Workload Balance based Dynamic Resource Allocation Model in the Cloud Data Center

Hairui Zhang [1, a], Minjuan Li [2], Jianbo Cui [1]

[1]School of Information Science and Engineering, Lanzhou City University, Gansu 730000, China

[2]Lanzhou Jiao Tong University, Gansu 730000, China

[a]82791881@qq.com

Keywords: workload balance, cloud data center, resource allocation.

Abstract. In the cloud computing environment, the data center is very important, which provides on-demand information technology resources to end-user. The major function of cloud data center is a balanced allocation of resources for application requests. Clearly, the end-user will pay more attention to response time of request and service rate of cloud resource, while the cloud resource provider may focus on costs and energy consumption. The workload balance is a key factor. In the paper, we will firstly study the relationship between the workload balance and these parameters, and then an algorithm is proposed based on the workload balance for guaranteeing the end-user's Quality of Service and the provider's costs. A simulation experiment will show that our algorithm can allocate resources more reasonably in terms of application requests in the cloud data center.

Introduction

Cloud computing is a recent technology that concerns with online distribution of computing resources and services [1]. These services can be accessed by the users in a Pay-per-Use-On-Demand model, which can access shared IT resources such as server, data storage, application, network, and so on, through the Internet [2]. Data center is the fundamental physical unit of cloud computing, and has large-scale software infrastructure, data storage resources and hardware platform. In a cloud data center, end-users can request a resource to run their application whenever they are in need. For this reason, Cloud Computing is also described as on-demand computing [3]. From perspective of end-user, this process of using resources does not only enhance stability, but also reduces response time. Thus, how to increase the utilization of resources of cloud data center is very important for each cloud resource provider.

A cloud data center construction and operation costs is very high. At present, an increasing number of organization started to provide IT resources (e.g., CPUs, memory, disks, network) [4]. Since a cloud data center consists of physical servers, to appropriately leveraging these servers has decreased considerable operation costs. There are two very important problems about leveraging physical servers in cloud data center. One is overload of server, which lead to extend response time and lower the QoS for end-users. Another aspect is workload of server is lower or even idle. One important issue associated with this field is dynamic load balancing or task scheduling [5]. Workload balance of each server will reduce costs and guarantee QoS in cloud data center.

In this paper, we will study workload model and present an algorithm for solving the above-mentioned problems. State data of the center will be fed back to workload manager, and then used by the algorithm to satisfied QoS of end-users. Moreover, we must pay attention to energy consumption and costs. Energy consumption and resource utilization in clouds are highly coupled [6], so each server's workload is too low or even idle to avoid waste of resources and energy in the cloud data center.

Workload Model Study

Workload Framework Environment. In our workload architecture, we are separating between control flow and data flow, which is very significant principle. As Fig.1 shows, our system architecture consists of three fundamental components.

The data center will provide several resources for task request as resource pool. Its role is infrastructure platform for overall system architecture. The monitor node regularly gathers and updates the state information (e.g., resource utilization, server workload and task request). These information be used in analyze current situation of entire system, and to provide data support for workload balance strategy. The manager has two main modules, which are workload balance strategy and balance algorithm. It will do three important things: (1) communicate with the monitor node and receive the state information of overall system; (2) allocate resource for end-user's task request; (3) control server workload in terms of balance strategy to decrease energy consumption and waste of resources in cloud date center (e.g., if there are two servers which hold lower workload, we can combine them).

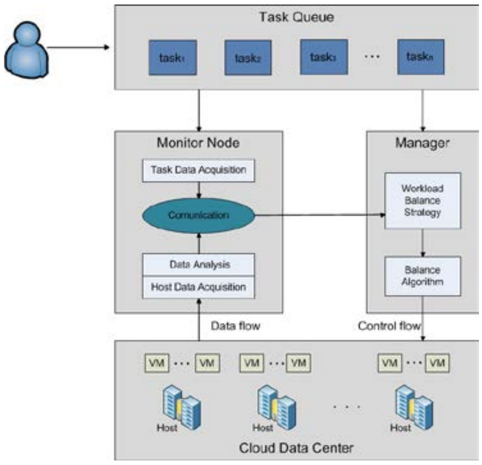


Fig. 1 Workload Architecture

System Framework Model. In cloud data center, we assume that each server cloud provide n kinds of different resources. We express them with $R_i = (i = 1, 2, \cdots, n)$, here n is a finite positive integer. These resources will be provided in the form of virtual machine (VM), and each server can hold several VMs. When the system allocates resources for end-user's task request, the number of VMs cannot exceed capacity of server workload.

