Cloud Management Systems and Virtual Desktop Infrastructure Load Balancing Algorithms - A Survey

Micheal Ernest Taylor and Jian Shen(✉)

School of Computer and Software,
Nanjing University of Information Science and Technology,
Nanjing 210044, China
delen007@live.com, s_shenjian@126.com

Abstract. Cloud Computing Technology has hatched some fascinating services, including hosted file servers, applications on demand and business continuity solutions. The volume of data storage upsurges quickly in open environment. So, load balancing is a main challenge in cloud environment. Load balancing helps to distribute the dynamic workload across multiple nodes to ensure that no single node is overloaded. It helps in proper utilization of resources. It also improve the performance of the system. Cloud data center management is an essential problem due to the numerous and heterogeneous strategies that can be applied, ranging from the virtual machine location to the coalition with other clouds. Load Balancing is important for essential operations in cloud virtual environments. Many algorithms have been developed for allocating client's requests to available remote nodes. This paper presents a survey on load balancing algorithms in cloud management systems and virtual desktop infrastructure as well as properties of cloud computing. The survey also presents a comparative metrics on load balancing algorithms such as round robin, honey bee, ant colony, and active clustering and others as main the focus.

Keywords: Cloud computing · Virtualization · Algorithm · Load balancing

1 Introduction

The model of running shared desktops in virtual machines hosted on servers is a gripping proposition. In distinction to traditional desktop management strategies, these virtual desktops are easily maintained, upgraded and updated, and use of a variety of devices in various locations to access sensitive data without ever leaving the confines of the data center is highly possible for a user. This form of server-hosted desktops is known as virtual desktop infrastructure (VDI). However, instigating VDI has historically been a complex responsibility usually reserved for large enterprises due to demanding requirements for high-end server and storage hardware [5].

X. Sun et al. (Eds.): ICCCS 2017, Part I, LNCS 10602, pp. 300–309, 2017.
https://doi.org/10.1007/978-3-319-68505-2_26

Cloud Computing Technology has hatched some fascinating services, including hosted file servers, applications on demand and business continuity solutions. Several vendors are now taking the technology to the next level, coalescing virtualization with cloud services [7]. A cloud-based VDI solution is a computing model where an end users system can access all the essential files and data virtually in spite of being alienated from the physical IT infrastructure. The VDI layer acts as an intermediary between backend and end-user application. Voluminous establishments use VDI technologies from vendors like Citrix and VMware to push virtual desktops out across their enterprises [7]. There are numerous existing issues in cloud computing. However, the key amongst them is load balancing. Load balancing is quite a new technique that accelerates networks and resources by providing a maximum throughput with minimum response time. Load balancing ensures efficient and fair distribution of computing resources to provide high satisfaction, better response time and utilization to users and prevent system bottlenecks which may occur due to load imbalance. Distributing the traffic between servers, data can be sent and received without high delay. Diverse types of algorithms are available that aids traffic loaded between available servers. Devoid of load balancing, users could experience delays, timeouts and possible long system responses. Load balancing solutions usually apply redundant servers which help a better distribution of the communication traffic so that the server availability is conclusively settled [6,9,11,14]. Due to this challenges of cloud computing its worthwhile to research in this area to improve on the existing load balancing algorithms and if possible an optimal solution provided. In this paper, a survey on related Load balancing algorithms in Cloud System and Virtual Desktop Infrastructure are presented.

The rest of the paper is organized as follows. In the second section, presents a literature review. The subsequent sections delineate in detail the architecture of Cloud Computing System, VDI, its pros and cons, and those of the platform that supports its services and methodology. This paper concludes with comments about the state of Cloud Computing and VDI and the likely component of the platform required for the support for business as a discussion of future work.

2 Related Works

The Cloud is made up of enormous resources. Management of these resources requires efficient planning. While planning an algorithm for resource provisioning on cloud the engineer should take into consideration the different existing cloud scenarios and must be aware of the issues that are to be resolved by the proposed algorithm. So, resource provisioning algorithm can be classified into different classes based on the environment, purpose and technique of proposed solution. Join-Idle Queue uses distributed dispatchers by first load balancing the idle processors across dispatchers and then assigning jobs to processors to reduce average queue length at each processor [8]. The disadvantage of this algorithm is that it is not scalable. Moharana [10] Proposed this load-balancing algorithm for dynamically scalable web services. It effectively reduces the system load, incurs no communication overhead at job arrivals and does not increase actual response time. It can perform close to optimal when used for web services.

