RESEARCH ARTICLE OPEN ACCESS

Performance Evaluation of Hybrid DE-GA based Load Balancing in Cloud Environment

Pooja Mangla[1], Dr. Sandip Kumar Goyal[2]
[1,2](Computer Department, Maharishi Markandeshwar University, Mullana)

Abstract:

Cloud computing is a new computing paradigm that, just as electricity was firstly generated at home and evolved to be supplied from a few utility providers, aims to transform computing into a utility. It is a mapping strategy that efficiently equilibrates the task load into multiple computational resources in the network based on the system status to improve performance. The objective of this research paper is to show the results of Hybrid DEGA, in which GA is implemented after DE.

Keywords — **Cloud Computing Load Balancing, Genetic Algorithm, Differential Evolution**

I. INTRODUCTION

With the exponential rise in the demand of clients worldwide, a large scale distributed systems have been introduced as a computing environment. Cloud computing is one of the emerging technologies, as a new paradigm of large scale distributed computing. It has moved computing and data away from desktop and portable PC's, into large data centers [1].

Today a lot of people are consulting their mail online through webmail clients, writing collaborative documents using web browsers, creating virtual albums to upload their photos of the holidays. They are running applications and storing data in servers located in Internet and not in their own computers. Something as simple as enter in a web page is the only thing a user needs to begin to use services that reside on a remote server and lets him share private and confidential information, or using computing cycles of a pile of servers that he will ever see with his own eyes. And every day its being used more this services that are called cloud computer services. That name is given because of the metaphor about Internet, as something than the user see like a cloud and cannot see what's inside.

Cloud computing is a new computing paradigm that, just as electricity was firstly generated at home and evolved to be supplied from a few utility providers, aims to transform computing into an utility. It is being forecasted that more and more users will rent computing as a service, moving the processing power and storage to centralized infrastructures rather than located in client hardware. This is already enabling startups and other companies to start web services without having to invest upfront in dedicated infrastructure. Unfortunately, this new model also has some actual and potential drawbacks and remains to be seen whether concentrating computing at a few places is a viable option for everyone. Consumers are not used to renting computing capacity. The question of how to measure performance is already a major issue for cloud computing customers.

II. LOAD BALANCING

Load balancing [2] is one of the generic term, which is used for distributing a larger processing load to smaller processing nodes to enhance the overall performance of the system. Load Balancing is crucial to computational grids. It is a mapping strategy that efficiently equilibrates the task load into multiple computational resources in the network based on the system status to improve performance. In computer networking, load balancing is a technique to spread work between two or more computers, network links, CPUs, hard drives, or other resources, in order to get optimal resource utilization, throughput, or response time. Using multiple components with load balancing, instead of a single component, may increase reliability through redundancy.

Load balancing can be achieved on a busy system by arranging for more than one program instance to service a queue. In a cloud environment, executing application requests on underlying grid resources consists of two key steps. The first, which we call VM Provisioning, consists of creating VM instances to host each application request, matching the specific characteristics and requirements of the request. The second step is mapping and scheduling these requests onto distributed physical resources (Resource Provisioning). Most virtualized data centers currently provide a set of general-purpose VM classes with generic resource configurations, which quickly become insufficient to support the highly varied and interleaved workloads. Furthermore, clients can easily under- or overestimate their needs because of a lack of understanding of application requirements due to application complexity and/or uncertainty, and this often results in over-provisioning due to a tendency to be conservative.

Load balancers can work in two ways: one is cooperative and non-cooperative. In cooperative, the nodes work simultaneously in order to achieve the common goal of optimizing the overall response time. In non-cooperative mode, the tasks run independently in order to improve the response time of local tasks.

III. IMPLEMENTATION

In the proposed model, hybrid DE and GA are used [5]. In this, Differential Evolution starts upto the point where the trial vector is generated. If that vector satisfies the equation, then it is included in the population otherwise algorithm enters the Genetic algorithm phase and generates a new candidate solution.

1. Sampling the search space at multiple, randomly chosen initial points i.e. a population of individual vectors.
2. Differential evolution is a nature derivative-free continuous function optimizer, it encodes the parameters as a floating-point numbers and manipulates them with simple arithmetic operations. For this differential evolution it mutates a (parent) vector in the population with a scaled difference of the other randomly selected individual vectors.
3. The resultant mutation vector is a crossed over with corresponding parent vector to generate a trial or a offspring vector.
4. Then, finally it takes a decision in a one-to-one selection process of each pair of offspring and parent vectors.
5. If the best population is generated, it is taken into consideration; otherwise the population is generated by using Genetic algorithm.
6. In Genetic algorithm, crossover and mutation operations are applied on the candidates and new candidate generation is achieved.
7. The one with a better fitness value survives and enters the next generation.

Fig. 1 Pseudo-code of DEGA model.

