ENERGY-EFFICIENT SCHEDULING IN CLOUD COMPUTING ENVIRONMENT USING META-HEURISTIC OPTIMISATION

B. Anand

Department of Computer Science, SNS College of Arts and Science, India

Abstract

The cloud computing utilization has increased markedly, leading to traffic obstruction on web servers. The high number of customers has also created a disproportionate distribution and allocation of resources in the demand for resources. The proposed work addresses the cost-effective approach to load balancing and resource planning. Implementing various algorithms to effectively allocate resources and balance the workload in the cloud also causes a high energy consumption. To make more efficient use of algorithms, we use the cloud Analyst simulator to monitor their results. Finally, our work is based on the algorithm for Cuckoo Search Algorithm (CSA) which decreases energy consumption and time of performance as compared to other algorithms.

Keywords:

Energy Efficiency, Cloud Scheduling, Cuckoo Search Optimization

1. INTRODUCTION

Cloud computing is a merger of two main phenomena, IT reliability and corporate agility. The cloud has unlimited data storage, which hides huge volumes of data from other users. The necessary files, documentation and apps can be accessed on request by users. Instead of acquiring a costly infrastructure, users pay for the services offered by the cloud. There are different features of cloud computing, such as on-demand distribution of resources, service efficiency, elasticity, etc., making both the university and the business sector particularly active. The continuing demand for cloud computing has led to a need to control machinery loads, electricity generation and resource scheduling. Load balancing is the way multiple functions are allocated across various resources in the data centre. A datacenter, a virtual computer or a physical machine may also have the available tools [2] [3].

The delivery of resources and services must be carried out systematically to make any resource the same load at every time and increase the total resource use rate [4]. When load imbalances of some type occur, the device output decreases dramatically. The energy usage should be taken into account when managing the load. Green cloud computing is an expression which efficiently encompasses the use of resources and reduces the energy consumption of those resources [5]. Both cooling and computer services are responsible for the energy utilization resources in the datacenter. While the computing facilities account for about 60% of the overall energy consumption, the cooling infrastructure accounts for another 30% [6].

There are two sections for the issue of energy consumption: one of operations around the server and, (b) network side. An important idea is to optimize resource allocation and reduce overhead costs. This can be seen as a business section throughout the platform. Schedulers are used to plan the resources; the load balancers are used for resource balancing and load prediction, as well as energy reduction. The services or equipment are either reserved or de-allocated in the case of cloud computing. The greatest advantage of Cloud integration is the abolition of spending pressure and the lowering of operating and maintenance costs. The allocation of resources is the role of resource allocation while ensuring an adequate ecosystem balance. Resource planning algorithms are implemented to maintain a good equilibrium to achieve optimal efficiency.

Nowadays, fog computing and edge computing are also used, since IOT computers are using cloud computing for data storage. Fug computing aims to mitigate and control traffic in the network load of data centres [8] [9]. There has been a lot of work performed on cloud services using different algorithms of optimization, but the increased everyday use of the cloud system also demands development. There are more and more people using growing numbers of servers, but this increased use often increases the load, the energy consumption and the cloud system resources required.

The core reason behind this research was therefore to enhance the efficiency of cloud systems from a new metaheuristic perspective called cuckoo search algorithms: load balance, resource planning, energy use, etc. We conducted an experimental survey on different algorithms which are useful for load balance, resource planning and energy conservation before introducing the WEA algorithm. We used the cuckoo search algorithm (CSA) to compute the values in order to balance the load over the clod scheme. We then tested CSA algorithms, which were fine for the assignment of tasks according to the literature, and got their results for the planning of resources. Finally, we suggest and apply a whale-optimization algorithm that gives the best result for a cloud system for task efficiency, response time and energy use.

2. RELATED WORKS

Here we include a brief analysis of newly designed frameworks focused on the planning and optimisation of resources. We also study the applications that are already in place using different approaches to balance the load in the cloud.

The following section summarizes and discusses various newly evolved load balancing systems. In order to accomplish load balancing, in [7] have proposed a scheme for public clouds with a ravenous rooster optimisation strategy (RROP). The RROP architecture comprises the Mission Subset Formulator (TF) and the Physical Machine Formulator, two functional modules (PMF).

In [1] introduced PMK-ELM based on the NSGA-II integrated algorithm for the KNN and the NSGA-II variant of the TS-non-dominated sorting genetic algorithm. The PMK-ELM enables mission execution estimation while TS-NSGA-II makes it easier to pick the right reducers.