Load balancing in cloud computing system [4] discussed on basic concepts of Cloud Computing and Load balancing and studied some existing load balancing algorithms, which can be applied to clouds. In addition to that, the closed-form solutions for minimum measurement and reporting time for single level tree networks with different load balancing strategies were also studied. The performance of these strategies with respect to the timing and the effect of link and measurement speed were studied. To maintain the load equilibrating in the cloud computing system, [1] proposed a scheduling algorithm. It combines the capabilities of both OLB (Opportunistic Load Balancing) [15] and LBMM (Load Balance Min-Min) [16] scheduling algorithms and are relatively more capable. Further, [1] gave an estimate to find the most beneficial cloud resource while regarding Cooperative Power aware Scheduled Load Balancing, a solution to the Cloud load balancing challenge.

In [2], the authors designed a load balancing algorithm based on round robin in Virtual Machine (VM) environment of cloud computing in order to achieve better response time and processing time. The load balancing algorithm is done before it reaches the processing servers the job is scheduled based on various parameters like processor speed and assigned load of Virtual Machine (VM) and etc. It maintains the information in each VM and numbers of request currently allocated to VM of the system. It identify the least loaded machine, when a request come to allocate and it identified the first one if there are more than one least loaded machine. In addition [12] projected a novel server-based load balancing policy for Internet servers that are distributed everywhere the world. The main contributions of this work are: (a) the evaluation of client-based server selection schemes in scenarios where several clients use the same schemes; and (b) the proposal of a new solution that outperforms existing ones by dynamically adapting the fraction of load each client submits to each server. In order to evaluate the solution, the author have implemented in a discrete event simulator framework using Java. The author has used the PackMime Internet traffic model [3] to generate HTTP traffic in the simulations. Pack Mime allows the generation of both HTTP/1.0 and HTTP/1.1 traffic.

3 Properties of Cloud Management Systems

3.1 Cloud Computing

The principal goal of Cloud Computing is to deliver on-demand computing services with high reliability, scalability, and availability in distributed environments. Despite this common goal, Cloud Computing [1] has been defined in many different ways [4] and no standard definition has been adopted until now.

A recently posted working definition of cloud computing by The Information Technology Laboratory at the National Institute of Standards and Technology (NIST) [9] states: “Cloud Computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics (Rapid Elasticity, Measured Service, On-Demand Self-Service, Ubiquitous Network Access, Location-Independent Resource Pooling), three delivery models (Software as a Service, Platform as a Service, and Infrastructure as a Service), and four deployment models (Public Cloud, Private Cloud, Community Cloud and Hybrid Cloud)".

Theoretically, in Cloud Computing everything is anticipated as a service (XaaS) [1], such as TaaS (Testing as a Service), SaaS (Software as a Service), PaaS (Plat-form as a Service), HaaS (Hardware as a Service). To this end, a large number of cloud service providers and middleware suits have emerged, each providing different Cloud Computing services. These providers include Amazon EC2, Google App Engine (GAE), SalesForce.com (SFDC), Microsoft Azure, IBM Blue Cloud, 3Tera etc.

3.2 Virtualization

Virtualization refers to that which does not exist in real, but provides everything like it is real. Virtualization is the software implementation of a machine, which will execute different programs like a real machine. Through the virtualization, user can use the different applications or services of the cloud. Ostensibly, this is the main part of the cloud environment. There are different types of virtualization used in cloud environment. Two types of virtualization are:

- Full Virtualization: Full virtualization refers to when a complete machine is installed on another machine. That virtual machine provides all the function, which exists on the original machine. It facilities only when an actual machine is unavailable then the user can use the virtual machine.
- Para virtualization: Para virtualization refers to when the hardware allows multiple operating systems to run on single machine. It also allows efficient use of system resources such as memory and processor.

3.3 Virtual Desktop Infrastructure VDI

Virtual desktop infrastructure is a desktop virtualization approach in which a desktop operating system, typically Microsoft Windows, runs and is managed in a data center. The desktop image is delivered over a network to an endpoint device, which allows the user to interact with the OS and its applications as if they were running locally. The endpoint may be a traditional PC, thin client or even a mobile device. The term coined by VMware. Virtual desktop infrastructure (VDI) is a virtualization technology that hosts a desktop operating system on a centralized server in a data center. VDI is a variation on the client-server computing model, sometimes referred to as server-based computing [10]. Virtual desktop infrastructure or VDI is a computing model that adds a layer of virtualization between the server and the desktop PCs. By installing, this virtualization in place of a more traditional operating system, network administrators can provide end users with access anywhere capabilities and a familiar desktop experience, while simultaneously heightening data security throughout the organization [13].

