ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

Energy Aware Load Balancing Scheduler for Green Cloud.

A Paulin Florence[1]*, and V Shanthi[2].

[1]Research Scholar, Sathyabama University, Associate Professor, St. Joseph's Institute of Technology, Chennai, Tamil Nadu, India.
[2]Professor, St. Joseph's College of Engineering, Chennai, Tamil Nadu, India.

ABSTRACT

Cloud computing has immensely improved the computational prowess of corporates by offering dynamic allocation of resource being used and exponentially increasing the amount of data being processed. It has many benefits including reliability, security and ease of maintenance. However cloud is still an evolving technology and needs focus majorly on the areas of Resource Management and Energy management. This paper relates to the previous work done on load balancing model using firefly algorithm(FA) in cloud computing and energy conservation for dynamic voltage frequency scaling(DVFS) for computational cloud. Here we have applied the benefits of both the papers and the output obtained is also compared with Adaptive genetic algorithm model. Thus enabled us to achieve optimal management of cloud resources and effectively negates any adverse environmental impacts to a large extent.
Keywords: Cloud Computing, load balancing, Firefly Algorithm, Green Cloud, Dynamic Voltage frequency.

Corresponding author

INTRODUCTION

Cloud computing is used wherever there is extensive usage of data either in the form of e-com activities or Web based queries or grid computational method. It enhances storing, analysing and processing of data to large scale. Cloud computing is nothing but distribution of interconnected resources at various locations communicating with each other. Cloud can be deployed as a Private, Public or Hybrid model depending on customer needs. It can be availed either in the form of Software as a service (SaaS), Platform as a service (PaaS) or Infrastructure as a service (IaaS) [1]. Private cloud is when individual companies own and operate data centres. Public cloud is where internet is used for providing services and hybrid is a mix of private and public cloud depending on customer's requirement and needs [2].

Resources available in cloud are managed by load balancing or scheduling. It's the process of effectively managing the available infrastructure by dynamically allocating total load over available nodes. This establishes improved performance, addresses redundancy & stability between systems. Some of the commonly used algorithms for scheduling are Genetic algorithm, FCFS, Min-Min and Max-Min [5].

There has been an exponential growth of unstructured data resulting in rapid increase in Cloud computing data centres. This has benefitted the way data handling is done however has left an unpleasant impact on the overall environment. Reduced energy consumption in data centres has been a hotly discussed topic due to the impact it has on environment because of the alarming increase in usage of power and eventual impact on carbon footprint.

Today's need is to pave way for green world, to achieve this many energy management techniques are used in cloud computing environment. The primary objective of this paper is to achieve load balancing together with energy conservation. In order to achieve this, scheduling list is prepared using Firefly algorithm [3], best fit strategy is applied over the scheduled list. Now incoming jobs are classified based on their time complexity using pattern analyser algorithm [4]. By evaluating the size of the job, it is multiplied by the time complexity, thus MIPS is computed.

Now through LB scheduled list, using best fit strategy, the appropriate host is identified and the incoming job is allocated to the victimized host. Then MICC decides the frequency at which the host is to be operated. According to that processor frequency is scaled up or down to enable energy saving.

This paper is organized as follows: related work is discussed in Section 3, followed by proposed methodology in Section 4, and results in section 5, finally we conclude the paper with summary.

RELATED WORK

Rajwinder kaur et al. [5] proposed Min-Min Heuristic Algorithm which is a traditional algorithm based on the concept of minimum completion time and works on the principles of two phases i. Calculation of expected completion time for any incoming requests ii. Request with the overall minimum completion time is given priority over other requests. However this results in requests with longer waiting time left in queue and this subject to poor load balancing in cloud.

Kobra Etminani et al. [6] proposed a hybrid algorithm called Resource aware scheduling algorithm (RASA) which incorporates both min-min and max-min scheduling methods. Rasa also calculates the overall completion time of each request and apply either min-min strategy or max-min strategy. However it inherits the same pitfalls of both of the strategies resulting in imbalance of loads.

Upendra Bhoi et al. [7] suggested Max-Min Heuristic algorithm works on similar principles of Min-Min algorithm, however requests with maximum completion time are scheduled at the top of the queue. This resulting in starvation of jobs which involves minimum completion time, in case, number of larger tasks increases.

Dharmesh Kashyap et al. [8] analysed various load balancing algorithms used in cloud computing and stated their advantages and disadvantages. He also compared their performances.

Jasmeen Kaur et al. [9] used Adaptive firefly algorithm for job scheduling cloud and compared the results with ACO and he has considered only on high intensity jobs.

S.R.Suraj et al. [10] using Adaptive Genetic algorithm managed overloading in cloud by migrating excessive load in PM to virtual machine.

