IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 21, Issue 4, Ser. I (Jul - Aug 2019), PP 31-36
www.iosrjournals.org

Green Cloud Computing Using Genetic Algorithm For Load Balancing By Considering Individual Node Metrics

PranavShankar[1], Vidushi Arora[2], Pratik Joshi[3]
[1](*Department of Computer Science, University Institute of Technology, Panjab University, Chandigarh, India)*
[2](*Department of Computer Science, Chandigarh Group of Colleges, Punjab Technical University, Chandigarh, India)*
[3](*Department of Computer Science, University Institute of Technology, Panjab University, Chandigarh, India)*
Corresponding Author: PranavShankar

Abstract: *This paper proposes a genetic algorithm for load balancing to achieve green cloud computing by making use of the metrics of individual server nodes such as Thermal Design Point and other singly computed metrics as parameter to have a much profound fitness heuristic function for obtaining optimally distributed load vector. Individual node metric consideration provides workload distribution which not only results in minimal energy consumption and lower carbon footprint but also outperforms the standard genetic algorithm-based load balancing techniques.*

Date of Submission: 08-07-2019 Date of acceptance: 23-07-2019

I. Introduction

Development in the field of computation systems stems from the need of improvement in the performance which is further driven by the rise in the demand from scientific, consumer and business domains. However, the augmenting energy usage by computational systems has started to confine the performance levels majorly because of the overwhelming cost of run and its preeminent carbon dioxide footprints. Albeit there has been varied approaches to tackle the problem from altering angles but the focus has been on Green Cloud Computing. The major aim of the Green Cloud Computing approach is efficient utilization of the resources such as servers and storages. Eventually leading to reduced consumption of energy. Another prominent reason for usage of cloud computing is that customers make use of only what they require, paying for what they have actually utilized. The required resources are made available by the means of the Internet Data Centers (IDC) which may cause higher levels CO2 emission due to energy intensive usage, raking in notable portion of disposable energy [1]. The Report by Greenpeace has claimed that demand of electricity of data centers is appraised at about 31 GW globally which is equal to 180,000homes'use[2].Dynamic nature of cloud computing makes the task even more difficult because availability of a particular resource is variable thus before task scheduling, the characteristics and disposition of the cloud should be taken into account.

Computing an optimal task schedule is a NP-complete problem [4],[5]. To tackle the problem, numerous heuristics have been proposed over time based on techniques including integer programming, searching, branch-and-bound, graph theory, randomization, hybrid methods, genetic algorithms, and evolutionary methods. Genetic Algorithms are designed to balance load in clouds so that the excess dynamic local workload is distributed evenly across all the whole network. Doing so, a much higher resource utilization ratio and user satisfaction is achieved. Load balancing bears an optimal utilization scheme for the given nodes, resulting in minimal resource dissipation. Its importance is extended in implementing failover, bottlenecks avoidance, over-provisioning, higher scalability and reduced response time etc. Thus, load balancing is imperative for procuring Green cloud environment as load balancing curtails energy consumption by circumventing overheating and further lowers carbon emissions by reducing energy use and further the carbonfootprint.

Further, the percussion on the carbon footprint is also affected by the varied configurations of the systems. For instance, Thermal Design Point or the Maximum power consumed by a CPU or GPU varies drastically as a GeForce 9800 GTX uses 156 Watts whereas a Radeon HD 4870 X2 uses 286 Watts. Similarly, Software types and versions such as different operating systems like Unix and Nemesis have varied impacts on power and resource consumption. This paper aims to achieve green cloud computing by presenting a load balancing algorithm for computing optimal distribution strategies possible for the given types of nodes/servers.

The Cloud Network simulated in this paper is assumed to have the following type of servers [17], [18] and their TDP in Watts (Thermal Design Power, maximum power consumption in active state).

Table 1. TDP values of Server/Node types used for cloud network simulation.

