Middle-East Journal of Scientific Research 24 (Techniques and Algorithms in Emerging Technologies): 103-108, 2016
ISSN 1990-9233;

DOI: 10.5829/idosi.mejsr.2016.24.TAET23329

A Survival Study on Efficient File Sharing in Virtual Cloud Servers with Load Balancing and Virtualization Techniques

[1]C. Nelson Kennedy Babu and [2]V. Priya

[1]Professor, Computer Science and Engineering, C.M.S College of Engineering, India
[2]Assistant Professor, Information Technology, K.S.Rangasamy College of Technology, India

Abstract: Cloud data center comprises the network of heterogeneous commodity servers providing with virtualized computing services. Users access resources of data centers by allocating virtual machines (VMs) to hosts. A data center with multiple servers aggregated the performance of the data center optimized by load distribution and load balancing. The file sharing systems are globally scattered with different nodes to interconnect that realize collaborative file services. However, the initial allocation of VMs is not included in planning the load balancing heuristics. In addition it also takes higher execution time and decreases the throughput level. Our research work helps to share the files in cloud virtual servers with minimum resource scheduling time and to increase the throughput using load balancing techniques.

Key words: Cloud data center · Virtual Machines (VM) · Load balancing heuristics · Load distribution · Virtual servers and resource management

INTRODUCTION

Cloud computing is considered as the next dominant technology in IT industry. It presents an easy system preservation and scalable resource administration with Virtual Machines (VMs). Because of essential techniques of cloud computing, VM is the new topic. The high overhead of virtualization is solved through hardware development in CPU industry and with software execution enhancement in hypervisors. Cloud Computing model present the customers with virtual services of the quality which gathers the customer's needs. A cloud service operator is interested with the infrastructure in effective way while helping the customers. Cloud computing is a distributed internet based model for remote sharing, utilization of many resources and services. Load balancing concerns distribution of resources between the users or requests in consistent way in order that no node is congested or sitting idle. Resource scheduling is an essential problem of distribution and cluster calculation. Resource scheduling provides the user task execution effectiveness and the resources of the system numbers.

This paper is organized as follows: Section II discusses reviews on file sharing in virtual cloud servers, Section III describes the existing load balancing and resource scheduling methods, Section IV identifies the possible comparison between them, Section V explains the limitations as well as the related works and Section VI concludes the paper, key objective of the research is to share the files in cloud virtual servers with minimum resource scheduling time and to increase the throughput using load balancing techniques.

Literarture Review: Named Data Networking (NDN) based Virtual Machines migration protocol [1] was developed in cloud data center. However, migration policy could not avoid service interruptions. Also, the migrated virtual machines could not learn its IP address timely. Centralized Hierarchical Cloud-Based Multimedia System [2] was designed in cloud data system with a resource manager, cluster heads and server clusters. However, multimedia service task are not stopped in a single time step. To identify the static or dynamic partitioning in cloud data, Autonomous Agent Based Load Balancing Algorithm [3] was developed. A2LB mechanism includes three agents namely Load agent, Channel Agent and Migration Agent. However, the properties of collecting information moves from source to destination, but they do not necessarily come back to their source.

In [4], a novel locality-aware hashing scheme called NEST achieves locality-aware storage with Locality-sensitive hashing and load-balanced storage by

Corresponding Author: Dr. C. Nelson Kennedy Babu, Professor, Computer Science and Engineering, C.M.S College of Engineering

using the cuckoo driven method. However, NEST is not possible to use in real-world industrial applications. Qos-driven task scheduling algorithm [5] was used to deal in a market palace with more profit to the user in real-world applications. Scheduling algorithm is derived from Qos-driven user needs, user rights and development expectation in Cloud Computing. However, implementation of more factors is not improved for measuring in real time with many other algorithms. Distributed Problem Solving Techniques was illustrated in [6], to measure load balancing approach in many applications. Collaborative agents are provided by load balancing technique and an energy-aware consolidation protocol. However, devising load balancing heuristics for the initial allocation of VMs is not included. Also, resource usage profiles of virtual machines are not constructed.