PROPOSED METHODOLOGY

The overall frame work for Energy Aware Load Balancing scheduler for Computational cloud is shown in figure 1.

For the purpose of research we have comparatively analysed varying loads using adaptive genetic algorithm against on Load Balancing Model using Firefly algorithm(FA) in cloud computing. Adaptive genetic algorithm is in itself a dynamic searching method using randomness to allocate incoming jobs to a virtual machine.

In Adaptive Genetic Algorithm, multiple phases are constructed which involves virtual calculation of node weights, distribution of load amongst the nodes, and job execution.

In terms of weight calculation, available CPU, CPU utilisation rate and Memory usage rate are used. Higher weightage is given for CPU utilisation compared against other variables used while identifying node weights.

Once the individual node weights are identified, it's then compared against the total weight of available nodes. The node with the highest individual weight is given priority over other nodes in job allocation.

In the earlier work published under "A Load Balancing Model Using Firefly Algorithm in Cloud Computing" we have enumerated the benefits of Firefly Algorithm(FA) [3] which allows for optimal load balancing in cloud. It works on the concept of victimising a node based on its attraction value and its affinity towards the job. It's been proven that the mean processing time and CPU utilisation rate has shown a considerable decrease when FA is used. Given below a brief view of the methods followed under FA for scheduling.

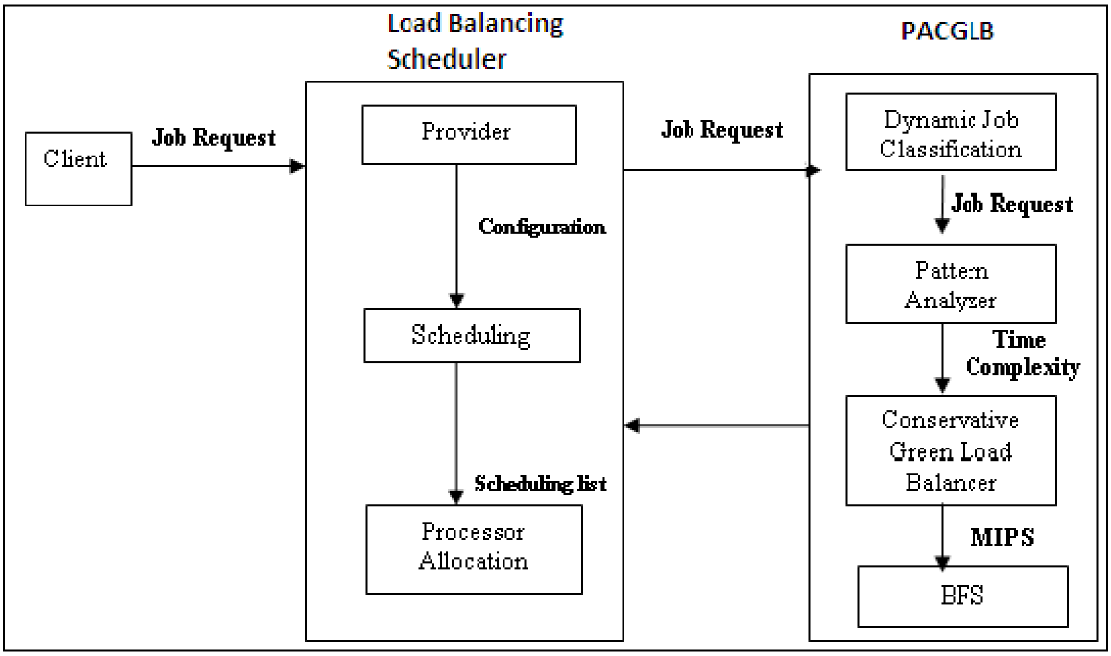

Figure 1: Framwork for Energy Aware Load Balancing Scheduler for green cloud

Load balancing Scheduler

1. Generating the list of all available nodes in cloud
2. Indexing each of the nodes identified based on attraction value

$$\operatorname{attr}(n_i) = \frac{p_i}{cpu_i + mem_i}$$

3. Produce a cartesian value which can define the distance between each node and identify node with the least distance to the request.
4. Rank the nodes based on attr value and distance and find the best node that can be used for processing the incoming job request.

Post the implementation of FA to obtain efficient scheduling framework we have included DVFS as part of the overall model to conserve energy in cloud [5].

Load balancing can be achieved by introducing a Load Balancing (LB) scheduler which is designed to collect the available node configuration from the provider and generate optimized node schedule list using firefly algorithm and submits the list for processor allocation.

