UNIVERSITI PUTRA MALAYSIA

DYNAMIC LOAD BALANCING ALGORITHM BASED ON DEADLINE CONSTRAINED IN CLOUD ENVIRONMENT

MUZZAMMIL MANSUR

FSKTM 2019 36

DYNAMIC LOAD BALANCING ALGORITHM BASED ON DEADLINE CONSTRAINED IN CLOUD ENVIRONMENT

By

MUZZAMMIL MANSUR

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfillment of the Requirements for the Degree of Master of Computer Science

June 2019

DEDICATION

This research is dedicated to my father Alhaji Mansur Muhammad Yahaya, my late mother Malama Hauwa'u Musa and family for their love, endless patience and support throughout my life.

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfilment of the requirement for the degree of Master of Computer Science

DYNAMIC LOAD BALANCING ALGORITHM BASED ON DEADLINE CONSTRAINED IN CLOUD ENVIRONMENT

By

MUZZAMMIL MANSUR

June 2019

Supervisor : AP Dr. Rohaya Binti Latip
Faculty : Computer Science and Information Technology

The Performance of cloud depends on the task scheduling as well as load balancing. Cloud Service Provider (CRP) provides services on demand to the users, as the application as well as the numbers of users are gradually growing over the cloud environment which leads to the increasing in the workload that are deployed over the virtual machine (VM). Due to the growing in traffic as well as workload, there is need for the cloud resource broker to have effective as well as efficient algorithm that disseminate task properly within the entire running VM, also decreases the rejection ratio of the task. This research work implements a scheduling algorithm which balances the workload among the whole VM using last K interval and to measure makespan time as well as number task that meet their respective deadline when the resources and task rejection is increasing. An experiment was carried out using Cloudsim Simulator. The results show that makespan time was reduce and improves the ratio of task that will meeting to their respective deadline when compared with the

First Come First Serve (FCFS), Dynamic Min-Min, and Shortest Job First (SJF) algorithm.

Abstrak tesis dikemukakan kepada senat University Putra Malaysia sebagai memenuhi keperluan untuk ijazah untuk Master Sains Komputer

ALGORITMA BALIK DINAMIK BERDASARKAN BERDASARKAN DEADLINE YANG DILAKUKAN DALAM ALAM SEKITAR

Oleh

MUZZAMMIL MANSUR

June 2019

Pengerusi : AP Dr. Rohaya Binti Latip

Fakulti : Sains Komputer dan Teknologi Maklumat

Prestasi awan tiba-tiba bergantung kepada penjadualan tugas serta pengimbangan beban. Pembekal perkhidmatan awan (CRB) menyediakan perkhidmatan yang diminta kepada pengguna, kerana aplikasi serta kuantiti pengguna secara beransur-ansur tumbuh di atas persekitaran awan yang mengakibatkan peningkatan lalu lintas dan beban kerja yang dikerahkan melalui mesin maya (VM). Kerana semakin meningkatnya trafik serta beban kerja, terdapat keperluan bagi sumber daya awan untuk mempunyai algoritma yang berkesan dan efisien yang menyebarkan tugas dengan betul dalam keseluruhan VM yang berjalan, juga mengurangkan nisbah penolakan tugas. Kerja projek ini melaksanakan algoritma penjadualan yang mengimbangi beban kerja di kalangan keseluruhan VM menggunakan selang K terakhir dan untuk mengukur masa Makespan serta tugas nombor yang memenuhi tarikh akhir masing-masing apabila sumber dan penolakan tugas semakin meningkat. Percubaan dilakukan menggunakan Simulator Cloudsim. Hasilnya menunjukkan bahawa masa Makespan telah mengurangkan dan meningkatkan nisbah tugas yang

akan bertemu dengan tarikh akhir masing-masing apabila dibandingkan dengan algoritma FCFS, dinamik Min-Min, dan SJF.

