ATLANTIS PRESS
Advances in Engineering Research, volume 166
3rd International Conference on Automation, Mechanical Control and Computational Engineering (AMCCE 2018)

Energy-Efficient Resource Allocation Strategy Based on Task Classification in Data Center

Hongjian Li[1,2,a], Shiwang Ding[1,2,b], Pengfei Zhang[1,2,c], Jun Lai[1,d]

[1]School of computer science and technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China;

[2]Chongqing Engineering Research Center of Mobile Internet Data Application, Chongqing, 400065, China;

[a] lihj@cqupt.edu.cn, [b] 1870901779@qq.com, [c] 474644669@qq.com, [d] laijun@cqupt.edu.cn

Abstract: In view of the fact that the research on energy efficiency in data centers has not fully considered the heterogeneity of workload tasks, a data center resource allocation scheme based on task classification is proposed. Through cluster analysis, tasks are classified into subsets of similar resources and performance requirements. The same types of tasks to configure a reasonable type of virtual machine to improve the compatibility between workload requirements and configuration resources. While ensuring the QoS requirements of different types of tasks, resources preemption of tasks of the same resource requirement type is avoided, enabling energy-efficient resource allocation with low-power and QoS guarantees in the data center. Simulation experimental results find that, compared with the traditional resource allocation algorithm, this scheme effectively improves the data center energy efficiency.

Keywords: data center; workload characterization; task classification; energy efficiency; resource allocation

1. Introduction

High energy consumption has always been an important challenge to data center resource management. Literature [1] shows that an average data center consumes the same amount of energy as 25,000 homes. High energy costs not only lead to increased costs, lower cloud infrastructure return to investment (ROI), but also generate significant carbon dioxide (CO2) emissions, with data center emissions estimated to be 2% of global emissions. Therefore, to meet the service level agreement (service level agreement, SLA), that is to meet the QoS requirements and ensure performance, reducing the energy consumption of cloud data center is of great significance.

Heterogeneity characteristics exist on data centers [2], Including heterogeneous machines with different performance and energy consumption characteristics, as well as workload tasks with different QoS and resource requirements. Currently, most energy-efficient research on data centers neglects heterogeneity feature. In order to enabling energy-efficient resource allocation with low-power and QoS guarantees in the data center, an energy-efficient resource allocation strategy based on task classification (Classification-Based Resource Allocation Strategy, CBRAS) is proposed. This scheme fully considers the heterogeneity of workload task in the data center and classify the task into different task classes with similar resource and performance requirements. Then, based on Knapsack Problem, the same types of tasks are assigned a reasonable type of virtual machine to achieve data center energy efficiency resource allocation. Simulation experimental results find that, compared with the traditional

resource allocation algorithm, the CBRAS based on task classification designed in this paper effectively improves the data center energy efficiency.

The rest of the paper is organized as follows. Section 1 presents the related work in this area. Section 2 provides a classification and analysis of a publicly available workload traces from Google. Section 3 introduces the resource allocation scheme based on task classification. Section 4 experiments on CloudSim simulation tool to evaluate the proposed scheme. Finally, Section 5 presents conclusions and discusses future research directions.

2. Related Work

There is a great deal of research on improving data center energy efficiency. The goal of resource management has shifted from simply reducing energy consumption to reducing energy consumption while meeting application QoS, that is, to meet SLA compatibility and reduce energy consumption. Ren et al. [3] studied the problem of scheduling heterogeneous batch workload across geographically distributed data centers, but the assumption that the workload has been divided into different categories. Srikantaiah et al. [4] studied the resource scheduling problem of Virtual heterogeneous system in multi-layer web applications, in order to solve the optimization problem of many kinds of resources, the author proposed a heuristic multi-dimensional packing algorithm for workload consolidation. However, this method depends on the workload type and application.