Radeon HD 4870	Radeon HD 3870	Geforce GTX 295	Geforce GTX 280	Geforce 9800 GTX	Phenom II X3 740	Core i5- 2xx Gen	Athlon II X4 650
160	105	289	236	156	95	95	95

Albeit there are several works presented in this area (as expounded in section 2), however the indispensable problems of these works have been their inability to consider the impact of the individual node configuration on the network along with their impotence to be applicable in real time. Our work makes use of an efficient fitness function for computing an optimal distribution for the provided workload into servers or nodes in real time. The Genetic Algorithm proposed uses Dynamic Performance Scaling (DPS) and uses Dynamic Component Deactivation (DCD) for avoiding overheating by balancing load according to the fitness value computed, hence reducing significant amount of resourcesexpenditure.

Paper comprises of five major sections starting from introduction which is followed by related works and proposed work. Succeeded by results and conclusions which are further followed by the acknowledgements and at last thereferences.

II. Related Works

Genetic Algorithms have been widely in the recent years due to their capability to provide optimization and search techniques which remains to be a colloquial issue for varied tasks. Thus, several general and task oriented genetic algorithms have been proposed such as resource scheduling for balancing VM load in the clouds by Zhongni et al. [6], using Genetic algorithm. Singh et al. [7], delineates several algorithms for job scheduling and furthers draws comparison between these algorithms. To conclude that an optimal scheduling algorithm increments should schedule resources for optimal usage only. Portaluri et al. [8] presented a genetic algorithm task management in a manner to make judicious use of power. Development of the approach ensured scalability and performance of system under consideration. Approach proposed, tackled allocation of resources for data centers in the cloud which perform joint allocation of network and computational resources. Fang et al. [9] proposed an architecture for dealing with the job scheduling problems specifically for groups of requests from cloudusers. Provisioning of resources has been given prime importance with and aim to lower average tardiness of requests related to connection

Hu et al. [10] presented an approach for virtual machine (VM) load balancing by making use of Genetic Algorithm. Approach aims for a proper load balancing whilst lowering dynamic migration. This resolves issues of imbalance of load and obtains better rate of utilization. Further, noticeable work for energy consumption in the cloud can be observed in the paper by Sahu et al. [11]. They presented an approach for minimizing the number of machines to be turned on the basis of dynamic comparison against some threshold.

Quang-Hung et al. [13] proposed Genetic algorithm for power aware (GAPA). GAPA aimed for power aware allocation of virtual machines for reducing energy consumption when used with SVMAP methodology. Pushpendra et al. [14] made use of Divisible load scheduling theorem for achieving extremes of parameters under consideration such as latency and through put with clouds of varied sizes. Ramietal. [15] proposed a sequence algorithm for task scheduling in multiprocessors systems by minimizing time of completion of each task and augmenting the throughput. For achieving the same, algorithm comprised of a unique form of initialization of the chromosomesandsystematicmethodologyforthecrossoveroperation.Randlesetal. [16] presented algorithm for scheduling of resource and balancing of load for large- scale distributed computation models such as in cloud computing, aiming for efficient allocation of resources as in jobs scheduling in the distributed computation model.

III. Proposed Work

The algorithm proposed is based on the general architecture of the Genetic Algorithm but in order to achieve its purpose of Green Cloud Computing, optimal usage of energy is achieved majorly by efficient load balancing which involves performing Dynamic Performance Scaling (DPS) and Dynamic Component Deactivation (DCD). Algorithm is divided into multiple steps or phases as depicted in Fig. 1. Each phase comprises of several sub steps, acting as necessary ancillary measures.