In [2] suggested a load balancing solution planned for the cloud world in several steps. The method has a JIJS that combines: a) the shortest queue approach and b) the idle queue approach. It includes 3 main steps:: (a) select from the table a scheduler with the maximum length of the queue, assignment of the task is made and updates are carried out; (b) select a VM with the lowest queue length to assign the task; (c) complete the task; update a VM table with a VM scheduler, and finally update the appropriate scheduler

In [10] proposed a new method for moving the cargo from a simulated overloading unit to a loaded VM. The strategy helps to prevent the fast allocation of loads on the under loaded devices. This leads to an optimal use of resources. The service level agreement ensures adequate follow-up.

In [11] suggested using an autonomous agent-based self-governing system to balance the workloads of virtual machines. A freelance migration agent performs the load balancing role in an effective way. The key elements for the SGA LB algorithm include: (a) in-house agent and (b) external agency. When a computer is overrun, a second appropriate VM is found on the in-house agent.

The modular solution for workload balance was proposed in [12]. It uses a multi-time approach. A cloud watch, a crone work algorithm and a round-robin algorithm were used to provide a wide range of techniques. First, all requests start working when the watch is set to 0. Furthermore, crone workers are used for treatment, along with a round robin algorithm to prevent famine and shorten the overall operating period.

3. RESOURCE SCHEDULING

A careful planning of resources is a must to make the job effective. For this, resource planning algorithms must be applied to make sure that certain VMs that are free and that VMs are still free after completing the tasks are correctly allotted as seen in Fig.1.

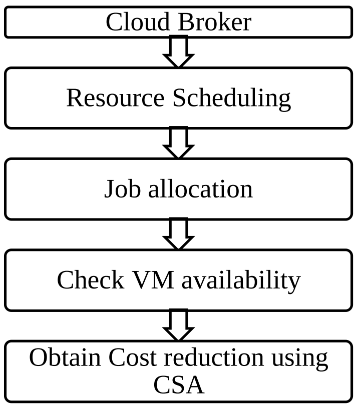

Fig.1. Workflow of Resource Scheduling

The integrated energy efficient resource management model for cloud architectures uses such load-balancing algorithms that effectively minimize energy consumption when balancing loads between server models [4]. The system's core objective is to balance loads by transferring loads from overloaded servers to below loaded servers and to efficiently schedule tasks which help to reduce energy consumption in a cloud environment. While all of these programs are optimized for this reason, energy consumption can be greatly reduced by using an algorithm hybrid. The framework is designed to construct and imitate cloud infrastructures using the Cloud Analyst app. This thesis follows a certain method to balance workload and the distribution of resources to reduce energy consumption.

First, the Json file gives parameters for various tasks and collects their input in the form of a task. Then there is the scheduler that assigns the task to separate virtual machines depending on the load that this specific machine has been assigned. After this, a proposed algorithm monitors the performances and redistributes the resources to boost the device efficiency of various devices depending on their roles.

By following such algorithm moves, the proposed optimized model of energy-efficient load balance works. The following are the methodological steps:

Step 1: Enter the number of parameters needed to construct the number of tasks.

Step 2: Host the server numbers to assign separate services for each virtual machine. Step 2:

Step 3: Use a data center to start and reserve the capacity for all virtual machines on it.

Step 4: Use the optimization methods to plan the resources better and match loads from one server to another by resource migration and energy usage calculation.

Step 5: We must calculate the time for migration and migration, energy usage and load equilibrium performance on the cloud before and after migration.

Step 6: We should start the event one by one, to create the GUI for each optimization method with a proper execution button.

Step 7: We will find the best approach for analysis by applying all the approaches suggested. viii. Work through the Java and Cloud Analyst tool kit to complete all planned work.

Many models, such as CPU models, systems models and parallel device models, are possible to measure the energy used by servers. Cloud platforms use servers with multiple hardware components, but the CPU, which uses most resources, is an integral part of the physical computer. Software can be built on the basis of less CPU use and use may be controlled by other programs as well. On that basis, the energy efficiency of a cloud infrastructure can be improved. The algorithm suggested efficiently arranged the job to manage server loads. When the load is collected by the server for the execution of operations, the server is at full power usage, while the server consumes minimum power while there is no load on the server. We believe that P_{min} for an optimal server's minimum power consumption is 200 Watts, and that the highest power consumption is 400 Watts, P_{max}. Otherwise, 70%-80% of the CPU usage is best maintained and machine stability would be compromised.

4. CUCKOO SEARCH ALGORITHM

This is an optimization algorithm for balancing the load of certain cuckoo species that lay their eggs in the nests of other host birds of other species. Inspired by the necessary or forced parasite performance, the next generation is the crib, which is better with high-quality shells. There is a constant or set number of host nests. He needs to chuck eggs out of the host nest to produce a new nest solution.