In previous data center resource scheduling solutions, ubiquitous heterogeneity in data centers was often overlooked. There are currently some studies that analyze the characteristics of workload tasks. A comprehensive analysis of Google cluster tracking data by Reiss [5] and Md. Rasheduzzaman [6] found that the machine configuration and workload composition are highly heterogeneous and dynamic over time, indicating that recognizing the heterogeneity of the workload is important to resource scheduling. Chen et al. [7] observed the statistical analysis and cumulative distribution of Google cluster tracking data set, then applied K - means algorithm to workload characteristics, provided the characteristic description of Google cluster workload at job level. Raque V. Lope et al. [8] proposed a general classification method for distributed system scheduling problems, and provided a detailed classification analysis of workload description, resource description and scheduling requirements, which has some reference significance. However, these studies basically only analyze the characteristics of workload resource requirements, and do not consider the heterogeneity of task QoS requirements well. Most of the classification indexes focus only on resource requirements, such as CPU, memory, disk capacity, Bandwidth and other different resource requirements for classification, did not fully consider the user QoS requirements, which can not to reduce energy consumption at the same time better QoS guaranteed.

In view of the research status quo, this paper proposes an energy efficient resource allocation scheme based on workload task classification to improve the compatibility between workload requirements and configuration resources, and to achieve energy efficient resource allocation in data centers.

3. Task Classification and Analysis

3.1 Google Cluster Workload Traces Overview

This paper uses Google data center workload traces data public in 2014 (version 2) [9] as the basis, the data is a 29-day workload traces data feed from a cluster in Google Data Center that consists of 6 tables, which are introduced as follows.

The Machine event and Machine attribute tables describe the machine events and attribute in the cluster, and the Job event table and the Task event table describe the job / task and its lifecycle. All jobs

and tasks have a scheduling class field information, which roughly indicates the delay sensitivity of work / task. The scheduling class is represented by a 0~3 number, the greater the value, the higher the sensitivity of delay. There is a priority field information in task event table, the degree of priority rough representation of the task, the priority is represented by the 0~11 number, task priority for 4 different task types The Task constraint table describes the placement constraints of task scheduling. The Resource usage table describes the resource usage of the task's CPU, memory, disk, and so on. In the available traces, resource utilization measurements and requests are normalized, and the normalization is performed separately for each column. In this context, to get a real sense of the data, we assume the largest amount of resources for each column, which CPU maximum core frequency of 3.2 GHz, the memory is 4GB, the disk is 1GB, then each recorded data needs to be multiplied to related values (e.g. for recorded memory utilization we have $Real_{util} = Recorded_{Util} * 4$).

3.2 Task Classification

3.2.1 Task Classification Method

The data center receives a large number of heterogeneous resource requests, which have different resource requirements, durations, priorities and performance goals. The goal of task classification is to divide tasks into classes with similar resource requirements and performance characteristics in order to efficiently allocate available resources. And the selection of clustering criteria affects the goal of resource allocation strategy. In order to achieve the goal of energy-efficient resource allocation, task classification needs to fully consider user QoS requirements while considering the task resource requirements. This paper uses the following four clustering criteria: task length, task priority, task delay sensitivity and Task resource requirements including task CPU, memory requirements.

In this paper, Based on the task clustering using the classical k-means algorithm, combines the existing coarse-grained classification results of some fields in the Google original tracking data, and obtains the best clustering results through continuous loop analysis and experiment.