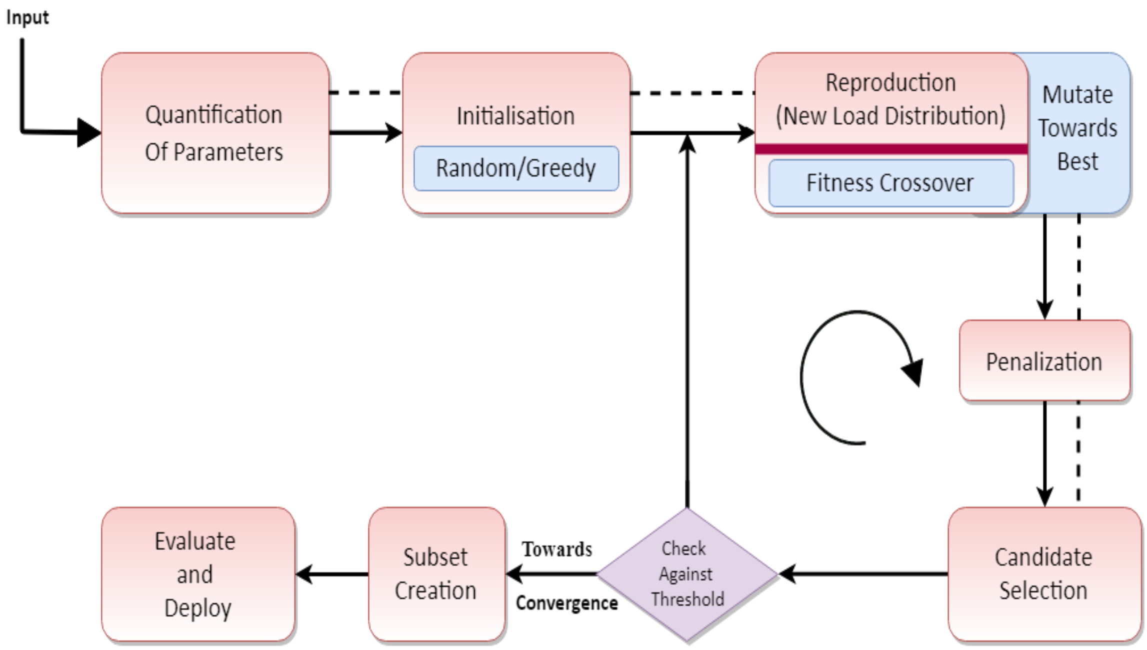

Fig. 1. Different phases involved for obtaining Optimal Load Balance Configuration.

Step 1 Quantify the Parameters which are to be considered for balancing the load amongst the servers in the cloud. Different server systems have different utility according to their resource requirement and energy usage. Algorithm provides the provision to incorporate one or more such factors such as Thermal Design Power (TDP), Data Center infrastructure Efficiency (DCiE), Power usage Effectiveness (PUE), Performance per Watt (PpW), Energy Reuse Factor (ERF), Compute Power Efficiency (CPE), Data Center Productivity (DCP), Green Energy coefficient (GEC), Water usage Effectiveness (WUE), Carbon usage Effectiveness (CUE). These tend to be an antecedent in the overall measure of the load thus balanced and impact of such a distribution in forms of performance apace with energy consumption.

Step 2 Initialization involves evaluation of the fitness value per server or node available which is to be considered for balancing load in each crossover set attained. *Fitness Evaluation:* Fitness value is computed from the load to be distributed factored by the average of the parameter under consideration.
Each node in the population M, annexed fitness value is assigned by equation:

$$\text{Fitness}(x) = \sum (\text{ABS}(av_L - L)\,\delta)\ . \qquad (1)$$

Where Li is the workload of current node in cloud and 'av_L'is the average load of generation. Multiplicative factor δ accommodates the measure(s) to be considered while distribution of load givenas:

$$\delta = \sum_i W_i\,(x_i / \sum x)\ . \qquad (2)$$

Where 'x' represents the measure such as Thermal Design Power (TDP) or Power usage Effectiveness (PUE) etc. Ratio of an individual measure with summation of the same measure of all nodes is multiplied with the weight factor W (taken as 1 below).
External summation represents the instance where multiple measures are taken as parameters which are to be considered while balancing the load.
Further, if 't' is the length of current individual (x) then $n/2 <= t <= n$. Individual having smaller fitness value delineates higher probability to get transformed into the following generation. For depicting fitness variation amongst overall population, the fitness value of an individual is evaluated proportional to fitness summation of all nodes or individuals in the population given as:

$$\text{Fitness}(x) = \text{fitness}(x) / \sum_j \text{fitness}(j)\ . \qquad (3)$$

Step 3 The core of the algorithm is the Genetic Function Phases which comprises of: *Reproduction:* The individual having the best fitness value is selected for transmission and creation of the next generation.