Pattern Analyzer with Conservative Green load balancer(PACGLB)

To attain green cloud energy conservation, Dynamic Job Classifier submits the Job to the Pattern Analyzer (PA)[4]. It has been designed to analyze the behaviour of the given cloud request using asymptotic notation. It identifies associated type of algorithm, and determines data size based time complexity following which the Conservative Green Load Balancer computes Micro Instruction Clock Cycle (MICC) count time complexity. Now by multiplying these two parameters Million Instructions Per Second (MIPS) requirement of the incoming job request is estimated. Subsequently best fit strategy is applied to select the appropriate VM.

Energy Aware Load Balancing Scheduler for Green Cloud

Now Energy Aware Load Balancer adapts best fit strategy to identify the appropriate host from generated LB scheduled list and allocate the incoming job to the victimized host. Then PACGLB adjusts the frequency of the host according to the calculated MICC. Hence promotes energy saving.

With the dual benefit of optimised scheduling under FA and energy conservation using DVFS, we obtain better CPU utilisation and faster processing of incoming requests with optimal energy conservation.

The results obtained shows that Energy aware load balancing scheduler for green cloud is superior in overall performance under Jobs processed and time consumed.

RESULTS

As given below in Figure 2 the jobs processed in MIPS (million instructions per second) using firefly model shows a directly proportional increase compared with adaptive genetic algorithm. There is an increase of close 30% higher scheduling under Firefly compared against genetic algorithm.

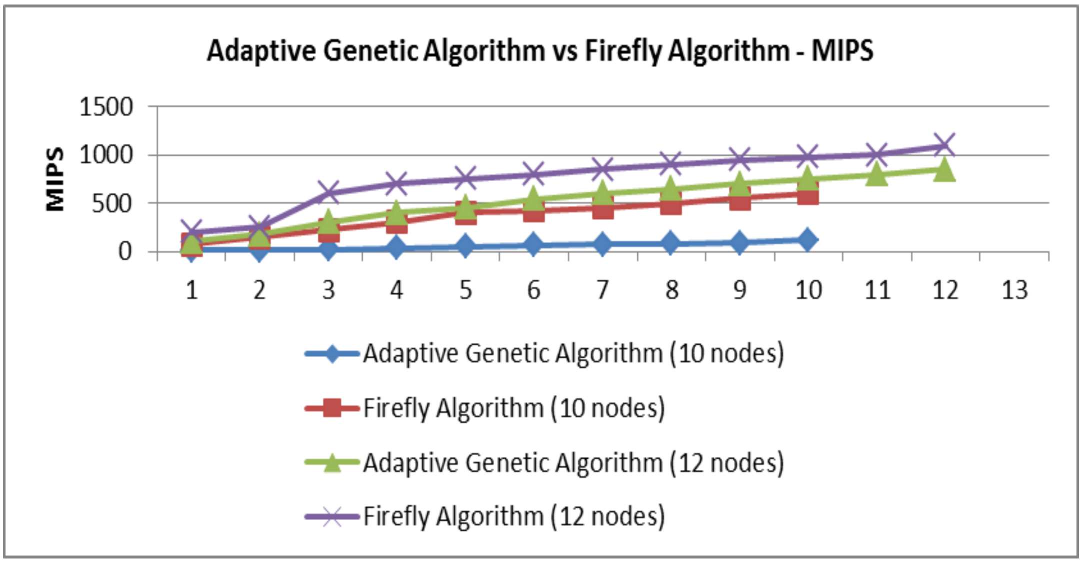

Figure 2: Firefly Algorithm Vs Adaptive genetic Algorithm

In Figure 3 we have considered different job sizes and compared them under Firefly and Adaptive genetic algorithm and it's evident that the response time received under Firefly far lower. This enables for much faster accepting of jobs in cloud which reduces the overall processing time. Even though the response time is similar when the job size is less, it amplifies when job size increases. Table 1 shows job size in MIPS and response time of AGA FA.

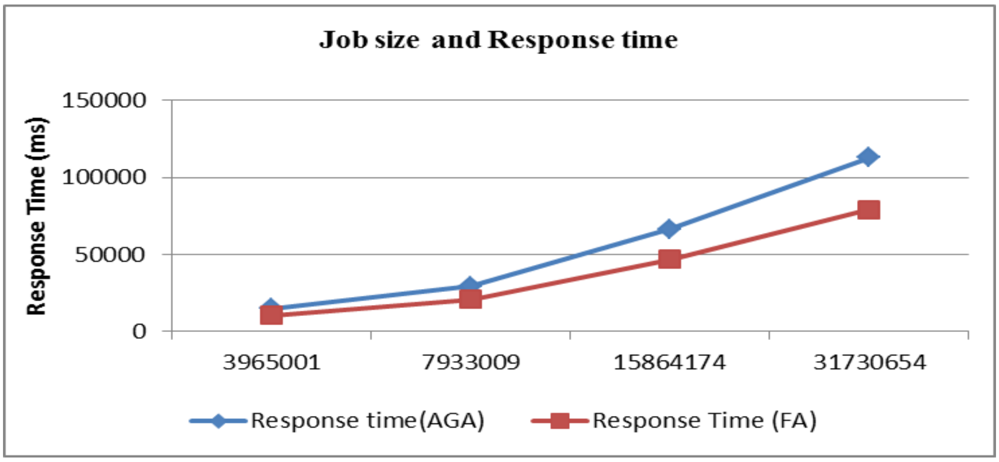

Figure 3: Job size and Response time

Table 1. Job size and Response time

Job Size(mips)	Response time(AGA)(ms)	Response Time (FA)(ms)
3965001	14976	10483
7933009	29421	20595
15864174	66621	46635
31730654	113075	79153