Specifically, such as the size of the task i can be modeled as a vector $\boldsymbol{s}^i = (\boldsymbol{s}^{i1}, ..., \boldsymbol{s}^{iF})$, where F denotes the set of features used for clustering. Let $\boldsymbol{N}_k$ denote the tasks that belong to cluster k, Then, the centroid of each cluster can be defined as a vector $\overline{\mu}^k = (\overline{\mu}^{k1}, ... \overline{\mu}^{kF})$, where $\overline{\mu}^{kr} = \frac{1}{|N_k|} \sum_{s^i \in N_k} \boldsymbol{s}^{ir}$. The K-means clustering algorithm essentially tries to minimize the following similarity score:

$$score = \sum_{i=1}^{k} \sum_{i \in N_k} || \boldsymbol{s}^i - \overline{\mu}^k ||^2 \tag{1}$$

Where $|| a - b ||$ denotes the Euclidian distance between two points ‘a’ and ‘b’ in the feature space. Because the K value of K-means algorithm, that is, the number of clustering and the initial clustering center have a great influence on the clustering result, based on the result of K-means algorithm running and the result of statistical analysis of data, we update K value and the initial cluster center, and get more reasonable clustering results through continuous loop analysis and experiment.

3.2.2 Task Classification Results and Analysis

According to the task length, priority, delay sensitivity and resource requirements, the task is classified. The result shows that there are 10 task categories. The detailed classification result information is shown in Table 2, in which the priority and delay sensitivity values are the average of the major values of the task class.

In order to better to understand the characteristics of the task cluster, the use of a tree structure to display the classification results is shown in Figure 1, where the average task duration <1 and <5 h respectively represent the short and medium length, and the average task length> 5 h is considered Is

very long, task priority above 4 is considered a high priority, while the task CPU> 0.1, memory> 0.1 is considered a big task, represented by L, small S.

Figure 1 shows that nearly 78% of the tasks belonging to shorter categories (Cluster 1~6), and more than 50% tasks are very short (less than 100 seconds); in addition, the longest duration of the task, have high priority (Cluster 10); most of the task has lower resource demand, and most a large task is CPU intensive (Cluster 5) or memory intensive (Cluster 3), little task both CPU and memory utilization are high (7 Cluster).

Table 1 Task classification cluster statistics

Cluster	Priority	Delay sensitive	Resource requirements size		Task length	Proportion
			CPU	Memory		
Cluster _1	8	0	0.005	0.0038	6.04(min)	2%
Cluster _2	7	3	0.0213	0.0548	38.66(min)	8%
Cluster _3	0	0	0.0979	0.1441	56.82(min)	4%
Cluster _4	2	0	0.0101	0.0062	20.29(min)	26%
Cluster _5	2	1	0.1659	0.0543	34.39(min)	18%
Cluster _6	3	1	0.0101	0.0127	29.19(min)	20%
Cluster _7	4	0	0.1579	0.2895	1.04(h)	2%
Cluster _8	0	0	0.036	0.022	3.09(h)	10%
Cluster _9	1	2	0.0221	0.05	1.45(h)	10%
Cluster _10	9	1	0.011	0.125	18.19(h)	<1%