File Sharing in Virtual Cloud Servers with Load Balancing and Virtualization Techniques: Cloud computing provides larger advantages through moving the computing infrastructure to Internet, minimizing the costs for the preservation and organization of hardware and software resources. Cloud is one form of parallel and distributed method with group of interconnected and virtualized computers. The computers are animatedly provisioned and designed as one or more combined computing resources derived from service-level agreements created through cooperation connecting the service provider and consumers. The consumers access applications and data of Cloud from anyplace at any time. It is not easy for the cloud service providers to assign the cloud resources dynamically and capably. With the centralized organization of computing resources, cloud computing distributes the hosted services over the Internet where the access to shared hardware, software, databases, information and all resources are presented to user's on-demand. In a data center with many servers, the aggregated results of the data center are enhanced by load distribution and balancing. Cloud-based applications are based on load balancing and optimization than conventional enterprise functions. For end users, load balancing capabilities are taken when they select cloud computing provider. For cloud providers, load balancing abilities are the source of revenue connected with the service quality. Consequently, an efficient load balancing plan is a key component for constructing any cloud computing design.

Load balancing in cloud computing atmosphere is in requirement of maximum resource consumption, maximum throughput, lesser response time, dynamic resource scheduling with scalability and consistency. An Autonomous Agent based Load Balancing algorithm (A2LB) is designed to address the requirements. When a VM gets overloaded, the service provider distributes the resources where the existing resources are used in an accurate way and load at all virtual machines stay balanced. From Figure 1, A2LB mechanism has three agents. They are: Load agent, Channel Agent and Migration Agent. Load and channel agents are stationary agents and migration agent is an ant with a particular category of mobile agents. The reason

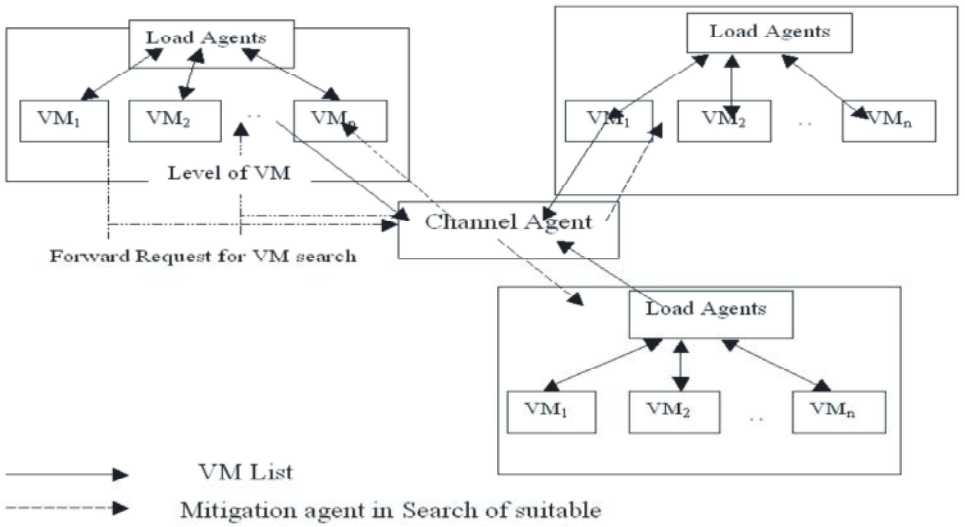

Fig. 1: High level view of A2LB

behind organizing ants is their capability to select shortest/best path to destination. Ant agents are motivated from biological ants to find path from their colonies to the food source.

Load Balancing in Cloud Environment: Load balancing in clouds is a method that distributes the excess load evenly across many servers. The technique is used to reach high user satisfaction and resource utilization to enhance the performance. Load balancing is the key issue in Cloud computing that allocates the dynamic workload across many nodes to guarantee that no single resource is overwhelmed or underdeveloped. It is taken as an optimization issue and a good load balancer change its strategy to the varying environment and the types of tasks.

Load Balancing allocates distribution of workload across one or more servers, data centers, hard drives or additional computing resources with Cloud Service Providers (CSP) a mechanism to allocate application requests across any number of application uses in data centers. A centralized hierarchical CMS comprises resource manager and number of server clusters managed through cluster head and the servers in several server clusters to present many services. Every time CMS collects client's demands for multimedia service tasks, the resource manager of CMS allocates the task requests to many server clusters consistent with the features of the requested tasks. The cluster head of all server clusters distributes the allocated task to many servers inside the server cluster. It is not hard to examine that the load of each server cluster considerably changes the results of the CMS. The resource manager of CMS is in pursuit of comparatively distributing the task load across server clusters and it is capable to manage load balancing in the CMS.

Resource Scheduling: Scheduling is difficult tasks in a cloud computing environment where many alternative computers with varying capacities are available to allocate the computing resources. Efficient task scheduling mechanism gathers user's needs and enhances the resource use. The cloud service providers collect many computing requests with dissimilar needs and preferences from users at the same time. The tasks are performed at lesser cost and less computing resources when tasks call for higher computing capability. When the cloud computing service providers obtain the tasks from users, the tasks are evaluated pair wise with the matrix methods. The computing resource or storage resource in a cloud computing environment is allocated to the related task consistent with the weight of each task once.