Crossover: On the basis of the fitness values, a crossover is attained. Cartesian Product tends to be the best suited measure for achieving crossover. Algorithm comprises of implicit *Mutation* beheld upon the generated set. Consider the child generated from parents x,y.

1. If x = y, then the child is the mutation ofx.
2. Else, the child has the shape of y, and the values contained in it are determined by using time as the random entity, where the value is inherited for x or y using a time variantcondition.

Preceding step is repeated till convergence or till threshold function is satisfied to attain a certain level of optimality in the sets being produced. To achieve the same, algorithm encompasses provision to penalize the similar sets and then select the best candidates to engender the next generation is possible.

IV. Results

The performance of the algorithm is evaluated by comparing with the Standard Genetic Algorithm [19], [3] for load scheduling using Power Consumption and Degree of Imbalance [12] asmetrics.

$$\text{Degree of Imbalance} = (T_{max} - T_{min}) / T_{avg} \,. \qquad (4)$$

Where Tmin, Tavg and Tmax are the minimum, average and maximum time for execution respectively amongst allnodes.

Table 2. depicts the observed maximum generations optimal convergence for different generations which are considerably less than the Standard Genetic Algorithm [19].

Table 2. Observations for fixed (k = 8) and variable (k = 4,6,8) Generation size.

Load	160	400	800	960	1200
MaxGen (k=8)	24	29	37	41	40
Max Gen (k=4,6,8)	14	26	25	26	26