ACKNOWLEDGEMENTS

To Almighty Allah (SWT), I am thankful for the blessings and virtues, and for reconciling, strength, patience, courage, and determination he gave me to complete this work to the fullest. Alhamdulillah.

I would like to express my sincere gratitude and appreciation to my humble supervisor in person of, **AP Dr. Rohaya Binti Latip** for her enermost continous support, advice and intersest. Her guidance has helped me alot throughout my research and wriinf of this thesis. I would also like to thank mu also humble assessor, **AP Dr. Nor Asilla Wati Abdul Hamid** for her encouragement and interest in the course of this research work.

Finally, I must extend my sincere thanks to the TETFUND, especially Waziri Umaru Federal Polytechnic Birnin Kebbi, Kebbi State, Nigeria for their support by sponsoring me in my study. Nevertheless, my gratitude to the Malaysian people in general for their perfect hospitability in their green land during my study period.

APPROVAL

This thesis was submitted to the Faculty of Computer Science and Information Technology of Universiti Putra Malaysia and has been accepted as partial fulfillment of the requirement for the award of degree of Master of Computer Science.

The members of the Supervisory Committee were as follows:

Supervisor: Assoc. Prof. Dr. Rohaya Latip
Department of Communication Technology and Network
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia

Date and Signature: _____

Assessor: Assoc. Prof Dr. Nor Asilla Wati Abdul Hamid
Department of Communication Technology and Network
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia

Date and Signature: _____

DECLARATION

I declare that the thesis is my original work except for quotation and citations which have been duly acknowledged. I also declare that it has not been previously, and is not concurrently, submitted for any other degree at Universiti Putra Malaysia or other institution.

Signature: _____ Date: _____

Name and Matric No: Muzzammil Mansur GS51578

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

VM	Virtual machine
OVM	Overloaded virtual machine
UVM	Underloaded virtual machine
CSP	Cloud service provider
CRP	Cloud resource provisioner
CRB	Cloud resource broker
MST	Makespan time
SLA	Service level agreement
FcFs	First come first serve
SJF	Shortest job first
CPU	Central processing unit
MI	Mili Instructions
NP	Optimization problem
IaaS	Infrastructure as a service
PaaS	Platform as a service
SaaS	Software as a service
QoS	Quality of Service
API	Application program interface
BS	Balanced spiral
ACO	Ant colony optimization algorithm
PSO	Particle swarm optimization
EA	Evolution algorithm
AD	Adaptive load balancing
DTSLB	Dynamic task scheduler and load balancer
CLRI	Cloud load and resource information
ERPD	Elastic resource provisioning as well as deprovisioning

CHAPTER 1

INTRODUCTION

1.1 Background

In the recent years cloud computing have become developing technology and also a raised area for present distributed computing environment that simplifies the computing resources over the internet to the user (Adhikari & Amgoth, 2018). It also provides services on demand in a way of hardware infrastructures as well as software application based upon pay per use on the internet such as platform as a service(PaaS), Infrastructure as a service IaaS), storage as a service(SaaS), software as a service(SaaS), and etc. The type of services that are provided are mostly valuable in a business, industrial as well as scientific applications. Many users demand for resources by sending a request to the cloud service providers (CSP) while CSP choose the best resource within user given budget and deadline. In which the cloud service provider delivers the service demanded to the users. Several application as well as the users are gradually increase over the cloud environment therefore there is an increase of the workload as well as traffic within the web application which are deployed over the VM. Due the rise in the workload as well traffic within the deployed VM cloud resource broker need to have an effective algorithm that can allocates task properly in whole the running VM as well as minimizes rejection ratio of the task in order to make sure that the whole user task is executed (Kumar & Sharma, 2018).

The primary responsibility of any load balancing is to make use of resource available on the cloud in such a way that enhances response time, as well as scalability of the application. An efficient load balancing not only rises performance of the system in

such way that user can get feedback within a smallest time but also produce least makespan time and prevent system bottleneck which may likely happen because of the imbalance of the load. Conversely load balancing is one of the thought-provoking study area or research area within the field of cloud computing other challenges area may include communication delay, data loss, security and heterogeneity. The main aim of the load balancing is to balances the load between the VM.