CSA Algorithm

Begin

Objective function *f*(*x*)

Generate initial population of n host nest

Evaluate fitness and rank eggs

While (*t* > MaxGeneration)

$T = t + 1$

Get a cuckoo randomly

Generate new solution by Levy flights

Evaluate quality/fitness, F_i

Choose a random nest *j*

If ($F_i > F_j$)

Replace j by the new solution

End if

Worst nest is abandoned with probability *Pa*

New nest is built

Evaluate fitness

Rank the solution and fit current best

End while

Post process results and visualization

End

There are essentially two notions in a cloud system: one is the assignment of resources and the next the assignment of tasks. Assignment of services means assignment of virtual machines acting as cloud system resources. Virtual machines are assigned to tasks depending on such assumptions, such as which machine is optimal, which carries out the same task and more. The second task is the assignment of tasks, where tasks come from users and it is to be carried out on our virtual machines [4]. Each job includes parameters such as the ID, time of arrival, use of the CPU, etc. The allocation of resources is only achieved where matching resources for this specific purpose are available. A linear programming equation can be used as described in equation (3) for minimizing the energy resource allocation of the cloud system:

$$\min E = \sum_{T=1} \sum_{i=1}^{m} E_i(T) \tag{1}$$

$$E_i(T) = (P_{\max} - P_{\min}) \frac{U_i(T)}{100} + P_{\min} \tag{2}$$

$$U_i(T) = \sum_{j=1}^{n} u(i,j) \leq T \tag{3}$$

5. PERFORMANCE EVALUATION

We considered 8 servers with various configurations for the proposed algorithm, such that some servers have 2 GB of RAM, some 3 GB of RAM. In the same vein, some servers are 32-bit, most are 64-bit. This is used when loading the computers onto the system. The XML file also contains P_{min} and P_{max} values for each server for the proposed WOA algorithm, since our algorithm not only schedules tasks, but also calculates the energy usage by this same computer during the assignment of tasks.

Cloud Analyst is a toolkit for simulation that includes both modeling and simulation of the cloud functionalities, such as event handling, building cloud entities, collaboration with different entities, etc. This toolkit offers several resources and functions: In a recurring and managed setting, test applications resources. Experiments on imitated development infrastructure with various workload combinations and various efficiency scenarios. Set device restrictions before applications are deployed in the cloud. It is a comprehensive platform for the modeling, delivery and allocation approach of cloud service agents. It is similar to a cloud system in that it can connect infrastructure from both the public and private realms. The availability of a virtualization engine that supports the development and maintenance of various independent virtual resources on a DC node.

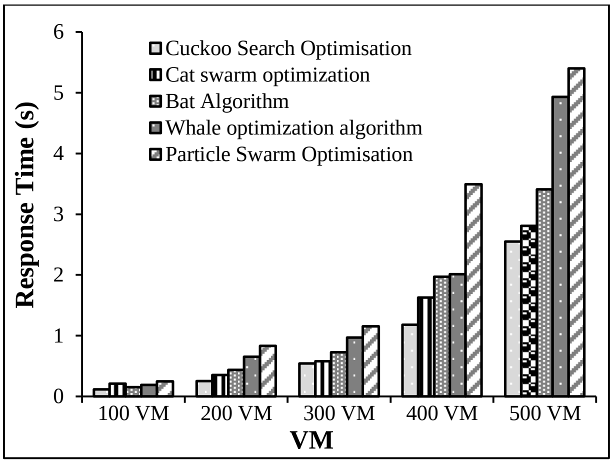

Fig.2. Response Time

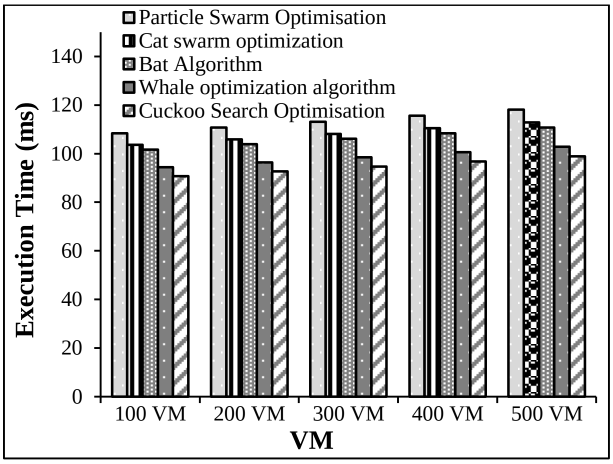

Fig.3. Execution Time

The Cloud Analyst toolkit contains the datacenter tab that helps to set the datacenter's configuration. Values for different parameter fields such as name, area, operating system, server number, etc. must be provided. A second table occurs in the data center which requires additional parameter values such as machine id, memory, storage, processor speed, VM assignment policy. The Data Center comprises many physical servers which all have different resource characteristics, such as storage, CPU, speed, storage, architecture, etc.

