Energy Aware Resources Allocation Heuristic For Efficient Management of Data Centers For Cloud Computing

Sarbjot Kaur, Amandeep Kaur
Research Scholar, Assistant Professor
Department of Computer Science & Engineering,
Desh Bhagat Unversity, Mandi Gobindgarh

Abs*tract* - Scheduling is very important for cloud provider to accomplish with effort producing a lot with very little waste valuable supply combining and ability to stretch. Scheduling problem becomes very clearly connected in computers that do work for you, but that are stored somewhere else and maintained by other companies situations where the providers have to operate at very much producing a lot with very little waste to be competitive and take advantage at scale. The wide acceptance of computers that do work for you, but that are stored somewhere else and maintained by other companies means data centers with many more machines and the usage model is much different than traditional groups, for example hour edges/borders, sale to the highest payer based prices etc. This way scheduling in cloud data center is more challenging than traditional group schedulers. Also, these data center run many different kinds of applications with different expectations from basic equipment needed for a business or society to operate. Valuable supply usage patterns in traditional data centers are have less variance in than the unpredictability faced by cloud data centers. The way operating system scheduler tries to improve as much as possible the use of useful things/valuable supplies on a single machine, (in almost the same way) cloud schedulers tries to improve the use of data center as a whole. It is clear that in such surrounding conditions, the role of a scheduler becomes very important in accomplishing or gaining with effort high use without producing/making happen application performance.

***Keywords* - Scheduling, Data centers, Performance, accomplishing**

I. INTRODUCTION

In today's era, Cloud computing becomes a significant part in the computing world. The usages of the cloud computing has been increasing day by day. It is a technology to give the several services over the internet. Cloud is generally acted as data centre and it is combination of some concepts like virtualization, networking and etc. So many users are dependent on the application of cloud computing. Cloud computing helps to minimize the cost overhead of users. Cloud computing provides the three type of services to the user i.e. infrastructure as a services, platform as service, and software as services. In infrastructure as service, is a model for the cloud services which deal with cloud infrastructure services such as accessing, monitoring and managing data storages. Software as a Service is also referred as cloud application services. This type of service model consists of the majority of the cloud service provider (CSP) in cloud market. This model is a software distribution model where software are hosted and managed by cloud providers, for the clients over the network. Platform service is referred as cloud platform services. In this model Cloud Service Provider (CSP) provides cloud component to software for application and other developments to the users. Users can use PaaS framework for developing, running and customizing applications. Platform as a Service model is used for development, testing, running, customizing and deployment of applications.

Cloud load balancing is one of the significant category of load balancing which can be performed in cloud computing. Load balancing is the method of distributing the workload across several resources for computing such as CPU, computers, hard disks, disk drives and many more. It also helps to minimize the cost which is linked with the management system and maximize the resources availability. The major goal of this load balancing approach is to optimize the resource use, reduce the response time, enhancing throughput. Also it helps to avoid the overload of any single resource. By using multiple numbers of components with load balancing it can enhance the reliability and availability through redundancy. Load balancing usually involves dedicated software or hardware, such as a multilayer switch or a Domain Name System server process. In this, load balancing segregate traffic among network interfaces on a network socket basis. And in channel bonding, traffic is segregated among the physical interfaces at a lower level, either per packet or on a data basis with a protocol like bridging. The mainly used application of this load balancing technique is to give a single Internet service from multiple servers, sometimes known as a server farm. Some common load-balanced systems involves popular web sites, large Chat networks, high-bandwidth File Transfer Protocol sites, Network News Transfer Protocol (NNTP) servers, Domain Name System (DNS) servers, and databases.

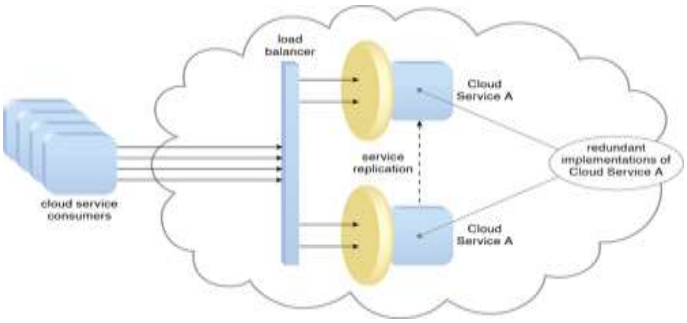

Fig. 1:

As figure 1 depicted that The load balancer forwards requests to one of the "backend" servers, that gives reply to the load balancer. This allows the load balancer to reply to the client without the client ever knowing about the internal separation of functions. It also prevents clients from contacting back-end servers directly, which may have security benefits by hiding the structure of the internal network and preventing attacks on the kernel's network stack or unrelated services running on other ports.

