GRD Journals | Global Research and Development Journal for Engineering | International Conference on Innovations in Engineering and Technology (ICIET) - 2016 | July 2016
e-ISSN: 2455-5703

Energy Optimization in Cloud Computing by EGC Algorithm

[1]Mr. A. Gokul [2]Ms. E. Shanmuga Priya
[1,2]Department of Computer Science & Engineering
[1,2]Anna university Regional Campus Madurai, Madurai-625019, India

Abstract

Now a days, Cloud computing is an emerging technology, which is a new society for providing remote computing resources through a network. Cloud providers have many problems like getting an energy-efficiency control and satisfying a performance guarantee in cloud. In this paper, we implement three power-saving policies in cloud systems in order to reduce server idle power. To get optimize operational cost within a performance guarantee; we study the challenges of controlling service rates and applying the N-policy. Here we develop a cost function that includes the costs of power consumption, server startups and system congestion. Here we explain the operating modes, incurred costs and the effect of energy-efficiency controls on response times. To minimize cost within a response time under varying arrival rate by finding the optimal service rate and mode-switching restriction is our objectives. We propose an algorithm called Efficient Green Control (EGC) algorithm which is developed for solving optimization problems and making costs or performances tradeoffs in systems with different power-saving policies. By applying the power-saving policies combined with the proposed algorithm, we get our result that the benefits of reducing operational costs and improving response time.
Keyword- Cloud Computing, Energy Efficient, Power Saving, Response Time

---

I. INTRODUCTION

The cloud computing model is comprised of a front end and a back end. These two elements are connected through a network. The front end is the vehicle by which the user interacts with the system and the back end is the cloud itself. The front end is composed of a client computer, or the computer network of an enterprise, and the applications used to access the cloud. The back end provides the applications, computers, servers, and data storage that creates the cloud of services.

Users can pay for a service and access the resources made available during their subscriptions until the subscribed periods expire. While subscribing IaaS resources, the web service operators aimed to provide a service level Agreement (SLA) with their clients, e.g.,a guarantee on request response time. The resource provisioning of IaaS allows consumers to elastically increase or decrease the system capacity by changing configurations of computing resources.

A. Characteristics

The characteristics of cloud computing include on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service. On-demand self-service means that customers (usually organizations) can request and manage their own computing resources.

B. Benefits of Cloud Computing

- Reduce spending on technology infrastructure. Maintain easy access to your information with minimal upfront spending. Pay as you go (weekly, quarterly or yearly), based on demand.
- Streamline processes. Get more work done in less time with less people.
- Reduce capital costs. There's no need to spend big money on hardware, software or licensing fees.

Minimize licensing new software. Stretch and grow without the need to buy expensive software licenses or programs.

II. EGC ALGORITHM DESCRIPTION

This algorithm uses the three power saving polices such as ISN policy, SI policy, SN policy for the cost minimization process. Here the jobs are in the waiting queue which allows jobs in the server by using the FCFS and LCFS scheduling process.
In the ISN Policy, A server ends its busy mode when all current job requests have been finished. When the server stays in an idle mode and waits for subsequently arriving jobs before switching into a sleep mode. If a job arrives during an idle period, a server can switch into a busy mode and start to work immediately. A server begins a next idle period until all job requests have been successfully completed. If there has no job arrival, a server switches into a sleep mode when an idle period expires. A server

remains in a sleep mode if the number of jobs in the queue is fewer than the controlled N value. Otherwise, a server switches into a busy mode and begins to work.

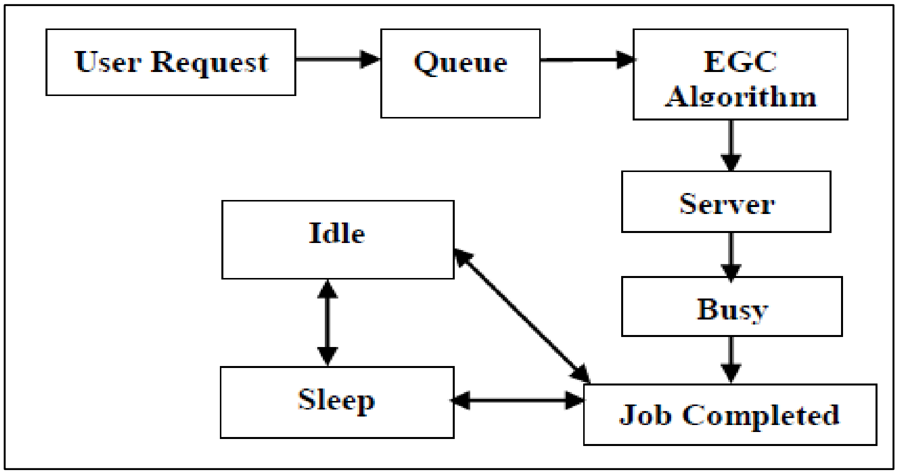

Fig. 1: System Architecture

III. USING CLOUDSIM SOFTWARE

A. CloudSim

CloudSim is a famous simulator for cloud parameters developed in the CLOUDS Laboratory, at the Computer Science and Software Engineering Department of the University of Melbourne. The CloudSim library is used for the following operations: Large scale cloud computing at data centers.

B. Downloading, Installing and Integrating Cloudsim

CloudSim is free and open source software. It is a code library based on Java. This library can be directly used by integrating with the JDK to compile and execute the code. Using Eclipse or NetBeans IDE, the CloudSim library can be accessed and the cloud algorithm implemented.