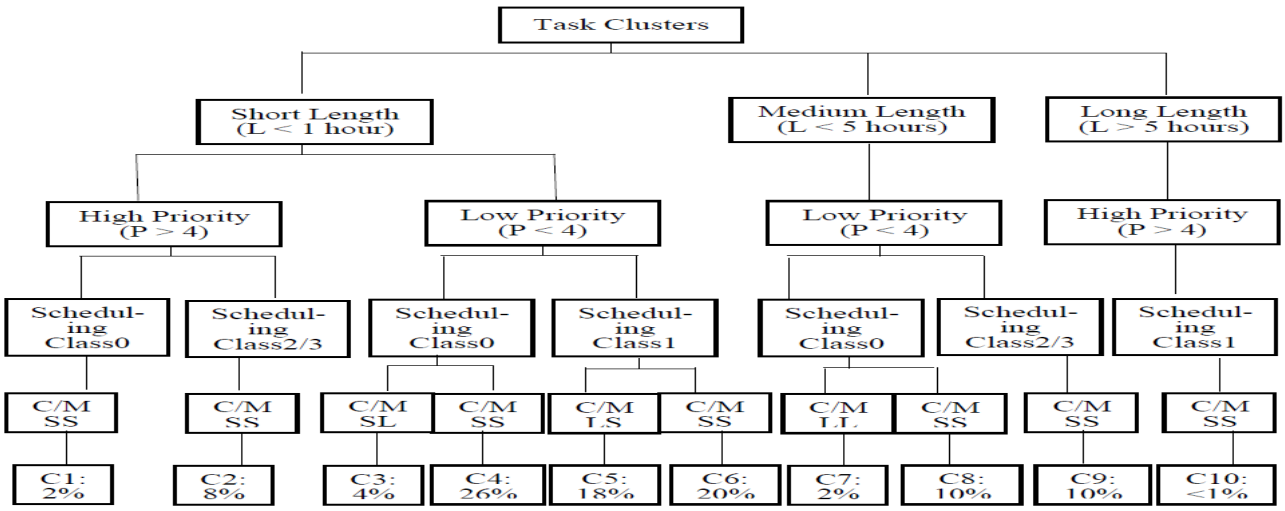

Figure 1 Task classification data statistics tree display

Cluster 1 and 2 are both short-time and high-priority tasks, and Cluster 2 has higher latency and longer average length. Cluster 3, 4, 5 and 6 are short-time and low-priority tasks, is 68%, and from Cluster 3 ~ Cluster6 priority and delay sensitivity have increased corresponding. Cluster 7, 8 and 9 are medium-time and low-priority tasks, Cluster-8 has the longest time, and Cluster-9 has a delay-sensitive the highest degree. A long-time, high-priority task has only one cluster, Cluster 10, with less latency sensitivity and at the same time as a longest-duration task, has a higher priority so that its resources are less likely to be preempted. This logic is implemented in the Google Cluster scheduler to avoid wasting resources from restarting long tasks during execution, which is consistent with the experimental results in this paper.

4. Resource Allocation Strategy Based on Task Classification

4.1 System Architecture

The system is based on a cloud computing environment specification where users request services from cloud service providers to perform tasks. The system consists of two main parts: users and providers, as shown in Figure 2. The user submitting a job to the cloud consists of a set of tasks that contain the basic resource requirements as well as implied QoS requirements. The provider component includes two models: Schedular and Data Center. The Schedular, as an interface between users and the cloud infrastructure, analyzes the requirements for submitting tasks and then allocates resources based on appropriate policies to reduce the energy consumption while meeting the task QoS requirements. The system resource scheduling model includes the task classification phase and the mapping phase. In the task classification phase, the task is divided into the 10 categories mentioned above, and the mapping phase will be based on the task classification, mapping different tasks to a reasonable virtual machine and mapping the virtual machine to a reasonable physical machine, which will be described in detail in the resource allocation strategy model in the next section.