ACO based scheduling algorithms

Ant Colony Optimization (ACO) scheduling approach is motivated by the activities of ants detecting the shortest path among their colonies and a source of food. While walking along with their colony and the food source, ants leave pheromones on the ways they move. The pheromone intensity on the way growing with the numbers of ants passing through and drops with the evaporation of pheromone. After sometime, pheromone intensity helps ants to find their smaller paths to the food source. These algorithms are used for solving discrete optimization problems that need to detect the path.

GA based scheduling algorithm.

GA based algorithm represents a population based optimization method based on a metaphor of the evolution process observed in nature. In GA, each chromosome (individual in the population) represents a possible solution to a problem and is composed of a string of genes. The initial population is taken randomly to serve as the starting point for the algorithm. A fitness function is defined to check the suitability of the chromosome for the environment. On the basis of fitness value, chromosomes are selected and crossover and mutation operations are performed on them to produce offspring's for the new population. The fitness function evaluates the quality of each offspring. The process is repeated until sufficient offspring are created.

PSO based scheduling algorithm

Particle Swarm Optimization (PSO) is introducing an evolutionary computational technique. Every particle is associated with position, velocity and moves through a multi-dimensional search space. In iteration, every particle change its velocity depending on its suitable position and the position of the appropriate particle of the whole population. PSO combines global search methods with local search methods and trying to balance exploration and exploitation. PSO become popular because of its simplicity and its usefulness in broad range of applications with low computational cost.

Firefly Algorithm

Firefly algorithm (FA) is a easiest and efficient nature-inspired search technique for global optimization. Firefly algorithm becomes more popular since the it was developed and also it grab attention of several researcher. Firefly algorithm has capability to solve the numerous real world problems. It can be said that firefly algorithm is a swarm-based intelligence algorithm, which reproduce the flashing behaviour of fireflies.

II. RELATED WORK

IztokFister,Xin-She yang, Jamez Brest et. Al [7] presented a survey of swarm intelligence that indicates that the firefly algorithm may be applied to each issue occurs in practice. Moreover it supports new developers and researcher to use efficient, simple and accurate algorithm for solving the problem.

Mohsen, HosseinDeldaet.al [18] proposed a system of adaptive fuzzy ants to face the load balancing challenge. In this type of environment ant may committing suicide based on their existing condition. Analysis has been done to verify this new strategy which surpasses its predecessor. In order to implement the new mechanism, lots of work is to be done.

Ngaam J Cheung, Hong bin shen, Xueming ding et.al [9] in this paper adaptive firefly algorithm has been proposed. The several parameters has also been selected for the proposed mechanism. In this paper, three schemes of adaptive firefly algorithm are being presented which includes 1. Coefficient of distance based light absorption. 2. Coefficient of enhancing fireflies to share information. 3. Five approaches for the randomization parameter. The parameters have been selected in order to get better performance of adaptive filter. The proposed algorithm has been used for numerical experiments, statistical test and used for selecting parameter which effect the performance of adaptive firefly approach.

Meenakshi Sharma, Sandeep Sharma, Sarbjit Singh etal. [22] In this paper load balancing algorithm has been proposed. Analysis of the static, dynamic load balancing approach has been studied. The several parameters such as precision, stability, overload rejection etc has considered for the comparison and for the performance analysis. Load balancing algorithm is selected depending on the condition in which work load is assigned i.e. at run time or compile time. Static algorithms are proved to be more stable as compare to dynamic load balancing algorithms.

Parveen Jain ,Daya Gupta et.al proposed [20] proposed an approach for distributed system with the help of supporting nodes by developing the interrupt service. This proved that the efficiency of the system has been improved by replacing the centralized node with a number of nodes added with interrupt service. This approach can help to minimize the waiting time by significant amount of time.