3.4 VDI Architecture

VDI is an architecture requiring carefully crafted solutions that meet specific needs. All VDI solutions have virtualization of the users desktop in common. A complete VDI solution may also include other design elements that compliment, extend, or leverage the core features of VMware Infrastructure virtualization platform. A full spectrum VDI solution starts with the users access device and includes a number of logically sequential components spanning the full lifecycle of user activity [13]. VDI is the one of peer-to-peer solution that reduces IT management workload, in-creases security and increases control of end user access while lowering costs by centrally delivering desktop services. Though using a remote display protocol over the network, VDI clusters put desktop environments on cloud servers or local servers to deliver a desktop-centric service [14].

4 Load Balancing in Cloud Computing

Load balancing, as is shown in Fig. 1, is a new technique that provides high resource time and effective resource utilization by assigning the total load among the various cloud nodes [12], side by side; it solves the problem of overutilization and underutilization of virtual machines. Load balancing resolve problem of overloading, focuses on maximum throughput, optimizing resource utilization, and minimize response time. Load balancing is the prerequisite for maximizing the cloud performance and utilizing the resources efficiently. In utilization of clouds, there has been an improved resource allocation method using preemptible task

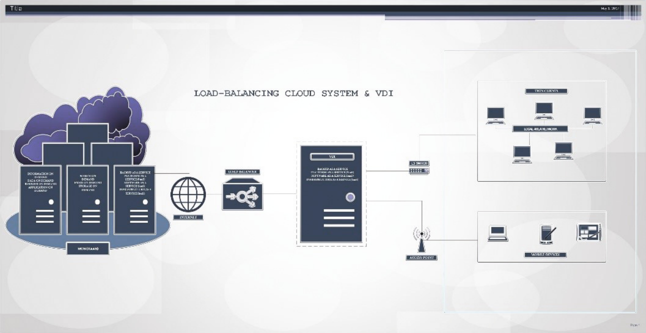

Fig. 1. Load balancing cloud system and VDI

execution. Adaptive resource allocation algorithm is presented for cloud system with preemptible tasks but this approach does not resolve the problem of response time and effective cost utilization.

4.1 Load Balancing Algorithms

In cloud computing, different load balancing algorithm have been proposed of which the main purpose is to achieve high throughput and minimum response time. Generally, load-balancing algorithm is of two types:

- A. Static load balancing algorithm
- B. Dynamic load balancing algorithm

The subsequent load balancing algorithms are currently prevalent in clouds.

4.2 Static Load Balancing Algorithm

The load does not depend on the current state of the system but it requires knowledge about the application and resources of the system. Static load balancing is a load balancing algorithms that distributes the workload based strictly on a fixed set of rules such as input workload. There are four different types of Static load balancing techniques: Round Robin algorithm, Central Manager Algorithm, Threshold algorithm and randomized algorithm.

- Round Robin Algorithm. It is one of the simplest scheduling algorithms that utilize the principle of time slices. Here, time is divided into multiple slices and each node is given a particular time interval. Each node is given a quantum and in this given quantum node has to perform its operations. If the user request completes within time quantum then user should not wait otherwise have to wait for its next slot. It means that this algorithm selects the load randomly, while in some case, some server is heavily loaded or someone is lightly loaded.
- Throttled Load Balancing Algorithm. This algorithm is totally based on the allocation of request to virtual machine. Here, client will first request the load balancer to check the right virtual machine, which access that load easily and performs the operations request by client or user. In this algorithm, the load balancer maintains an index table of virtual machines as well as their states (Available or Busy) [12]. Therefore, the client first requests the load balancer to find a suitable Virtual Machine to perform the required operations. These dynamic algorithms are being experimentally performed using the cloud analyst tool, which gives the output with respect to virtual machine

4.3 Dynamic Load Balancing Algorithm

Dynamic algorithms are more flexible than the static algorithm and do not rely on prior knowledge but depends on current state of the system. In a distributed system, dynamic load balancing has two different ways: distributed and non-distributed. In the distributed one, all nodes present in the system execute this algorithm and the task of load balancing is shared among these servers. The interaction among nodes to achieve load balancing can take two forms: cooperative and non-cooperative. In the first one, the nodes works side-by-side to achieve a common objective, which means is to improve the overall response time, etc. In the second form, each node works independently toward a goal local to it.