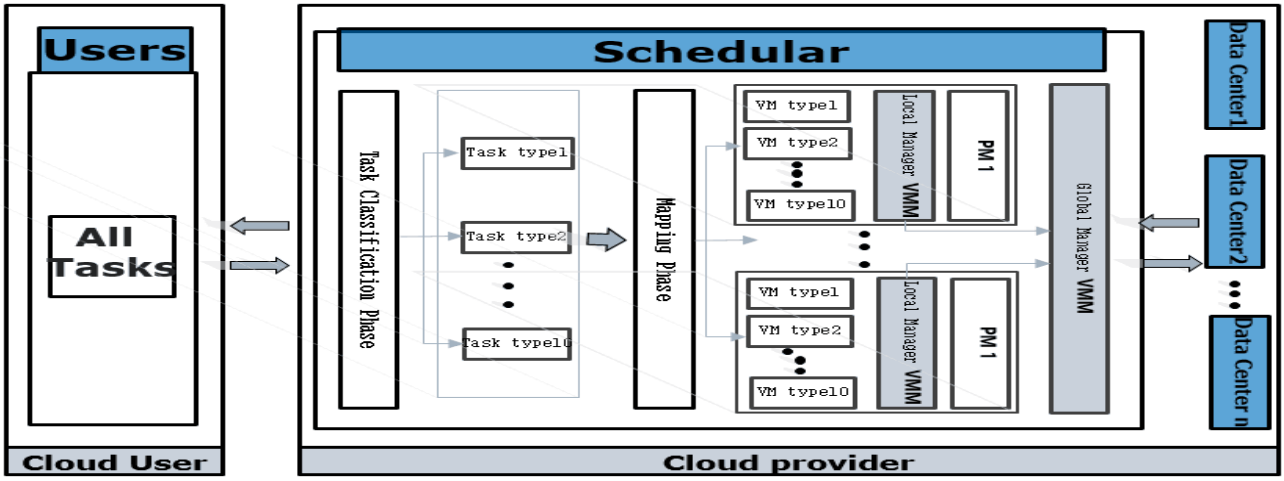

Figure 2 System architecture

4.2 Resource Allocation Strategy

In cloud computing, the problem of allocating resources is NP-hard. In resource allocation, the system S is modeled as a four-tuple (D, PM, VM, J). The D is a set of data centers, each element $D_d \in D$ represents a single data center in the system, we focus only on resource allocation strategies in a data center. PM is a set of physical machines in the data center, each element $PM_{i,d} \in PM$ represents a single PM_i in data center D_d. VM is a set of virtual machines associated with physical machines in the data center, each element $VM_{i,PM_i,d} \in VM$ represents a single VM_i on single PM_i in data center D_d. T is a set of tasks, each element $task_i \in T$ represents a single task.

Among these components, the CPU consumes the most amount of energy. Data center energy efficient resource allocation ensures QoS minimizes energy consumption at the same time. The energy consumed by PMs in data centers usually determined by the CPU, disk storage, memory, and network interfaces [10], Among these components, the CPU consumes the most amount of energy, and the energy consumption of the network element is very small which can be neglected. Hence, in this paper, three factors of CPU, disk and memory are selected for energy consumption evaluation. The PM_i represents a specific PM, and P_{cpu}, P_{memory}, P_{disk} represent the power consumed by the CPU,

memory, and storage disk, respectively, as shown in Equation (2).

$$PM_i^P = P_{cpu} + P_{memory} + P_{disk} \tag{2}$$

CPU as the main factor of energy consumption in physical machine, and its power consumption model of CPU is the sum of both CPU static power (P_{cpu_static}) and CPU dynamic power ($P_{cpu\_dynamic}$) [11], Equation (3) is used to compute the power consumed by CPU.

$$P_{cpu} = P_{cpu\_dynamic} + P_{cpu\_static} \tag{3}$$

Where P_{cpu_static} is a constant, say ω, and $P_{cpu\_dynamic}$ is given in (4).

$$P_{cpu\_dynamic} = ACV^2 f \tag{4}$$

Where A is an activity factor that accounts for frequency gates switching, C is the total capacitance at the gate outputs, V is the voltage of the CPU, and f is the operating frequency. Voltage V can be expressed as a linear function of frequency f, as shown in Equation (5).

$$V = \alpha f \tag{5}$$

Where α is a constant. All constants（ω, A, C） can be combined together in one constant β, Therefore, (3) (4) (5) can be rewritten as shown in (6):

$$P_{cpu} = \beta f^3 \tag{6}$$

That is, the power consumption of CPU is determined by its operating frequency.