Mala kalra, Sarbjeet Singh et. al[21] in this paper review of several scheduling approaches has been discussed which is used for cloud and grid environment depends on three popular meta-heuristic techniques: Ant Colony Optimization (ACO), Genetic Algorithm (GA) and Particle Swarm Optimization (PSO), and two novel techniques: League Championship Algorithm (LCA)and BAT algorithm.

III. PROPOSED METHODOLOGY

Decision making is a process of an individual in a society which is governed by his own considerations or the opinions of others. Opinions are formed by the direct or indirect influence of cultural interactions, norms and mass media. Social influence is a combined effect of these influences, due to these, individuals act in accordance to the expectations ad beliefs of others. This forms the third rudiment of an algorithm. However, for simplicity of the modeling, only local dynamics representing is a social influence which has been considered in this work. Therefore, the social influence has been formulated by considering two factors example distance between two individuals and the social ranking of the individuals 11. The social ranking of the individuals is determined by their respective fitness values. These fitness values are the output values of an objective function to be minimized. The individual with minimum fitness value is assigned the largest SR, the highest possible SR being the number of individuals. The individuals with same fitness values are assigned the same SR.

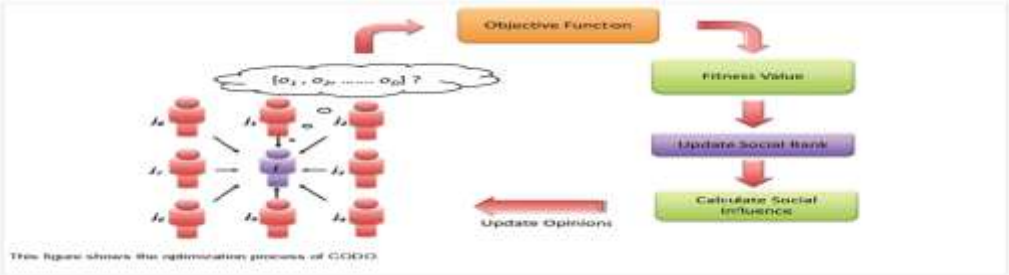

Figure 2: CODO schema

It shows the simplified schema of the CODO process. At particular time/iteration t, the opinion vectors of all the individuals are executed by the objective function. The fitness values thus obtained are used to assign a social rank to each individual. The lower fitness value, the higher rank and vice versa. Individuals with the same fitness values extract the same social rank. The social influence of each individual is then determined. At the end of iteration, the opinion vector of an each individual is updated. This iterative process has been terminated after attaining preset fitness error value or maximum number of objective function evaluations. The proposed optimizer has been tested on benchmark functions for 10 and 30 dimensions or compared with local best PSO, one of the variants of popular PSO. A best PSO has been selected for comparison due to its related structural similarity to the proposed algorithm; in general, the proposed algorithm runs better as compared to best PSO. The algorithm has been investigated for effects of disintegrative forces in the society. The conclusion suggests that, the interplay or balance of integrative and disintegrative forces in the society may be utilized for solving the complex mathematical problems.

IV. RESULTS

To carry out the simulation of the proposed algorithm of Adaptive Firefly in load balancing over cloud computing, we have used Cloud-Analyst developed by cloud-bus using basic cloud-sim toolkit which provide and interface for developing and integrating your proposed approach. In this simulator to get appropriate result we have to set the configuration of the Virtual machine at the end of cloud server over the Datacenter part of cloud. Different result has been evaluated using different environment for the same algorithm. In Cloud-Analyst, we have to set the number of datacenter over the world and the configuration of virtual machine which has been placed over every datacentre. Along with that we need to create client i.e. user base, which actually request datacentres depending upon the policy for choosing datacentre, here we are using closed datacentre policy to select the datacentre to execute user's request over the cloud.