Traditionally two steps can be followed in other to achieved load balancing within the cloud environment namely, task scheduling as well as monitoring the Virtual machine (Kumar & Sharma, 2017). Task scheduling, in computer science is among the best famous optimization problems (NP Complete), cloud environment contain heterogeneous resource in which many host which are different I one way or the other as well as many dissimilar VM configuration and also on demand request keep on changing very rapidly. Consequently, it is hardly to ascertain and also compute the whole likely task resource mapping on the cloud environment. Therefore, there is need for efficient as well as effective task scheduling algorithm that can disseminate task in an efficient way in other to have a minimum number of VM that can faces overloaded as well as under loaded condition. The second steps of achieving the load balancing is to monitor the VM constantly and carry out load balancing operation using one of the two technique which are Virtual machine migration or task migration technique. Consequently, task migration technique has a lot of benefits compared to the VM migration, due to the benefit of task migration technique over it counterpart we are going to use it in this research work. Furthermore CRB observe the VM constantly within cloud environment, so that if their exist a VM that is within under loaded or over loaded state when task scheduling has already happened then CRB begin load

balancing on the VM and also move the task that is in the overloaded VM to under loaded VM.

In this research work, have investigated the makespan time as well as number of tasks that are able to meet the deadline. In other to achieve this, we considered some part of scalability which simply refers to capability of a system to cough with a problem whenever there is an increase in the scope of the problem such as the size of the request differ randomly, increasing in the number of request etc. Two types of scalability exist which are Vertical scalability referred to as "scale up" as well as horizontal scalability which known as "scale out". Scale up as one of the scalability is achieved through making modifications on the already exist resources such hard drivers, CPU's, memory, scale up is not commonly used in cloud environment due to the fact that most well-known operating systems do not sustenance these modifications without restarting on existing resource such as memory and CPU. Scale out simply refers to discharging or adding of one or more computing node or machine instance of same type. In cloud environment horizontal scaling (scale out) is advantageous than it counterpart vertical scaling (scale up) since it is not limited by hardware capacity and also less expensive (Kumar, et al, 2018).

The aptitude of auto scaling with respect to on upcoming requests in cloud computing serves as the major benefits not only to the users but also to the service providers (Hwang, Shi & Bai, 2014). Auto scaling has an advantage of minimizing the risk which is mostly related with demands and excess of load that can causes server failure. Two kind of auto scaling approaches are available within cloud environment which are proactive as well as reactive approach. A proactive scaling approach allows the

service providers to plan vigorous modifications in the capacity that should be match with likely changes within the application request. Proactive scaling, one need to initially understand the expected modifications in the workload what modification on the workload are likely to happen within anticipated workload. Proactive approaches have some limitation which is whenever the predictive model fails therefore the resources will likely be underutilization as well as overutilization and the defined service level agreement will be violated. While Reactive approach provides their reply to an event when the event has already happened this type of approaches is mostly good for services that have short term, it major drawbacks is very expensive for services that have long term. Is among the advantage of proactive approaches that its attempts to remove the problem in advance of occurrence (Kumar, et al, 2018). Proactive scaling approaches have been divided into 3 categories, such as event, cyclic and prediction based. In this research work has make use of prediction-based method which can ascertain the upcoming demand based upon the history.

1.2 Problem Statements

Cloud Service Provider (CSP) offers services to the users on demand, as the number of users and applications in the cloud environment continually grow at a rapid rate regarding the workload deployed over the virtual machine (VM), a serious challenge is faced. The challenge is increase in workload leading to most of the tasks missed their deadline and have very high makespan time. Subsequently, different algorithms were used by the Cloud Resource Broker CRB (like FCFS, SJF, Min-Min etc.) to overcome the problem but unfortunately the problem persist due to increase in users and workload that are deployed. Consequently, the Cloud Resource Broker is in dare need for an efficient and effective algorithm capable of handling the distribution of

the tasks properly in the running VMs to increase the ratio of task meeting the deadline and minimize the makes span time.