- Opportunistic Load Balancing Algorithm. This is static load balancing algorithm so it does not consider the current workload of the VM. It attempts to keep each node busy. This algorithm deals quickly with the unexecuted tasks in random order to the currently available node. Each task is assigned to the node randomly. It provides load balance schedule without good results. The task will process it slow in manner because it does not calculate the current execution time of the node.
- Min-Min Load Balancing Algorithm. The cloud manager identifies the execution and completion time of the unassigned tasks waiting in a queue. This is static load balancing algorithm so the parameters related to the job are known in advance. In this type of algorithm, the cloud manager first deals with the jobs having minimum execution time by assigning them to the processors according to the capability of complete the job in specified completion time. The jobs having maximum execution time has to wait for the unspecific period. Until all the tasks are assigned in the processor, the assigned tasks are updated in the processors and the task is removed from the waiting queue. This algorithm performs better when the numbers of jobs having small execution time is more than the jobs having large execution time. The main drawback of the algorithm is that it can lead to starvation.
- Max-Min Load Balancing Algorithm. Max-Min algorithm works similar to the Min-Min algorithm except the following: after finding out the minimum execution time, the cloud manager deals with tasks having maximum execution time. The assigned task is removed from the list of the tasks that are to be assigned to the processor and the execution time for all other tasks is updated on that processor. Because of its static approach the requirements are known in advance then the algorithm performed well. An enhanced version of max-min algorithm was proposed. It is based on the cases where meta-tasks contain homogeneous tasks of their completion and execution time. Improvement in the efficiency of the algorithm is achieved by increasing the opportunity of concurrent execution of tasks on resources.
- The Two Phase Scheduling Load Balancing Algorithm. It is the combination of OLB (Opportunistic Load Balancing) and LBMM (Load Balance Min-Min) Scheduling algorithms to utilize better execution efficiency and maintain the load balancing of the system. OLB scheduling algorithm keeps every node in working state to achieve the goal of load balance and LBMM scheduling algorithm is utilized to minimize the execution of time of each task on the node thereby minimizing the overall completion time. This algorithm works to enhance the utilization of resources and enhances the work efficiency.

- Ant Colony Optimization Based Load Balancing Algorithm. Aim of the ant colony optimization to search an optimal path between the source of food and colony of ant based on their behavior. This approach aims at efficiently distributing workload among the nodes. When request is initialized, the ant starts movement towards the source of food from the head node. Regional Load Balancing Node (RLBN) is chosen in Cloud Computing Service Provider (CCSP) as a head node. Ants keep records of every node they visit and record their data for future decision making. Ant deposits the pheromones during their movement for other ants to select next node. The intensity of pheromones can vary on the bases of certain factors like distance of food, quality of food etc. When the job gets successful the pheromones is updated. Each ant builds their own individual result set and later on built into a complete solution. The ant continuously updates a single result set rather than updating its own result set. By the ant pheromones trials, the solution set is continuously updated.
- Honeybee Foraging Load Balancing Algorithm. It is a nature inspired decentralized load balancing technique, which helps to achieve load balancing across heterogeneous virtual machine of cloud computing environment through local server action, and maximize the throughput. The current workload of the VM is calculated then it decides the VM states whether it is over loaded, under loaded or balanced according to the current load of VM they are grouped. The priority of the task is taken into consideration after it is removed from the overload VM, which are waiting for the VM. Then the task is schedule to the lightly loaded VM. The earlier removed task are helpful for the finding the lightly loaded VM. These tasks are known as scout bee in the next step. Honey Bee Behavior inspired Load Balancing technique reduces the response time of VM and reduces the waiting time of task.
- Active Clustering Load Balancing Algorithm. Active Clustering works based on grouping similar nodes and increases the performance of the algorithm the process of grouping is based on the concept of matchmaker node. Matchmaker node forms connection between its neighbors that is like the initial node. Then the matchmaker node disconnects the connection between itself and the initial node. The above set of processes is done repetitively. The performance of the system tends to increase because of high availability of resources, because of that, the throughput is propelled to also increase.

4.4 Motivation of Load Balancing Algorithms

There are countless reasons for coalescing cloud management systems with virtual desktop infrastructure of which the major is load balancing. Efficient load balancing algorithm provides efficient and dynamic workload across nodes. Research indicates the countless problems posed by this imbalance and inefficiency in these systems to provide these services to requesting nodes. Many cloud systems vendors keeps proliferating providing services to organizations hence the failure or imbalance of load distribution dynamically to various subscribers, therefore a critical reason which calls for thorough research in this area of study.

4.5 Analysis of Load Balancing Algorithms

Based on a collective and qualitative approach various documentations were reviewed relating to load balancing algorithms in cloud management systems and VDI. A comparison of the various existing load balancing algorithms were revisited. The table below presents the comparison table of the aforementioned. The table gives us detailed descriptions of the various mechanisms of load balancing algorithms which has been made simple with a TRUE and FALSE elaborating its capabilities and limitations individually in Table 1.