Taking into account the above problems, our system architecture is a time-slot architecture. Let time-slot is *t*, and we can dynamically adjust it. In our system, the state information is constantly changing. We must update these data information at the beginning of each slot, and then we will select appropriate server according to balance algorithm for requirement of end-user.

System Performance Metric. We will use QoS and average workload ratio as performance metric in our paper. Generally, the QoS is a user-oriented and includes two aspects which are response time and service rate respectively [4]. The provider of cloud data center must improve service rate and reduce response time to attract end-user. The average workload ratio refers to the weighted average workload ratio of each resource in the server. In different circumstances, the weighted value of workload ratio of every resource is different, so it is important to select appropriate weighted value for them according to continuous statistics.

Workload Model. In this paper, we assume that there are *K* servers in the cloud data center. According to the previously mentioned, each server will have n kinds of resources and they are provisioned to end-users who propose task request. For simplification, the maximum capacity of every resource is assumed to be same in the same server, which is denoted by $cap_i (i = 1, 2, \cdots, n)$.

In each slot t, the number of each kind of resource provided by the server k is denoted by $\lambda_{ik}(t)$. At the beginning of each slot, the monitor node will re-collect and calculate these values. So we have:

$$0 \le k \le K \tag{1}$$

$$0 \le \lambda_{ik} \le cap_i \tag{2}$$

In order to work in an optimal status, the workload of each server cannot exceed the peak. Let $L_k(t)$ be the workload ratio of the server k in slot t. obviously, the less is the resources which the server could provide, the more is the corresponding workload. So the workload ratio and the amount of resources provided by each server hold an inversely-proportional relationship, expressed as following:

$$L_k\left(t\right) = f\left(\lambda_{1k}, \lambda_{2k}, \cdots, \lambda_{nk}, cap_1, cap_2, \cdots, cap_n\right) \tag{3}$$

In this function, a linear model can be fixed among $\lambda_{ik}(t)$ and cap_i, where $i = 1, 2, \cdots, N$, and it can be established as follows under the help of two estimated parameter a, b:

$$L_k\left(t\right) = a\sum_{i=1}^{n}\lambda_{ik}\left(t\right) + b\sum_{i=1}^{n}cap_i \tag{4}$$

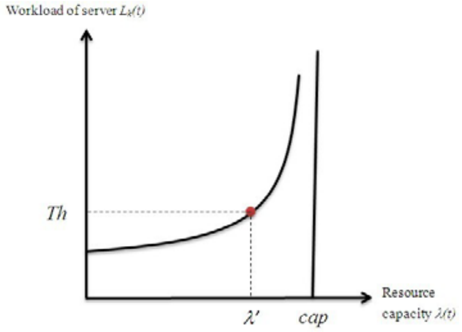


Fig. 2 Resource capacity and workload at the server.

Let $\lambda(t)$ be $\sum_{i=1}^{n}\lambda_{ik}(t)$ and cap be $\sum_{i=1}^{n}cap_i$. Formula (4) turns to be $L_k(t) = a \cdot \lambda(t) + b \cdot cap$. Obviously, $L_k(t)$ is a quadratic function about $\lambda(t)$, as shown in Fig. 2.

In Fig. 2, the workload ratio of the server increases as the value of $\lambda(t)$. When $\lambda(t)$ increases to a fixed value $\lambda^{'}$, the workload ratio will increase sharply. Let Th be the workload ratio peek of the server (i.e., maximum allowable workload). In our system framework, the working status of each server must be monitored dynamically in order to make sure the workload ratio cannot exceed the value of Th.

Workload Balance Algorithm

We design the workload balance (W-B) algorithm based on the state information of system and workload balance strategy. The state information will be provided by the monitor node, and the manager is responsible for carrying out workload balance strategy. For each slot *t*, we will execute the algorithm for allocating resource and ensuring workload balance of data center. Now, we describe our algorithm think based on server parameters.