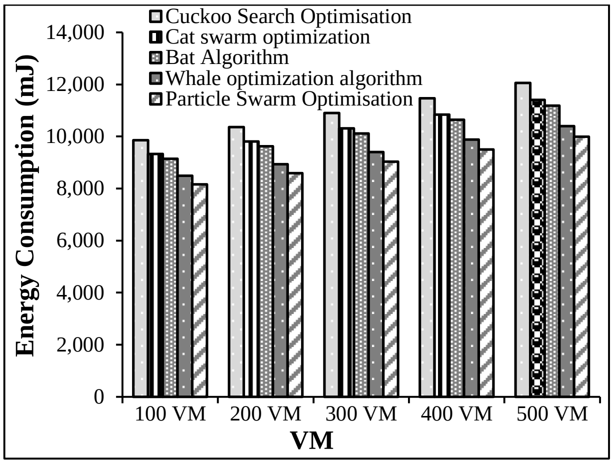

Fig.4. Energy Consumption

6. CONCLUSION

The use of cloud computing has greatly improved, resulting in traffic obstruction on web servers. The vast number of customers has also raised demand for services and caused an imbalance in resource sharing and allocation. The proposed thesis explores a cost-effective approach to load handling and resource planning. The introduction of various algorithms to efficiently allocate resources and balance workloads in the cloud has often been seen to induce high energy consumption. First of all, in order to make use of increased efficiency, we have implemented PSO, CSO, CSA, and BAT algorithms. Finally, our proposed study uses an algorithm of whale optimization to reduce energy consumption and time as compared with other algorithms.

REFERENCES

[1] Nikky Suryawanshi Rai, Susheel Jain and Anurag Jain, “Mining Interesting Positive and Negative Association Rule Based on Improved Genetic Algorithm”, *International Journal of Advanced Computer Science and Applications*, Vol. 5, No. 1, pp. 160-165, 2014.

[2] Karaboga Dervis and Bahriye Basturk, “A Powerful and Efficient Algorithm for Numerical Function Optimization: artificial Bee Colony (ABC) Algorithm”, *Journal of Global Optimization*, Vol. 39, No. 3, pp: 459-471, 2007.

[3] I. Berin Jeba Jingle and J. Jeya A. Celin, “Mining Optimized Positive and Negative Association Rule using Advance ABC Algorithm”, *Journal of Theoretical and Applied Information Technology*, Vol. 95, No. 24, pp. 6846-6855, 2017.

[4] I. Berin Jeba Jingle and J. Jeya A. Celin, “Markov Model in Discovering Knowledge in Text Mining”, *Journal of Theoretical and Applied Information Technology*, Vol. 70, No. 3, pp. 459-463, 2014.

[5] A. Kennedy and R. Eberhart, “Particle Swarm Optimization”, *Proceedings of IEEE International Conference on Neural Networks*, Vol. 4, pp. 1942-1948, 1995.

[6] L.A. Nivedita and K. Sravani, “Effective Service Security Schemes in Cloud Computing”, *International Journal of Computational Engineering Research*, Vol. 3, No. 2, pp. 2250-3005, 2012.

[7] K. Goyal and P. Supriya, “Security Concerns in the World of Cloud Computing”, *International Journal of Advanced Research in Computer Science*, Vol. 4, No. 4, pp. 976-997, 2013.

[8] Bing Wang, Yao Zheng, Wenjing Lou and Y. Thomas Hou, “DDoS Attack Protection in the Era of Cloud Computing and Software-Defined Networking”, *Computer Networks*, Vol. 81, No. 3, pp. 308-319, 2015.

[9] Joseph Idziorek and Mark Tannian, “Exploiting Cloud Utility Models for Profit and Ruin”, *Proceedings of IEEE International Conference on Cloud Computing*, pp. 33-40, 2011.

[10] P. Priyadharsini, P. Dhamodran and M.S. Kavitha, “A Survey on De-Duplication in Cloud Computing”, *International Journal of Computer Science and Mobile Computing*, Vol. 3, No. 11, pp. 149-155, 2014.

[11] R. Shobana, K.S. Shalini, S. Leelavathy and V. Sridevi, “DeDuplication of Data in Cloud”, *International Journal of Chemical Sciences*, Vol. 14, No. 4, pp. 2933-2938, 2016.

[12] Jun-Ho Lee, Min-Woo Park, Jung-Ho Eom and Tai-Myoung Chung, “Multi-Level Intrusion Detection System and Log Management in Cloud Computing”, *Proceedings of International Conference Advanced Communication Technology*, pp. 552-555, 2011.