Anew locality-aware balanced data structure called NEST is designed in cloud servers. NEST attains locality-aware storage with LSH and load-balanced storage by cuckoo driven technique to move crowded items to different empty positions. NEST reduces the continuous loop load in cuckoo hashing through allocating new items in neighboring buckets allocated in LSH. The designed structure presents a locality-aware data management in tri-tiered heterogeneous storage hierarchy. The top-layer DRAM as temporary buffer that recognizes and aggregates correlated files with help of LSH with a difficulty of O(1). To remove the hash collisions in hash table, a variant of the cuckoo hash and attain a constant-scale lookup difficulty. The middle-layer SSD stores the metadata in using the key-value pairs. A key is the hashed value of a file ID and the value is the metadata of that file. Correlated files are conducive to sequential operations with a high probability. The bottom-layer hard disk stores and preserves the correlated files. NEST effectively uses the locality of datasets to maintain sequential operations and data retrieval.

Performance Analysis Offile Sharing in Virtual Cloud Servers: In order to compare the file sharing in virtual cloud servers using load balancing and resource scheduling techniques, number of virtual cloud server taken to perform the experiment.

Resource Scheduling Time: Resource scheduling time is defined as the time required for scheduling the tasks of the virtual cloud servers. It is measured in terms of milliseconds (ms).

Resource Scheduling Time comparison takes place on existing Autonomous Agent based Load Balancing algorithm (A2LB), Centralized Hierarchical Cloud-Based Multimedia System (CMS) and NEST.

Table 4.1: Tabulation of Resource Scheduling Time for File Sharing in Cloud

Number of virtual cloud servers (Number)	Resource Scheduling Time (ms)		
	A2LB Algorithm	CMS Mechanism	NEST
10	42	26	51
20	48	29	54
30	53	34	59
40	58	38	63
50	61	42	66
60	65	46	70
70	69	52	74

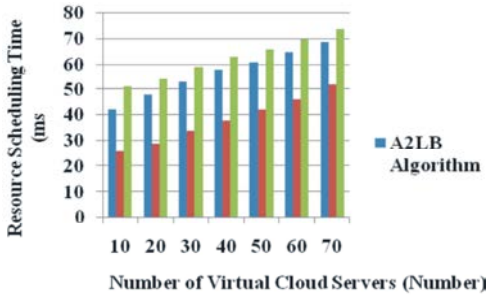

Fig. 4.1: Tabulation of Resource Scheduling Time for File Sharing in Cloud

From Figure 4.1, resource scheduling timeof existing techniques is evaluated. Centralized Hierarchical Cloud-Based Multimedia System (CMS) schedules the task with lesser time than Autonomous Agent based Load Balancing algorithm (A2LB) and NEST. Research in Centralized Hierarchical Cloud-Based Multimedia System (CMS) has 50.68% lesser resource scheduling time than Autonomous Agent based Load Balancing algorithm (A2LB) and 67.61% lesser scheduling time than NEST.

Memory Utilization for File Sharing: Memory utilization for file sharing is defined as the amount of memory space required for sharing the files in the virtual cloud servers. It is represented as megabytes (MB).

Resource Scheduling Time comparison takes place on existing Autonomous Agent based Load Balancing algorithm (A2LB), Centralized Hierarchical Cloud-Based Multimedia System (CMS) and NEST.

From Figure 4.2, memory utilization for file sharing of existing techniques is evaluated. Autonomous Agent based Load Balancing algorithm (A2LB) consumes less memory space than Centralized Hierarchical Cloud-Based Multimedia System (CMS)and NEST. Research in Autonomous Agent based Load Balancing algorithm (A2LB) consumes 56.77% lesser memory space than Centralized Hierarchical Cloud-Based Multimedia System (CMS) and 23.72% lesser memory space than NEST.

Table 4.2: Tabulation of Memory Utilization for File Sharing in Cloud

Number of virtual cloud servers (Number)	Memory Utilization for File Sharing (MB)		
	A2LB Algorithm	CMS Mechanism	NEST
10	15	36	25
20	19	39	29
30	23	43	34
40	25	48	39
50	29	52	43
60	34	58	48
70	38	62	53

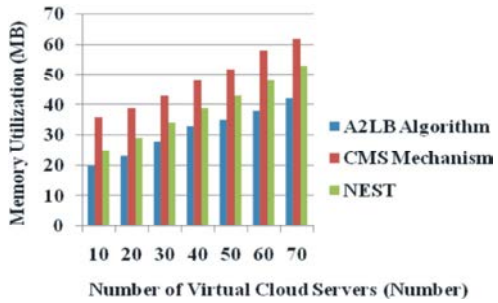

Fig. 4.2: Memory Utilization for File Sharing in Cloud

Throughput: Throughput is defined as the average rate of successful sharing of file in the virtual clouds. It is measured in terms of percentage (%).