Let $Q_j(t)$ be the amount of end-user's requests in each slot *t*, here $j = (1, 2, \cdots, m)$. Each request from end-users is an *n* tuple, expressed as $Q_j(t) = (q_{j1}, q_{j2}, \cdots, q_{jn})$ where q_{ji} denotes the requested number of each resource. So we can compute the total demand of each resource $\mu_i(t)$ in slot *t* with following:

$$\mu_i(t) = \sum_{j=1}^{m} q_{ji} \tag{5}$$

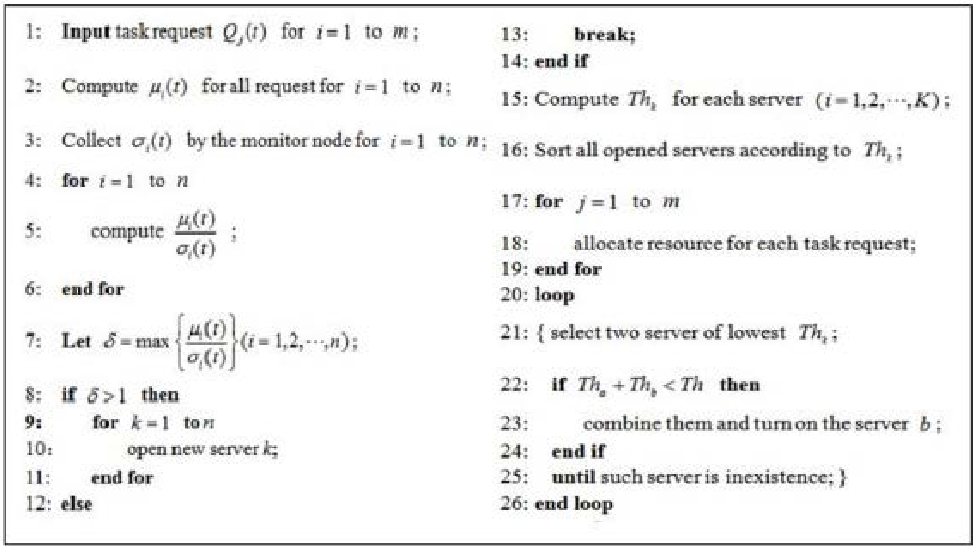
Let $\sigma_i(t)$ be the total number of each resource that can be used in the cloud data center. It will be collect and calculate by the monitor node at the beginning of slot *t*. This value also will decide whether the new servers need to open.

Let Th_k be the current workload ratio of the server *k*. In our data center, we specified that Th_k of each server cannot exceed Th . We can compute it as follows:

$$Th_k = \sum_{i=1}^{n}\left(\alpha_i \cdot \frac{\lambda_{ik}}{cap_i}\right) \tag{6}$$

$$\sum_{i=1}^{n}\alpha_i = 1, \alpha_i \in [0,1] \tag{7}$$

In order to decrease costs and ensure workload balance of server in cloud data center, we should select some servers with lowest workload and combine them when slot *t* is over. Fig.3 shows the all detail of our algorithm.



```
Input task request $Q_j(t)$ for $i=1$ to $m$;
Compute $\mu_i(t)$ for all request for $i=1$ to $n$;
Collect $\sigma_i(t)$ by the monitor node for $i=1$ to $n$;
for $i=1$ to $n$
    compute $\frac{\mu_i(t)}{\sigma_i(t)}$;
end for
Let $\delta = \max\left\{\frac{\mu_i(t)}{\sigma_i(t)}\right\}(i=1,2,\cdots,n)$;
if $\delta>1$ then
    for $k=1$ to $n$
        open new server $k$;
    end for
else
    break;
end if
Compute $Th_k$ for each server $(i=1,2,\cdots,K)$;
Sort all opened servers according to $Th_k$;
for $j=1$ to $m$
    allocate resource for each task request;
end for
loop
{ select two server of lowest $Th_k$;
    if $Th_a + Th_b < Th$ then
        combine them and turn on the server $b$;
    end if
    until such server is inexistence; }
end loop
```

Fig. 3 The workload balance algorithm

Result and Emulation

In this section we will demonstrate the validity and reliability of our system model in a simulation environment. We assume that there are 20 servers in cloud data center. Among these servers, 10 servers are always opened, and the rest will be dynamically turned on/off. We assume that each server will provide two kinds of resources which are CPUs and memory, and the capacity of the two resources is equal. So let $cap_i = 100$ for each server. The workload ratio also is a very important parameter. We used the average utilization rate of the two resources as the workload ratio of the server. Thus let $\alpha_1 = \alpha_2 = 0.5$, we will calculate the workload ratio of every server with formula (6). In our experiment, all servers have the same configuration.