1.3 Research Objective

The objective of this research work is to re-implement a Dynamic Load Balancing algorithm developed using the concept of last K-interval. The Algorithm balances the workload among the VMs, minimizes makespan time and increases the ratio of task meeting the deadline.

1.4 Research Scope

The scope of this research is to re-implement a dynamic load balancing algorithm based on that last k-interval using Cloudsim simulator with a datacenter having two host and 10 virtual machine and the result will be compared only with, First Come-First-Serve, Min-Min and Shortest-Job-First algorithms.

1.5 Organization of the Thesis

The project is written based on the standard structure of University Putra Malaysia to cover how the project research is accomplished, more so the remainder of the project is organized as follows: In Chapter 2, an extensive literature review of load balancing in cloud environment together with the taxonomy of comparison has been presented. Conversely, journals, conference papers, conference proceedings, thesis, seminars, books as well as online resources are used in other to enrich chapter and to serve as the main references.

In Chapter 3, the methodology adopted in this research has been presented. The flowchart of the load balancing among VM has been introduced and also many notations adapted were clearly defined as well as the hardware and software resources used has been presented.

In Chapter 4, the implementation and result discussion were thoroughly analyzed and has been presented in a manner that is easily understood.

In Chapter 5, Conclusion of the entire research as well as future work has been presented

REFERENCES

Abrishami, S., & Naghibzadeh, M. (2012). Deadline-constrained workflow scheduling in software as a service cloud. *Scientia Iranica*, vol. *19*, pp. 680-689.

Adhikari, M., & Amgoth, T. (2018). Heuristic-based load-balancing algorithm for IaaS cloud. *Future Generation Computer Systems*, vol. *81*, pp. 156-165.

Chen, H., Wang, F., Helian, N., & Akanmu, G. (2013, February). User-priority guided Min-Min scheduling algorithm for load balancing in cloud computing. In *Parallel computing technologies (PARCOMPTECH), 2013 national conference on* (pp. 1-8). IEEE.

De Coninck, E., Verbelen, T., Vankeirsbilck, B., Bohez, S., Simoens, P., & Dhoedt, B. (2016). Dynamic auto-scaling and scheduling of deadline constrained service workloads on IaaS clouds. *Journal of Systems and Software*, vol. *118*, pp. 101-114.

Dubey, K., Kumar, M., & Chandra, M. A. (2015, March). A priority based job scheduling algorithm using IBA and EASY algorithm for cloud meta scheduler. In *Computer Engineering and Applications (ICACEA), 2015 International Conference on Advances in* (pp. 66-70). IEEE.

Fu, X., & Zhou, C. (2015). Virtual machine selection and placement for dynamic consolidation in Cloud computing environment. *Frontiers of Computer Science*, vol. *9*, pp. 322-330.

Ghobaei-Arani, M., Jabbehdari, S., & Pourmina, M. A. (2016). An autonomic approach for resource provisioning of cloud services. *Cluster Computing*, vol. *19*, pp. 1017-1036.

Hwang, K., Shi, Y., & Bai, X. (2014, December). Scale-out vs. scale-up techniques for cloud performance and productivity. In *Cloud Computing Technology and Science (CloudCom), 2014 IEEE 6th International Conference on* (pp. 763-768). IEEE.

Hu, T., Yi, P., Zhang, J., & Lan, J. (2018). Reliable and load balance-aware multi-controller deployment in SDN. *China Communications*, vol. *15*, pp. 184-198.

Kong, W., Lei, Y., & Ma, J. (2016). Virtual machine resource scheduling algorithm for cloud computing based on auction mechanism. *Optik-International Journal for Light and Electron Optics*, vol. *127*, pp 5099-5104.