If n is the total number of tasks, x_i indicates the status of the PM_i, where x_i is equal to 0 if the machine PM_i is off, and equal to 1 if it is on. Thus, the total energy consumed (TP_d) by the data center for all tasks is given by Equation (7).

$$TP_d = \sum_{i=1}^{n} x_i * PM_i^p \tag{7}$$

Obviously, the goal of resource allocation to reduce energy consumption is to minimize Equation (7), that is, to minimize the number of active physical machines and their CPU operating frequency (6). The resource allocation strategy based on task classification (CBRAS) proposed in this paper, According to the above task classification results, in order to ensure the task QoS requirements, the same duration task class is configured with the same virtual machine to reduce the cost of reconfiguring the resources when the short task has been executed in the long task execution process, and the prioritized configuration priority Level and latency sensitive task classes. In addition, to ensure low power consumption, the space-shared policy which in a VM associated with one or more cores and time-shared policy which in a core that holds two or more VMs [12] are used in combination to flexibly configure the types of virtual machines and configure different types of virtual machines for different tasks. Specifically, while ensuring the balance of memory utilization, high CPU utilization tasks such as Cluster5 and Cluster7 adopt the space-shared policy, and other types of tasks adopts the time-shared policy to minimize the CPU core frequency and thereby reduce energy consumption. After configuring reasonable types of VM for different types of tasks, VMs belonging to the same type of tasks are placed on different physical machines base on Multi Choice Knapsack Problem (MCKP), to avoid the resource preemption generated during the same types of tasks execution, thereby enhancing the compatibility

between workload tasks requirements and configuration resources, and further ensures the task performance to reduce the number of active physical machines.

5. Experimental Simulation and Result Analysis

5.1 Experiment Introduction

The CloudSim cloud computing simulation platform is used to simulate the data center in the paper, and the data used is the second day's data of the Google produces data center workload tracking data (version 2). The efficiency performance of the proposed CBRAS, the RB algorithm and the MBFD（Modi-fied Best Fit Decrease）[13] algorithm are compared under the same workload task number. As is shown in figure 4, the three server configurations are defined according to the Google data center and its host configuration during the study tracking. The hosts of the Google cluster are heterogeneous in CPU, memory and disk capacity, while the hosts with the same platform ID have the same architecture. There are three types of the platform in Google data center, in order to eliminate the placed constraints of the task, the platform including the most submitted task is chosen in the paper, and the task running on this platform is considered only.

Table 2 Server Configuration

Server Type	Number of Cores	Core Speed (GHz)	Memory (GB)	Disk (GB)	Pidel (W)	Pmax (W)
Type_1	32	1.6	8	1000	70.3	213
Type_2	32	1.6	16	1000		
Type_3	32	1.6	24	1000		

5.2 Experimental Results and Analysis

5.2.1 Total Energy Consumption of Data Center

The total energy consumption of the data center is determined by the sum of the Euclidean distance that between CPU、disk utilization and best state for all physical nodes. According to the experimental results in [10], the physical nodes as the best state when the CPU utilization is 70% and the disk utilization is 50%. The total energy consumption and the number of physical node enabled for each algorithm strategy are shown in Figure 3 and Figure 4.