In our emulation environment, the end-user's requests will continue to arrival the system, and q_{ji} of each request is a random positive integer ($q_{ji} \in [0,50]$). There are 300 requests which will be given. We assume the maximum value of workload ratio cannot exceed 65% for guaranteeing QoS of end-users. As shown in Fig. 4, with W-B algorithm, the average workload ratio of most server is between40% to 65% there are 4 servers which the average workload ratio is lower (less than 40%). Therefore, our algorithm is effective to adjust workload balance of cloud data center.

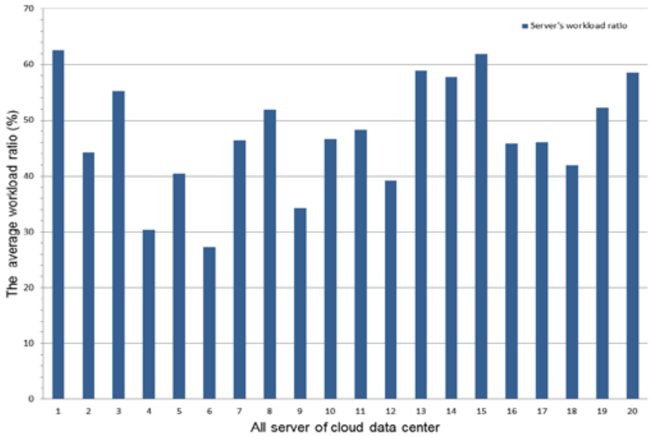


Fig. 4 The average workload ratio of server

Conclusion

This paper proposes a dynamic resources allocation model and W-B algorithm based workload balance strategy. This model has two core modules which are the monitor node and manager respectively. They will implement the workload balance strategy through collecting and analyzing state information of the system. Meanwhile, the end-users will obtain a good QoS and the costs and waste of resource also will be greatly reduced in cloud data center. In the future, an actual system will be implemented the model and algorithm which the paper study. In additionally, we only take into account the relationship between system workload and two resources (CPUs and Memory) in our simulation, hence, we should consider more resources (e.g., disk, network etc.) in our workload balance strategy.

Acknowledgment

The authors would like to thank the Science Foundation of Urban Development Institute of Gansu Province (2013-GSCFY-KJ08) for supporting this research.

References

[1] M. D. Dikaiakos, G. P. is, D. Katsaros, P. Mehra and A. Vakali, Cloud computing : Distributed Internet Computing for IT and Scientific Research, IEEE Internet Computing, Published by the IEEE Computer Society, 2009.

[2] S. K. Dhurandher, M. S. Obaidat, I. Woungang, P. Agarwal1, A. Gupta and P. Gupta, A Load Balancing Strategy for Cloud Computing Environment, International Conference on Signal Propagation and Computer Technology (ICSPCT), 2014, p.636-641.

[3] T. C. Chieu, A. Mohindra, A. A. Karve and A. Segal, Dynamic Scaling of Web Applications in a Virtualized Cloud Computing Environment, IEEE International Conference on e-Business Engineering, 2009, p.281-286.

[4] H. Zhang, Y. Yang, L. Li, W. Cheng and C. Ding, A Dynamic Resource Allocation Framework in the Cloud, Applied Mechanics and Materials, Vol. 441(2014), p.974-979.

[5] K. A. Nuaimi, N. Mohamed, M. A. Nuaimi and J. Al-Jaroodi, A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms, IEEE Second Symposium on Network Cloud Computing and Applications, 2012, p.137-142.

[6] Y. C. Lee, A. Y. Zomaya, Energy efficient utilization of resources in cloud computing systems, J Supercomputer (2012) 60, p.268–280.