Figure 4 shows that we can save a considerate amount of power if we can include both FA & DVFS as part of the model compared to Adaptive genetic Algorithm. Watts consumed under Adaptive genetic algorithm is nearly 50% higher. Higher the energy consumed will result in energy being generated to be higher and a direct impact on environment. So the combination of Firefly and DVFS implemented in cloud, helps reduce adverse effect cloud data centers could have on the environment.

ISSN: 0975-8585

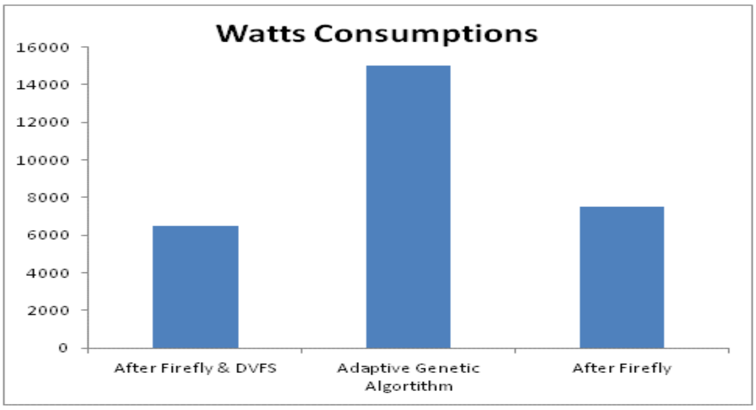

Figure 4: Watts Consumption

CONCLUSION

Cloud computing is the way forward for companies, big or small. However the cost of implementing, managing and running a cloud computing service is wide ranging and has extensive impact on environment by way of heat generation and emission.

To address the issue of environmental impact, we have proposed in the earlier work a two-step methodology which addresses the way any request is serviced 1. By balancing load between nodes and 2. Analyze any incoming request using asymptotic notation and mapping them to the resource without compromising on the speed of processing and the results obtained.

Load balancing scheduler analyses system performance, load of each nodes, compare and effectively balances the incoming job request based on factors such as node proximity using firefly algorithm and maximizes resource utilization by balancing load among resources. As part of balance scheduling, a list is prepared consisting of all nodes currently operational and using each node's CPU, Memory rate along with time factor a scheduling index is calculated. Once we have the above statistics, node with minimum load is selected, basis its proximity and attraction factor and the incoming request is allocated using optimized scheduling mechanism. The incoming algorithm is first analyzed by constructing a data flow table. Then by using a data flow graph the type of algorithm in the incoming request is mapped and its time complexity calculated. Now the load balancer takes over and allocates the request to the node with the least distance and maximum capacity.

This process/methodology helps in ensuring a better resource allocation and utilization which results in reduced consumption of resources and energy.

REFERENCES

[1] Peter M, Tim G. The NIST Definition of Cloud Computing, Information Technology Laboratory 2009.
[2] Rajkumar B, Yeo CS, Venugopal S, Broberg J, Brandic I. Future generation Computer Systems 2009; 25: 599 – 616.
[3] Paulin Florence A, Shanthi V. Journal of Computer Science (JCS) 2014; 10: 1156-1165.
[4] Paulin Florence A, Shanthi A, Sunil Simon CB. Scientific World Journal 2016; 1- 13.
[5] Rajwinder K, Prasenjit KP. International Journal of Computer Applications (IJCA) 2016; 76: 61-67.
[6] Kobra E, Mahmoud N, Noorali RY. A hybrid min-min max-min algorithm with improved performance; 1-9.
[7] Upendra B, Purvi N. Ramanuj. International Journal of Application or Innovation in Engineering & Management (IJAIEM) 2013; 2: 259-264.
[8] Dharmesh K, Jaydeep V. International Journal of Scientific & Technology Research(IJSTR) 2014; 3: 115-119.
[9] Jasmeen K, Vinay B. International Journal of Computer Applications (IJCA) 2016;147: 9-13.
[10] Suraj SR,Natchadalingam R. International Journal of Emerging Technology and Advanced Engineering 2014; 4: 350-356.