Krishna, P. V. (2013). Honey bee behavior inspired load balancing of tasks in cloud computing environments. *Applied Soft Computing*, vol. *13*, pp 2292-2303.

Kumar, M., & Sharma, S. C. (2017). Dynamic load balancing algorithm for balancing the workload among virtual machine in cloud computing. *Procedia Computer Science*, vol. *115, pp.* 322-329.

Kumar, M., & Sharma, S. C. (2018). Deadline constrained based dynamic load balancing algorithm with elasticity in cloud environment. *Computers & Electrical Engineering*, vol. *69*, pp. 395-411.

Khan, M. N. I., & Islam, M. S. (2019). A New Approach of Energy Efficient Load Balancing for Wireless Sensor Networks. In *2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)* (pp. 350-353). IEEE

Li, W., & Shi, H. (2009, December). Dynamic load balancing algorithm based on FCFS. In *Innovative Computing, Information and Control (ICICIC), 2009 Fourth International Conference on* (pp. 1528-1531). IEEE.

Li, G., Wang, X., & Zhang, Z. (2019). SDN-Based Load Balancing Scheme for Multi-Controller Deployment. *IEEE Access*, vol. *7*, pp 39612-39622.

Lorido-Botrcn, T., Miguel-Alonso, J., & Lozano, J. A. (2013). Comparison of auto-scaling techniques for cloud environments.

Malawski, M., Juve, G., Deelman, E., & Nabrzyski, J. (2015). Algorithms for cost- and deadline-constrained provisioning for scientific workflow ensembles in IaaS clouds. *Future Generation Computer Systems*, vol. *48*, pp. 1-18.

Madni, S. H. H., Latiff, M. S. A., & Ali, J. (2018). Hybrid gradient descent cuckoo search (HGDCS) algorithm for resource scheduling in IaaS cloud computing environment. *Cluster Computing*, vol. 62, pp1-34.

Mondal, R. K., Nandi, E., & Sarddar, D. (2015). Load balancing scheduling with shortest load first. *International Journal of Grid and Distributed Computing*, vol. *8*, 171-178.

Naha, R. K., & Othman, M. (2014). Brokering and load-balancing mechanism in the cloud–revisited. *IETE Technical Review*, vol. *31*, pp. 271-276.

Nayak, S. C., & Tripathy, C. (2016). Deadline sensitive lease scheduling in cloud computing environment using AHP. *Journal of King Saud University-Computer and Information Sciences*. Vol.7 pp 54-65.

Pacini, E., Mateos, C., & Garino, C. G. (2015). Balancing throughput and response time in online scientific Clouds via Ant Colony Optimization (SP2013/2013/00006). *Advances in Engineering Software*, vol. *84*, pp. 31-47.

Ramezani, F., Lu, J., & Hussain, F. K. (2014). Task-based system load balancing in cloud computing using particle swarm optimization. *International journal of parallel programming*, vol. *42*, pp. 739-754.

Sahoo, B., Kumar, D., & Jena, S. K. (2013). Analysing the impact of heterogeneity with greedy resource allocation algorithms for dynamic load balancing in heterogeneous distributed computing system.

Somasundaram, T. S., Govindarajan, K., Rajagopalan, M. R., & Rao, S. M. (2013). A broker based architecture for adaptive load balancing and elastic resource provisioning and deprovisioning in multi-tenant based cloud environments. In *Proceedings of International Conference on Advances in Computing* (pp. 561-573).

Suresh, A., & Vijayakarthick, P. (2011, June). Improving scheduling of backfill algorithms using balanced spiral method for cloud metascheduler. In *Recent Trends in Information Technology (ICRTIT), 2011 International Conference on* (pp. 624-627). IEEE.

Tsai, J. T., Fang, J. C., & Chou, J. H. (2013). Optimized task scheduling and resource allocation on cloud computing environment using improved differential evolution algorithm. *Computers & Operations Research*, vol. *40*, pp. 3045-3055.