Throughput comparison takes place on existing Autonomous Agent based Load Balancing algorithm (A2LB), Centralized Hierarchical Cloud-Based Multimedia System (CMS) and NEST. From Figure 4.3, throughput of existing techniques is evaluated. NEST has comparatively higher throughput than Autonomous Agent based Load Balancing algorithm (A2LB) and Centralized Hierarchical Cloud-Based Multimedia System (CMS). Research in NEST has15.123% higher throughput than Centralized Hierarchical Cloud-Based Multimedia System (CMS) and 24.41% higher throughput than Autonomous Agent based Load Balancing algorithm (A2LB).

Table 4.3: Tabulation of Throughput for File Sharing in Cloud

Number of virtual cloud servers (Number)	Throughput (%)		
	A2LB Algorithm	CMS Mechanism	NEST
10	46	52	65
20	49	56	69
30	53	61	73
40	58	66	76
50	62	69	80
60	65	73	83
70	71	76	86

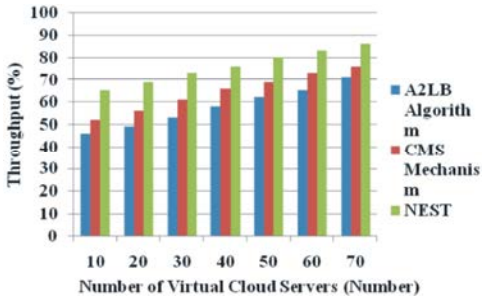

Fig. 4.3: Throughputs for File Sharing in Cloud

Discussion on Limitation of File Sharing in Virtual Cloud Servers: In A2LB mechanism, the service provider distributes the resources when virtual machines become overloaded. A2LB algorithm presents dynamic load balancing for cloud atmosphere. However, the properties of gathering information moves from source to destination, though not required returning to their source. NEST achieves locality-aware storage by using Locality-sensitive hashing (LSH) and load-balanced storage by using the cuckoo driven method. NEST is used to control cuckoo-driven locality-sensitive hashing to establish related items. The implementation of Cuckoo-driven method is used to obtain load-balancing buckets in hash tables. NEST reduces endless loop burden in cuckoo hashing through assigning the new items. However, NEST is not possible to use in real-world industrial applications.

CMS comprises the resource manager, cluster heads and server clusters where the resource manager assigns client's requests. Load balancing algorithm transmits multimedia data between server clusters and clients. The multimedia service tasks are allotted to server clusters consistent with the task features. An efficient load balancing algorithm increases the multimedia service task load on servers with less cost. Genetic algorithm efficiently manages dynamic multiservice load balancing in CMS. However, multimedia service task are not terminated inside a single time step.

Related Works: A Load balancing strategy using Genetic Algorithm (GA) was proposed in [7]. It was developed to provide an efficient utilization of resource in cloud environment. The algorithm is used to equal the load of cloud environment while reducing the duration of given tasks. However, GA algorithm cannot be applied for variation of the crossover and selection steps for getting efficient and tuned results. To balance the load between different computing virtual machines to obtain efficient result, SLA (Service-Level Agreement) architecture was developed in [8]. Service-Level Agreement is a fulfilled condition in scalable clouds with web-based load variability. A new scheduling policy handles the demanded cloud services derived from the SLA. However, virtual machine is not instant and there is a significant delay between the moments.

Multilevel caching scheme called MERCURY was planned in [9] to discover and develops data similarity and maintain efficient data placement. Multicore-enabled locality-sensitive hashing exactly collects the distinguished similarity across data. MERCURY minimizes cache conflicts and reduces the amounts of migrated data. However, locality-aware schemes are not implemented for efficient data placement in multiple memory controllers. To solve the Virtual Local Area Networks (VLANs) assignment problem in cloud data centers, a novel column generation approach [10] was designed. The use of VLANs provides scalable traffic management and the approach yields a substantial decrease in the size of the explored search space with encouraging optimality gap. However, the framework does not consider delay sensitive flows, multicast flows and dynamic flows.