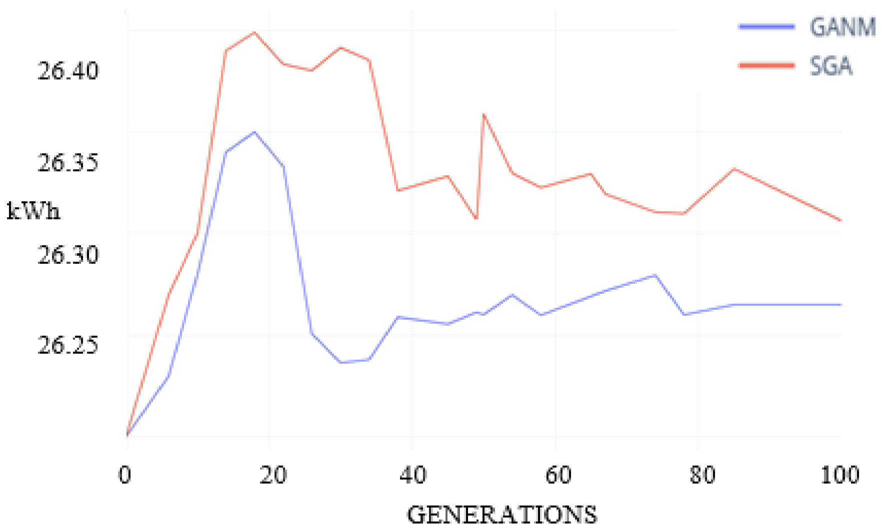

Fig. 2. GenerationsvsPower Consumption

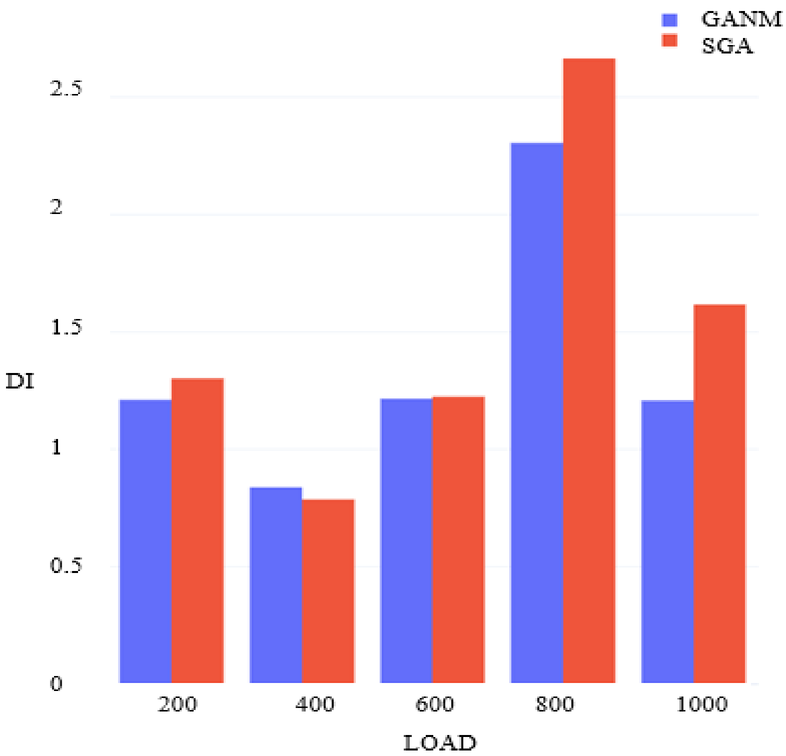

Fig. 3. Load vs Degree of Imbalance(DI)

Fig 2. depicts comparison between Proposed Genetic Algorithm with Node wise Metrics (GANM) and Standard Genetic Algorithm (SGA) [19], [20], [3]. Despite the presence of various metrics for comparison, power consumption appears to be most abstract and useful. The trend in power consumption by the cloud network delineated in Fig 2. clearly represents that GANM outperforms SGA thus GANM is more promising for green cloud computing. Fig 3. illustrates the capability of GANM to produce highly balanced load vectors in most of the scenarios. Degree of Imbalance also depicts that individual node metrics tend to have a much higher impact on the processes and measures considered in previous algorithms to perform load balancing.

V. Conclusions and Future Works

GANM provides optimal load vectors with different load values. Further, the problem of overheating is also tackled by efficient workload balancing amongst servers. Hence lowering power requirements. The existent techniques for load balancing in clouds, fail to consider metrics such as Thermal Design Power (TDP), Data Center infrastructure Efficiency (DCiE), etc. which are considered in the proposed algorithm, yielding promising results.

We aim to extend the technique to incorporate objective based fitness parameters in the function to attain finer and more realistic computations such as time-based node wise dynamic load transfer amongst servers. In addition, we aim to develop a crossover mechanism for handling much higher levels of computation for inter-cloud and intra-cloud load distribution with lesser complexity then the one being used in the proposed algorithm.

Acknowledgments

We acknowledge the support and guidance of the faculty of Computer Science Department, University Institute of technology, Panjab University for providing the necessary guidance and motivation to conduct the research.

References

[1]. Dung H. Phan, Junichi Suzuki, Raymond Carroll, et al. Evolutionary multi objective optimization for green clouds GECCO 12: The proceedings of the 14th annual conference companion on Genetic and evolutionary computation. NY USA. 2012
[2]. G. Cook, "How clean is your cloud," Catalysing an energy revolution,2012.
[3]. E. Hou, N. Ansari, and H. Ren, " A Genetic Algorithm for Multiprocessor Scheduling," IEEE Transaction on Parallel and Distributed Systems,1994.
[4]. Y. kwok and I. Ahmad, "Static Scheduling Algorithm for Allocating Directed Task Graph to Multiprocessors," ACMComputing