Table 1. Algorithm comparative metrics table

	ROUND ROBIN	OLB	MIN MIN	MIN MAX	TWO PHASE
Fault tolerance	F	F	F	F	F
Overhead	T	F	T	T	T
Throughput	T	F	T	T	T
Response time	T	F	T	T	T
Performance	T	T	T	T	T
Scalability	F	F	F	F	F
Resource utilization	T	T	T	T	
	ANT COLONY	HONEY BEE	ACTIVE CLUSTERING	BIASED RANDOM SAMPLING	
Fault tolerance	F	F	F	F	
Overhead	F	F	T	F	
Throughput	T	T	F	T	
Response time	F	F	F	F	
Performance	T	T	F	T	
Scalability	T	T	F	F	
Resource utilization	T	F	T	F	

5 Conclusion and Future Work

This paper has presented a survey on cloud computing and various algorithm for load balancing in cloud computing. It is no doubt that cloud computing is one of the most emerging technology in computer science but it also has some lapses and load balancing is one of the major lapses of the cloud. This issue can be resolved by using various load balancing algorithm that balances the workload. This paper reviews some of the various load balancing algorithms like Honey Bee, Round Robin, OLB, BRS, Active Clustering, Min-Min, Min-Max and Ant Colony Optimization. Further, the survey also highlights comparison of the parameters between algorithms under review and their distinguishing properties.

According to the survey, none of the algorithms seem to completely satisfy the comparative metrics to solve the lapses of load balancing in cloud computing, as such load balancing still remains a critical issue for scientific research.

References

1. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R.H.: Above the clouds: a berkeley view of cloud computing. Commun. ACM **53**(4), 50–58 (2009). Eecs Department University of California Berkeley
2. Begum, S., Prashanth, C.S.R.: Review of load balancing in cloud computing. Int. J. Comput. Sci. Issues **10**(1), s536 (2013)
3. Cao, J., Cleveland, W.S., Gao, Y., Jeffay, K., Smith, F.D., Weigle, M.: Stochastic models for generating synthetic HTTP source traffic. In: Joint Conference of the IEEE Computer and Communications Societies, vol. 3, pp. 1546–1557 (2004)
4. Cavoukian, A.: Privacy in the clouds - a white paper on privacy and digital identity implications for the internet (2008). https://www.ipc.on.ca/wp-content/uploads/2008/05/privacyintheclouds.pdf
5. Dasilva, D.A., Liu, L., Bessis, N., Zhan, Y.: Enabling green it through building a virtual desktop infrastructure. In: Eighth International Conference on Semantics, Knowledge and Grids, pp. 32–38 (2012)
6. Daz, M., Martn, C., Rubio, B.: State-of-the-art, challenges, and open issues in the integration of internet of things and cloud computing. J. Netw. Comput. Appl. **67**(C), 99–117 (2016)
7. Liu, J., Lai, W.: Security analysis of VLAN-based virtual desktop infrastructure. In: International Conference on Educational and Network Technology, pp. 301–304 (2010)
8. Lu, Y., Xie, Q., Kliot, G., Geller, A., Larus, J.R., Greenberg, A.: Join-idle-queue: A novel load balancing algorithm for dynamically scalable web services. Perform. Eval. **68**(11), 1056–1071 (2011)
9. Mell, P., Grance, T.: The nist definition of cloud computing. Commun. ACM **53**(6), 50 (2009)
10. Moharana, S.S.: Analysis of load balancers in cloud computing. Int. J. Comput. Sci. Eng. **2**(2), 101–108 (2013)
11. Moura, J., Hutchison, D.: Review and analysis of networking challenges in cloud computing. J. Netw. Comput. Appl. **60**, 113–129 (2016)
12. Nakai, A.M., Madeira, E., Buzato, L.E.: Improving the QOS of web services via client-based load distribution. In: XXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (2011)
13. Sharma, R., Kumar, A.: Load balancing in cloud computing system. AJES, 1(2) (2012)
14. Shimonski, R.: Windows server 2003 clustering & load balancing (ebook), 1st edn. Windows Server. McGraw-Hill Education (2003). ISBN-13: 978-0072226225, ISBN-10: 0072226226
15. Wang, S.C., Yan, K.Q., Liao, W.P., Wang, S.S.: Towards a load balancing in a three-level cloud computing network. In: IEEE International Conference on Computer Science and Information Technology, pp. 108–113 (2010)
16. Zhang, Z.P., Wen, L.J.: Loba-min-min: The grid resources scheduling algorithm based on load balance. Appl. Mech. Mater. **321–324**, 2507–2513 (2013)