A social network integrated peer-to-peer (P2P) file sharing system [11] was developed to enhance the file search with higher efficiency. The probabilities of proximity-close nodes are higher and it enhances efficiency and trustworthiness. However, social networks cannot assign weights to different factors to satisfy different user's requirements. A scalable Deduplication file system [12] was developed with low storage consumption and high-performance input and output which satisfy the requirements of virtual machine hosting for efficient data placement. However, special care is taken to attain high IO results.

Future Direction: The future direction of file sharing in virtual cloud servers carried out high throughput level and less scheduling time. Load balancing technique is used to distribute the loads equally in the cloud servers which fail in scheduling the resources exactly. For accurately scheduling the tasks or resources, resource scheduling method is carried out.

CONCLUSION

A comparison of different techniques for sharing the files in the virtual cloud servers are carried out. The initial allocation of VMs is not included in devising load balancing heuristics. Virtual machine images are not able to modify once it is written, because a few software application data's are read-only. NEST is not possible to use in real-world industrial applications. The multimedia service task may not be able to be finished within a single time step. The wide range of experiments on existing techniques calculates the comparative results of the various sharing techniques and its limitations. Finally from the limitation identified from the existing works, further research work can be carried out to achieve high throughput level and less scheduling time in virtual cloud servers.

REFERENCES

1. Ruitao Xie, Yonggang Wen, Xiaohua Jia and Haiyong Xie, 2015. "Supporting Seamless Virtual Machine Migration via Named Data Networking in Cloud Data Center", IEEE Transactions on Parallel and Distributed Systems, DOI 10.1109/TPDS.2014. 2377119.
2. Chun-Cheng Lin, Hui-Hsin Chin and Der-Jiunn Deng, 2014. "Dynamic Multiservice Load Balancing in Cloud-Based Multimedia System", IEEE Journal of Systems, 8(1), March 2014, pp: 225-234.
3. Aarti Singh, Dimple Junejab and Manisha Malhotra, 2015. "Autonomous Agent Based Load Balancing Algorithm in Cloud Computing", International Conference on Advanced Computing Technologies and Applications, Elsevier Procedia computer science, 45: 832-841.
4. Yu Hua, Bin Xiao, Xue Liu and Dan Feng, 2015. "The Design and Implementations of Locality-Aware Approximate Queries in Hybrid Storage Systems", IEEE Transactions on Parallel and Distributed Systems, 26(11), November 2015, pp: 3194-3207.
5. Nidhi Bansal, Amitab Maurya, Tarun Kumar, Manzeet Singh and Shruti Bansal, 2015. "Cost performance of QoS Driven task scheduling in cloud Computing", Third International Conference on Recent Trends in Computing (ICRTC 2015), Elsevier Procedia computer science, 57: 126-130.
6. Octavio Gutierrez-Garcia, J. and Adrian Ramirez-Nafarrate, 2015. "Collaborative Agents for Distributed Load Management in Cloud Data Centers using Live Migration of Virtual Machines", IEEE Transactions on Services Computing, 8(6): 916-929.
7. Kousik Dasgupta, Brototi Mandal, Paramartha Dutta, Jyotsna Kumar Mondal and Santanu Dam, 2013. "A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing", International Conference on Computational Intelligence: Modeling Techniques and Applications, Procedia Technology, Elsevier, 10: 340-347.
8. Jordi Vilaplana, 2014. Francesc Solsona, Jordi Mateo and Ivan Teixido, " SLA-Aware Load Balancing in a Web-Based Cloud System over Open Stack", Springer International Publishing, 8377: 281-293, Springer 2014.
9. Yu Hua, Xue Liu and Dan Feng, 2014. "Data Similarity-Aware Computation Infrastructure for the Cloud",IEEE Transactions on Computers, 63(1), January 2014, pp: 3-16.
10. Chadi Assi, Sara Ayoubi, Samir Sebbah and Khaled Shaban, 2014. "Towards Scalable Traffic Management in Cloud Data Centers", IEEE Transactions on Communications, 62(3), March 2014, pp: 1033-1045.
11. Guoxin Liu, Haiying Shen and Lee Ward, 2015. "An Efficient and Trustworthy P2P and Social Network Integrated File Sharing System", IEEE Transactions on Computers, 64(1), January 2015, pp: 54-70.
12. Xun Zhao, Yang Zhang, Yongwei Wu, Kang Chen, Jinlei Jiang and Keqin Li, 2014. Senior "Liquid: A Scalable Deduplication File System for Virtual Machine Images", IEEE Transactions on Parallel And Distributed Systems, 25(5), May 2014, pp: 1257-1266.