Surveys, vol. 31, no. 4, December1999.
[5]. K. Kaur, A. Chhabra and G. Singh, "Modified Genetic Algorithm for Task Scheduling in Homogeneous Parallel System Using Heuristics," International Journal of Soft Computing, pp. 42-51, vol. 5, no. 2, 20 IO.
[6]. Zhongni ZH, Wang R, Hai ZH, Xuejie ZH, "An approach for cloud resource scheduling based on parallel genetic algorithm". IEEEICCRD.
[7]. Singh RM, Sendhil Kumar KS, Jaisankar N. "Comparison of probabilistic optimization algorithms for resource scheduling in cloud computing environment". International Journal of Engineering and Technology (IJET).2013
[8]. Portaluri, Giuseppe and Dzmitry Kliazovich, A power efficient genetic algorithm for resource allocation in cloud computing data centers, In IEEE 3rd InternationalConference.
[9]. Fang Y, Wang F, Ge J. A task scheduling algorithm based on load balancing in cloud computing. Springer Web Information Systems and Mining.2010.
[10]. Hu, J., Jianhua, G., Guofei, S., Tianhai, Z.: A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment. In: IEEE PAAP. pp. 89–96. Dalian, China (2010).
[11]. Sahu Y, Pateriya RK, Gupta RK. Cloud server optimization with load balancing and green computing techniques using dynamic compare and balance algorithm. CICN' 13 Proceedings of the 2013 5th International Conference on Computational Intelligence and Communication Networks, IEEE Computer Society Washington, DC, USA. 2013
[12]. Mohammad Shojafar, Saeed Javanmardi, Saeid Abolfazli and Nicola Cordeschi. FUGE: A joint meta-heuristic approach to cloud job scheduling algorithm using fuzzy theory and a genetic method. Springer Science+Business Media New York2015.
[13]. Quang-Hung, Nguyen, Pham Dac Nien, Nguyen Hoai Nam, Nguyen Huynh Tuong, and Nam Thoai. A genetic algorithm for power-aware virtual machine allocation in private cloud. In Information and communication technology. Springer Berlin Heidelberg,2013.
[14]. Pushpendra Verma, Jayant Shekhar, Amit Asthana. A model for evaluating and maintaining load balancing in cloud computing. IJCSMC.2014.
[15]. Rami A. AI Na'mneh and Khalid A. Darabkh. A New Genetic-based Algorithm for Scheduling Static Tasks in Homogeneous Parallel Systems. (ROBIONETICS) Yogyakarta, Indonesia, November25-27,2013
[16]. Randles, M., Lamb, D., Taleb-Bendiab, A.: A comparative study into distributed load balancing algorithms for cloud computing. In: IEEE Advanced Information Networking and Applications Workshops (WAINA), pp. 551–556. WA, Perth(2010).
[17]. AnubhaJain,ManojMishra,SateeshKumarPeddojuandNitinJain."EnergyEfficientComputingGreen Cloud Computing", ©2013 IEEE
[18]. Tutorial: What is the Thermal Design Power (TDP), by JeGX, Publication Date: 06/11/2009, Retrieved Date: 13/03/2017
[19]. Nada Al Sallami and Sarmad Al Aloussi. A Genetic Algorithm in Green Cloud Computing, British Journal of Applied Science & Technology 179-185,2015.
[20]. M. Mohamed and M. Awadalla, "Hybrid Algorithm for Multiprocessor Task Scheduling" International Journal of Computer Science, pp. 79-89, vol. 8, no.2, May2011.

IOSR Journal of Computer Engineering (IOSR-JCE) is UGC approved Journal with Sl. No. 5019, Journal no. 49102.

PranavShankar. " Green Cloud Computing Using Genetic Algorithm For Load Balancing By Considering Individual Node Metrics" IOSR Journal of Computer Engineering (IOSR-JCE) 21.4 (2019): 31-36.