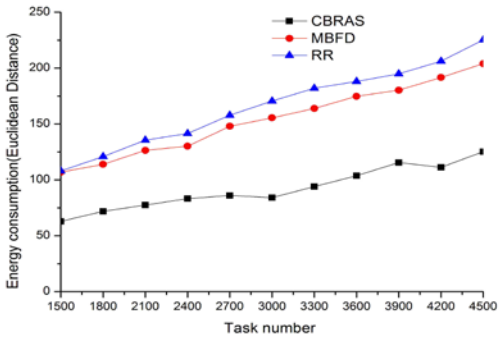

Figure 3 Euclidean distance of energy consumption

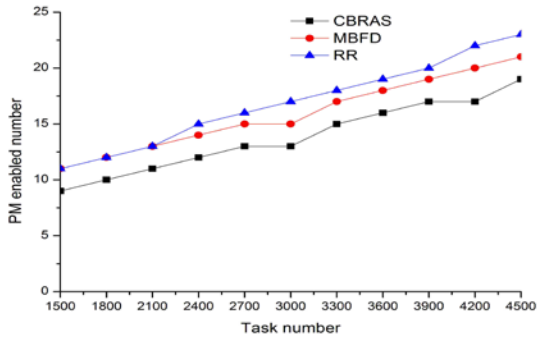

Figure 4 The average number of PM enabled

It is known from Figure 3 and Figure 4 that compared with RR and MBFD algorithm, the CBRAS has less the total Euclidean distance and the number of physical nodes, and with the number of tasks increases, the advantage of CBRAS is more obvious. Moreover, under the same number of physical nodes, CBRAS has smaller Euclidean distance. The reason for CBRAS having the lower energy consumption is that the traditional resource allocation algorithm ignores the heterogeneity of tasks and resources. It allocates the task to running resources simply when the resources are configured, which leads to the lack of full utilization of resources and higher energy consumption, While the CBRAS has considered the task characteristics when making resource coordination decisions. And it configures a

reasonable type of virtual machine for the same type of task, and place the different types of virtual machines into the same physical machine, which avoids resource preemption of the same resource demand type tasks. The CBRAS ensures that resources are fully utilized, and improves the compatibility between workload demands and configuration resources, thereby reducing energy consumption.

5.2.2 QoS Evaluation of Data Center

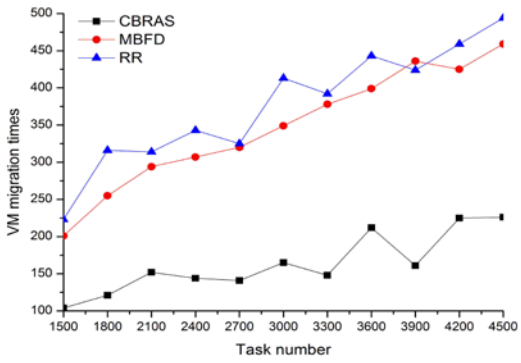

Figure 5 VM migration times comparing

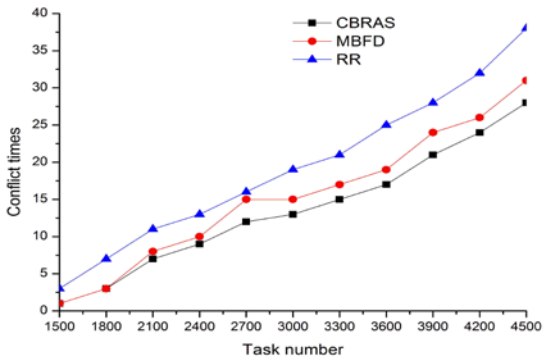

Figure 6 Conflicts times of VM migration comparing

Figure 5 and Figure 6 show the number of migration times and conflicts of the three resource allocation algorithms for the virtual machine under the same number of tasks. As can be seen from Figure 5 and Figure 6, CBRAS has less migration and conflict times than the RR and MBFD algorithm. Especially in the aspect of migration, CBRAS is significantly reduced. And with the increase of the number of tasks, the advantage of CBRAS is more obvious. It indicates that while ensuring high energy efficiency, the CBRAS can minimize the migration and conflict of virtual machines, run the cloud task more steadily, and guarantee the QoS. The result is because the traditional resource allocation algorithms only consider the task resource requirements but ignore tasks QoS requirements. The CBRAS not only takes the resource utilization ratio (CPU and memory) of different types of tasks into account, but also fully considers QoS requirements such as duration, priority, delay sensitivity. Configuring a reasonable type of VM for the same type of task can avoid the situation where a large number of short-time, low-priority, low-latency-sensitive tasks are allocated to the same location with long-term, high-priority tasks. In this way, it is possible to avoid the migration caused by low load after some short tasks are completed during the task period, and to avoid resource preemption of high-priority tasks for low-priority tasks. The rational allocation and utilization of resources can improve the compatibility between workload demands and configuration resources, reduce lowload or overload of physical nodes as well as the number of migration and conflict of virtual machine. Finally, to ensure the demand of QoS and improve data center energy efficiency.