IV. MODULE DESCRIPTION

– Describing states
– SLA Constraints
– EGC Algorithm

A. Describing States

It is the first module in our process. A busy mode is defined as the mode which indicates that jobs are processed by a server running in one or more of its VMs'; and an idle mode is a mode which indicates that a server remains active but no job is being processed at that time. Here in order to mitigate or eliminate idle power wasted, three power-saving policies are used and it involves different energy efficient controls, decision processes and operating modes. First, we try to make an energy-efficient control in a system with three operating modes called m= {Busy, Idle, Sleep}, where a sleep mode would be responsible for saving power consumption. A server is allowed to stay in an idle mode for a short time when there has no job in the system, rather than switch abruptly into a sleep mode right away when the system becomes empty. An idle mode is the only operating mode that connects to a sleep mode. A server doesn't end its sleep mode even if a job has arrived; it begins to work only by satisfying the constraints that the number of jobs in a queue is more than the controlled N value

B. SLA Constraints

It is the second module in our process. A service level agreement (SLA) due to the fact that shutting down servers may sacrifice quality of service (QoS). The SLA is known as an agreement in which QoS is a critical part of negotiation. A penalty is given when a cloud provider violates performance guarantees in a SLA contract. In short, reducing power consumption in a cloud system has raised several concerns, without violating the SLA constraint or causing additional power consumption are both important. The SLA constraint is focused on the response time guarantee by considering both the queuing delay and the job execution time.

C. EGC Algorithm

It is the final module in the process. Here the Efficient Green Control Algorithm is applied for the cost optimization process. In the waiting queue, based on the job arrival, the server runs the job, when all jobs are comes at the same time, then there is a collision occurs on running the first job. So in order to reduce the collision, we use the scheduling algorithm called FCFS and LCFS process.

Here this algorithm uses the policies such as ISN policy, SI policy, SN policy for cost savings and improvement on the response time can be verified. Finally the Graph is plotted to show our result.

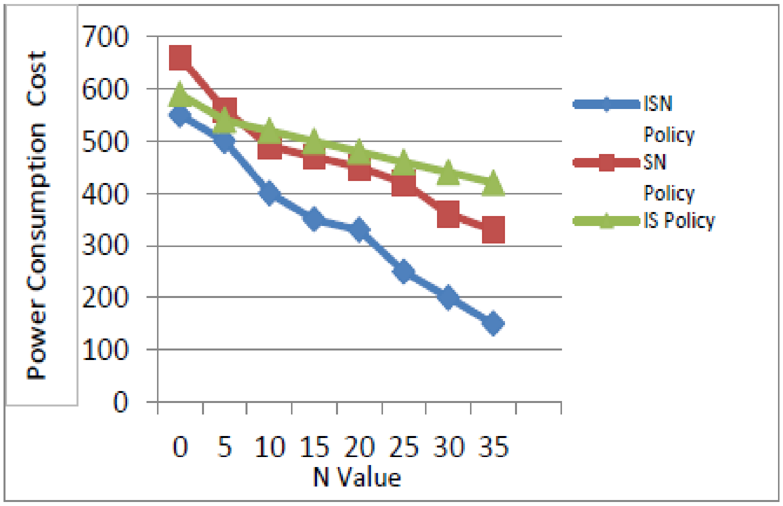

Fig. 2: Power Consumption Cost under Various N Values

V. CONCLUSION

Cloud Computing is the emerging technology but the growing crisis in power shortages has brought a concern in existing and future cloud system designs. Here to mitigate unnecessary idle power consumption, three power saving policies with different decision processes and mode switching controls are considered. Our algorithm allows for cloud providers to optimize the problem in decision-making on service rate and mode-switching restriction, so as to minimize the operational cost without sacrificing a SLA constraint. The issues such as choosing a suitable policy among diverse power managements which reaches a relatively high effectiveness has been examined based on the variations of arrival rates and incurred costs. Experimental results show that a system with the SI policy can significantly improve the response time in a low arrival rate situation.

REFERENCES

[1] G. Wang and T. E. Ng, "The impact of virtualization on network performance of amazon ec2 data center," in Proc. IEEE Proc. INFOCOM, 2010, pp. 1–9.
[2] R. Ranjan, L. Zhao, X. Wu, A. Liu, A. Quiroz, and M. Parashar, "Peer-to-peer cloud provisioning: Service discovery and load-balancing,"in Cloud Computing. London, U.K.: Springer, 2010, pp. 195–217.
[3] R. N. Calheiros, R. Ranjan, and R. Buyya, "Virtual machine provisioning based on analytical performance and QoS in cloud computing environments," in Proc. Int. Conf. Parallel Process., 2011, pp. 295–304.
[4] Server virtualization has stalled, despite the hype [Online]. Available: http://www.infoworld.com/print/146901, 2010.
[5] Y. C. Lee and A. Y. Zomaya, "Energy efficient utilization of resources in cloud computing systems," J. Supercomput., vol. 60, no. 2, pp. 268–280, 2012.
[6] Beloglazov, R. Buyya, Y. C. Lee, and A. Zomaya, "A taxonomy and survey of energy-efficient data centers and cloud computing systems," Adv. Comput., vol. 82, pp. 47–111, 2011.
[7] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Comput. Syst., vol. 25, no. 6, pp. 599–616, 2009.
[8] L. Wang, G. Von Laszewski, A. Younge, X. He, M. Kunze, J. Tao, and C. Fu, "Cloud computing: A perspective study," New Generation Comput., vol. 28, no. 2, pp. 137–146, 2010.
[9] R. Ranjan, R. Buyya, and M. Parashar, "Special section on autonomic cloud computing: Technologies, services, and applications," Concurrency Comput.: Practice Exp., vol. 24, no. 9, pp. 935–937, 2012.
[10] M. Yadin and P. Naor, "Queueing systems with a removable service station," Operations Res., vol . 14, pp. 393–405, 1963.