the RSE of the BSO proposed is lower than the current model for STM.

The response time (RT) between the proposed BSO and the existing STM model is shown in milliseconds in Fig.4. The findings reveal that the RT of the proposed BSO is less than the current STM model for all consumer requests.

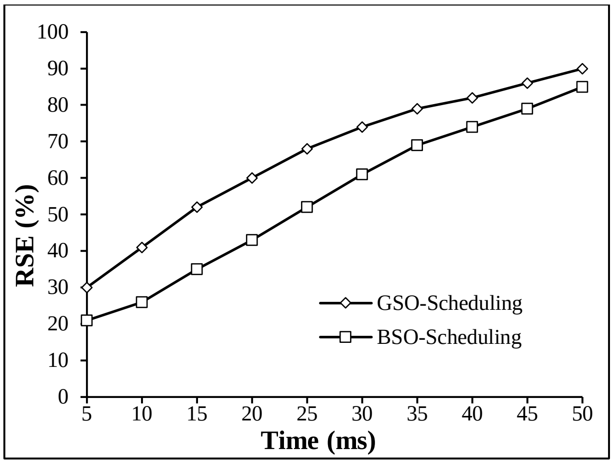

Fig.3. Resource scheduling efficiency (%)

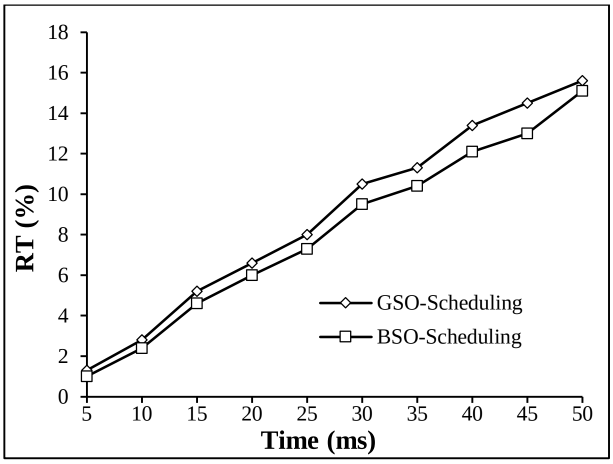

Fig.4. Response Time (ms)

5. CONCLUSIONS

This paper offers the BSO method for efficient resource planning and optimization of the load balance for every user's cloud application. The main objective of this proposed method is to use an efficiently integrated algorithm for scheduling and balance loads with the maximum use of cloud resources and minimal processing time. The performance of the BSO enhances the efficiency of resource planning on demands of the user. The efficient resource is then set up for similar services based on resources, with a minimal response time. The multi-dimensional queuing network maintains an optimal load balancing with the expected services. This boosts the overall performance rate for each operation of the cloud customer. By using models to assess the average performance and reaction time for resource planning, the effectiveness of the BSO approach is calculated. The simulation is performed for different criteria like average success rates, resource programming efficiency and time to respond. Simulation research reveals that resource productivity is improved by 8% compared with the modern interventions, which improve receptive time by 36%.

REFERENCES

[1] W. Li and L. Ping, "Trust Model to Enhance Security and Interoperability of Cloud Environment", *Proceedings of International Conference on Cloud Computing*, pp. 69-79, 2009.

[2] X. Meng, G. Zhang, J. Kang, H. Li and D. Li, "A New Subjective Trust Model Based on Cloud Model", *Proceedings of IEEE International Conference on Networking, Sensing and Control*, pp. 1125-1130, 2008.

[3] K. Dasgupta, B. Mandal, P. Dutta and J.K. Mandal, "A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing", *Procedia Technology*, Vol. 10, pp. 340-347, 2013.

[4] S. Ahirrao and R. Ingle, "Scalable Transactions in Cloud Data Stores", *Journal of Cloud Computing*, Vol. 4, No. 1, pp. 21-34, 2015.

[5] Shahin Vakilinia, Behdad Heidarpour and Mohamed Cheriet, "Energy Efficient Resource Allocation in Cloud Computing Environments", *IEEE Access*, Vol. 4, pp. 8544-8557, 2016.

[6] A. Shimpy and Jagandeep Sidhu, "Different Scheduling Algorithms in Different Cloud Environment", *International Journal of Advanced Research in Computer and Communication Engineering*, Vol. 3, No. 9, pp. 1-16, 2014.

[7] P. S. Pawar, M. Rajarajan, S. Krishnan Nair and A. Zisman. "Trust Model for Optimized Cloud Services", *IFIP Advances in Information and Communication Technology*, Vol. 374, pp. 97-112, 2012.

[8] K. Rzadca, A. Datta, G. Kreitz and S. Buchegger, "Game-Theoretic Mechanisms to Increase Data Availability in Decentralized Storage Systems", *ACM Transactions on Autonomous and Adaptive Systems*, Vol. 10, No. 3, pp. 14-22, 2015.

[9] H. Chen and B. Li, "Qualitative Rules Mining and Reasoning based on Cloud Model", *Proceedings of IEEE 2nd International Conference on Software Engineering and Data Mining*, pp. 523-526, 2010.

[10] G.L. Stavrinides and H.D. Karatza, "Scheduling Multiple Task Graphs with End-to-End Deadlines in Distributed Real-Time Systems Utilizing Imprecise Computations", *Journal of Systems and Software*, Vol. 83, No. 6, pp. 1004-1014, 2010.

[11] Q. Guo, D. Sun, G. Chang, L. Sun and X. Wang. "Modeling and Evaluation of Trust in Cloud Computing Environments", *Proceedings of 3rd International Conference on Advanced Computer Control*, pp. 112-116, 2011.

[12] Dzmitry Kliazovich, Johnatan E. Pecero, Andrei Tchernykh, Pascal Bouvry, Samee U. Khan and Albert Y. Zomaya, "CA-DAG:Modeling Communication-Aware Applications for Scheduling in Cloud Computing", *Journal of Grid Computing*, Vol. 12, No. 3, pp. 1-17, 2015.

[13] T. Dillon, C. Wu and E. Chang, "Cloud Computing: Issues and Challenges", *Proceedings of 24th IEEE International Conference on Advanced Information Networking and Applications*, pp. 27-33, 2010.

[14] Y. Wang and W. Shi, "Budget-Driven Scheduling Algorithms for Batches of MapReduce Jobs in

Heterogeneous Clouds”, *IEEE Transactions on Cloud Computing*, Vol. 2, No. 3, pp. 306-319, 2014.

[15] Q. Zhang, M.F. Zhani, R. Boutaba and J.L. Hellerstein, “Dynamic Heterogeneity-Aware Resource Provisioning in the Cloud”, *IEEE Transactions on Cloud Computing*, Vol. 2, No. 1, pp. 14-28, 2014.

[16] C.W. Tsai, W.C. Huang, M.H. Chiang and M.C. Chiang, “A Hyper-Heuristic Scheduling Algorithm for Cloud”, *IEEE Transactions on Cloud Computing*, Vol. 2, No. 2, pp. 236-250, 2014.