Three different Configurations (C_1, C_2, C_3) has been set to get the desired output of the simulator. The actual parameter of both the configurations is provided in the table below

Memory (Mb)	Storage (Mb)	Available BW	No. of Processor	Processor speed	VM Policy
204800	100000000	1000000	4	10000	TIME_SHARED

Table 1 Physical hardware detail of datacentre

Arch	OS	VMM	Cost per VM	Memory cost per VM	Storage Cost per VM	Datacentre transfer cost	Physical HW unit
X86	Linux	Xen	0.1	0.05	0.1	0.1	2

Table 2Datacenter configuration

Requests per User per Hour	Data size per request	Peak hour start	Peak hour end	Average peak user	Average off peak user
60	100	3	9	1000	100

Table 3 Userbase configuration

Configuration	No. of User base(UB)	No. of Datacentre (DC)
C_1	25	10
C_2	25	15
C_3	30	20

After setting the defined configuration over the cloud in the cloud-analyst different result has been came out which has been displayed in figure1. The graph represents the Overall response time and Datacenter processing time against the total execution time for the simulation in milli-seconds. The graphs shows that over all response time kept on decreasing as number of datacentre increases because of the sharing of the load among datacentre which provides parallelism and saves execution time for the job or request. Since request has been shared among different datacentre, the number of request per datacentre become less and hence reduces the processing time of datacentre for the request.

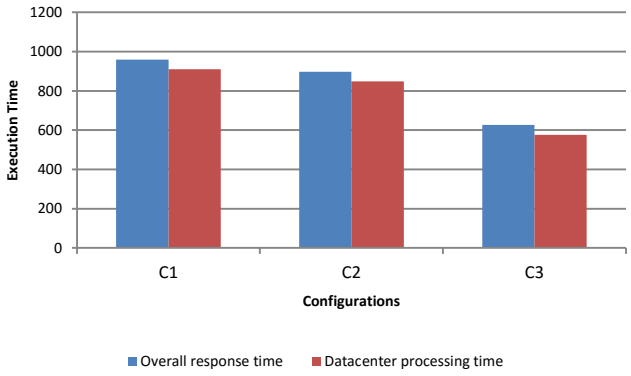

Figure 3: graphical representation of results

V. CONCLUSION

Efficient virtual machine scheduling has become a vide area of research in the field of cloud computing where actual computation happened over cloud. All the task and jobs are executed over these virtual machines, So it become more important to utilize these machines over the cloud effectively with the recently coming optimization technique. Here Adaptive firefly algorithm had been proposed which showed a quite supporting result as compared to the others. Since various optimization algorithm is being coming up in near future which shows that there won't be any end until researcher are keep finding the relation between the various nature's algorithm. So there won't be an end to keep finding the better virtual machine scheduling over the cloud computing to balance the load over the cloud.

REFERENCES

[1] IztokFister,Xin-She yang, Jamez Brest et. al A comprehensive review of firefly algorithms,dec.2013

[2]Mohsen &HosseinDelda"Balancing Load in a Computational Grid Applying Adaptive,Intelligent Colonies of Ants".Informatica32 (2008) 327–335.

[3]Ngaam J cheung, Hong bin shen, Xuemingding.Adaptive Firefly Algorithm: Parameter Analysis and its Application**.** IEEE ProceedingComputer Digital Tech 2006,pp.373–80.

[4]Sandeep Sharma, Sarabjit Singh and Meenakshi Sharma," Performance Analysis ofLoad Balancing Algorithms" World Academy of Science, Engineering and Technology,Vol:2 2008-02-21.

[5] Parveen Jain, Daya Gupta. An algorithm for dynamic load balancing in distributed s**y**stems with multiple supporting nodes by exploiting the interrupt service.in international journal of recent trends in engineeruing,may 2009,vol 1,no. 1,pp 232-236.

[6] Mala Kalra, Sarbjeet Singh. A review of metaheuristic scheduling techniques in cloud computing, in Egyptian informatics journal, 2015, pp.1-20.

[7] Yu J, Buyya R, Ramamohanarao K. Workflow Scheduling Algorithms for Grid Computing. Metaheuristics for Schedulingin Distributed Computing Environments. Springer; 2008, p. 173–214.

[8] Bagherzadeh J, MadadyarAdeh M. An improved ant algorithmfor grid scheduling problem using biased initial ants. In: 3rd intconfcomput res dev; 2011. pp. 373.

[9] Braun TD, Siegel HJ, Beck N, Bo¨lo¨ni LL, MaheswaranM,Reuther AI, et al. A comparison of eleven static heuristics formapping a class of independent tasks onto heterogeneousdistributed computing systems. J Parallel Distribution Computer 2001,vol.61,pp.810–37.

[10] Xhafa F, Abraham A. Computational models and heuristicmethods for Grid scheduling problems. FuturGenerComputSyst 2010;