The pseudo-code for efficient load balancing using DE & GA, shown above, defines the working of hybrid approach for load balancing in cloud environment. It creates new candidate solutions (called agents) by combining the parent individual and several other individuals of the same population. These agents are moved around in the search-space by using mathematical formulae to combine the positions of existing agents from the population. If the new position of an agent is improved than it is accepted and forms part of the population, otherwise the new position is easily throw away. The series of act ion is repetition until achieve the results and by doing so it is hoped, but not guaranteed, that a satisfactory solution will eventually be discovered. This is a greedy selection scheme that often outperforms traditional EAs (Evolutionary Algorithms).

IV. RESULTS AND DISCUSSION

In this, CloudSim toolkit has been used to analyze the working of hybrid DEGA algorithm. Here are the graphs, which show the parameters.

- Makespan: (completion time of a schedule) It is defined as the time variation between the start and finish of a sequence of schedule. The makespan or completion time can be reduced with the help of hybrid algorithm, since the jobs have been executed with in the specified time interval by allocation of required resources using the Hybrid algorithm. Shown below are the results of DE, GA and hybrid approach DE-GA.

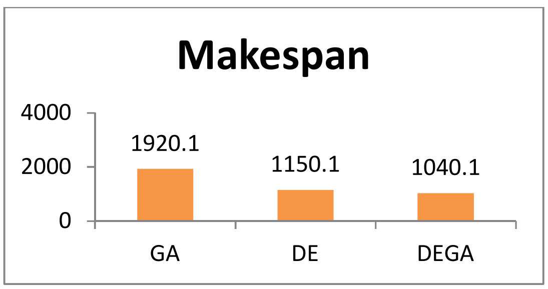

Fig. 2 Makespan for GA, DE and hybrid DE-GA

- Average response Time: Response time is the total amount of time it takes to respond to a request for service. That service can be anything from a memory fetch, to a disk IO, to a complex database query, or loading a full web page. Ignoring transmission time for a moment, the response time is the sum of the service time and wait time. Average RT has to be minimized as the approach must wait for minimum time for getting any resource. Following graph shows the results of average response time generated by implementing GA, DE and hybrid DE-GA.

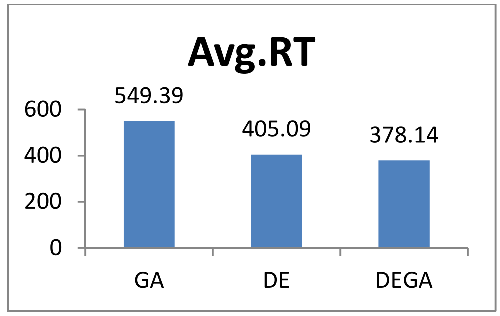

Fig. 3 Average Response Time (RT) for GA, DE and hybrid DE-GA

- Resource Utilization: This parameter defines how effectively the system utilizes the resources. The following graph depicts the resource(Virtual Machine(VM)) utilization for GA, DE and hybrid DE-GA.

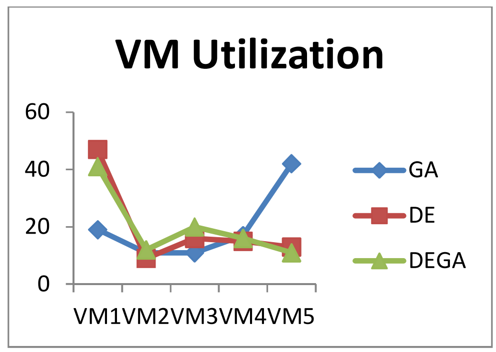

Fig. 4 Resource Utilization for GA, DE and hybrid DE-GA.

V. CONCLUSION

In this paper, Makespan, Average Response Time and Resource utilization are calculated for Genetic Algorithm, Differential Evolution and Hybrid DE-GA algorithm. The graphs, shown above, depict the results for the same and it is clear from the above results that hybrid DE-GA performs better than the two approaches i.e. GA and DE individually. In future, various other parameters such as energy efficiency, its reliability, throughput etc. can be calculated.

ACKNOWNLEGMENT

I would like to thank my guide, Dr. Sandip Kumar Goyal, who helped me a lot in this research work and supporting me. Also, I would like to thank our institution for providing the resources, so that the work can be done smoothly.

REFERENCES

1. *R.Buyya, R. Ranjan and R.N. Calheiros,"Modeling And Simulation of Scalable Cloud Computing Environments And The CloudSim Toolkit: Challenges And Opportunities", Proc. Of The 7th High Performance Computing And Simulation Conference (HPCS 09), IEEE Computer Society, June 2009.*

2. *N. Ajith Singh, M. Hemalatha, "An approach on semi distributed load balancing algorithm for cloud computing systems" International Journal of Computer Applications Vol-56 No.12 2012.*

3. *Price KV, Storn RM, Lampinen JA. Differential evolution: a practical approach to global optimization. Berlin: Springer-Verlag; 2005.*

4. *Storn R. System design by constraint adaptation and differential evolution. IEEE TransactionsonEvolutionaryComputation1999;3:22–34.*

5. *Pooja Mangla, Dr. Sandip Kr. Goyal, "Proposed Model for DE-GA based Load Balancing in Cloud Computing", International Journal of Engineering Development and Research, Vol.4, Issue 3, Sept, 2016.*