6. Conclusion

The energy efficiency of heterogeneous data centers is studied in this paper, and a data center resource allocation scheme based on task classification is proposed. For data centers carrying a variety of heterogeneous workload tasks, we analyzed the Google data center workload traces data by the k-means algorithm. Considering the demand of mission resources, we take full account of user QoS requirements, including priority, delay sensitivity and duration index. We divide workload tasks into different task classes with similar resources and performance requirements, and propose CBRAS resource allocation strategy. The CBRAS configures the same types of tasks to the reasonable type of virtual machine to improve the compatibility between workload requirements and configuration resources. While ensuring the QoS requirements of different types of tasks, resources preemption of

tasks of the same resource requirement type is avoided, enabling energy-efficient resource allocation with low-power and QoS guarantees in the data center. The simulation results show that compared with the traditional RR and MBFD resource allocation algorithm, the CBRAS effectively improves the data center energy efficiency.

The next, we will improve the resource allocation strategy combined with task prediction model, and study online learning algorithm to replace the static task classification discussed in this paper, that is updating the task category dynamically to achieve a more reasonable resource allocation.

7. Acknowledgement

This paper was financially supported by Chongqing science and Technology Commission Project (Grant No:cstc2017jcyjAX0142， cstc2014jcyjA10051) and Science and technology project of Chongqing Municipal Education Commission (Grant No: KJ1400429, KJ1400407)

References

[1] Kaplan J M, Forrest W, Kindler N. Revolutionizing data center energy efficiency[R]. Technical report, McKinsey & Company, 2008.
[2] Reiss C, Tumanov A, Ganger G R, et al. Heterogeneity and dynamicity of clouds at scale:Google trace analysis[C]// ACM Symposium on Cloud Computing. ACM, 2012:1-13.
[3] Ren S, He Y, Xu F. Provably-efficient job scheduling for energy and fairness in geographically distributed data centers[C]//Distributed Computing Systems (ICDCS), 2012 IEEE 32nd International Conference on. IEEE, 2012: 22-31.
[4] Srikantaiah S, Kansal A, Zhao F. Energy aware consolidation for cloud computing[C]//Proceedings of the 2008 conference on Power aware computing and systems. 2008, 10: 1-5.
[5] Reiss C, Tumanov A, et al. Heterogeneity and dynamicity of clouds at scale: Google trace analysis[C]//Proceedings of the Third ACM Symposium on Cloud Computing. ACM, 2012: 7.
[6] Rasheduzzaman M, Islam M A, Islam T, et al. Task shape classification and workload characterization of google cluster trace[C]//Advance Computing Conference (IACC), 2014 IEEE International. IEEE, 2014: 893-898.
[7] Chen Y, Ganapathi A S, Griffith R, et al. Analysis and lessons from a publicly available google cluster trace[J]. EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2010-95, 2010, 94.
[8] Lopes R V, Menascé D. A taxonomy of job scheduling on distributed computing systems[J]. IEEE Transactions on Parallel and Distributed Systems, 2016, 27(12): 3412-3428.
[9] Googleclusterdata - traces of google workloads, http://code.google.com/p/googleclusterdata/, 2014.
[10] Guenter B, Jain N, Williams C. Managing cost, performance, and reliability tradeoffs for energy-aware server provisioning[C]//INFOCOM, 2011 Proceedings IEEE. IEEE, 2011: 1332-1340.
[11] Chaudhry M T, Ling T C, Manzoor A, et al. Thermal-aware scheduling in green data centers[J]. ACM Computing Surveys (CSUR), 2015, 47(3): 39.
[12] Calheiros R N, Ranjan R, Beloglazov A, et al. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms[J]. Software: Practice and experience, 2011, 41(1): 23-50.
[13] Beloglazov A, Abawajy J, Buyya R. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing[J]. Future generation computer systems, 2012, 28(5): 755-768.