2011 IEEE 4th International Conference on Cloud Computing

Energy-Aware Virtual Machine Dynamic Provision and Scheduling for Cloud Computing

Ching-Chi Lin
Institute of Information Science,
Academia Sinica
Taipei, Taiwan
deathsimon@iis.sinica.edu.tw

Pangfeng Liu
Dept. of CSIE,
Graduate Institute of Networking and Multimedia,
National Taiwan University
Taipei, Taiwan
pangfeng@csie.ntu.edu.tw

Jan-Jan Wu
Institute of Information Science,
Academia Sinica
Taipei, Taiwan
wuj@iis.sinica.edu.tw

***Abstract*—Power consumption is one of the most critical problems in data centers. One effective way to reduce power consumption is to consolidate the hosting workloads and shut down physical machines which become idle after consolidation. Server consolidation is a NP-hard problem. In this paper, a new algorithms Dynamic Round-Robin (DRR), is proposed for energy-aware virtual machine scheduling and consolidation. We compare this strategy with the GREEDY, ROUNDROBIN and POWERSAVE scheduling strategies implemented in the Eucalyptus Cloud system. Our experiment results show that the Dynamic Round-Robin algorithm reduce a significant amount of power consumption compared with the three strategies in Eucalyptus.**

***Keywords*-Cloud Computing, Data Center, Power Saving, Virtual Machine Consolidation.**

I. Introduction

Power conservation is an important issue for data centers. To reduce power consumption of data centers, one can consolidate computation workloads onto a subset of servers, and power off servers that become idle after consolidation. The key idea is to reduce "idle power", i.e., the power consumed by idle servers that do not have workload, so as to reduce the number of servers needed to stay power on.

In this paper we propose a new strategy for deploying virtual machines to servers and migrating virtual machines among servers: *Dynamic Round-Robin (DRR)*. *DRR* extends the traditional Round-Robin scheduling by incorporating the notion of *retirement* and *retirement time threshold*. In addition, VM migration is used to consolidate server utilization. The migration may be triggered every time a virtual machine finishes.

We compare these new strategies with the GREEDY, ROUNDROBIN, and POWERSAVE scheduling strategies implemented in the Eucalyptus Cloud system. Our experiment results show that Dynamic Round-Robin reduces 43.7% power consumption compared with the power consumption required by the *ROUNDROBIN* policy in Eucalyptus. Dynamic Round-Robin also saves about 2% more power than the POWERSAVE strategy in Eucalyptus.

II. Methodology

Virtual machine consolidation is an NP-hard problem. The problem can be stated as follows. Virtual machines with different hardware requirements, such as the number of cores and the size of memory, must be deployed to physical machines with fixed hardware capacity in a way that minimizes the number of physical machine used. This consolidation problem is harder than the bin packing problem, which is known to be NP-hard. It is easy to see that greedy method for bin packing problem, like first-fit method, can also be modified and applied to our consolidation problem.

A more complicated virtual machine consolidation problem is considered in this paper. In addition to resource requirements, each virtual machine also has an *arrival time* and an *execution time*. Unlike objects in the bin packing problem, every virtual machine only exists during its *execution period*, which starts at its arrival time, and lasts for its execution time. After a Virtual machine finishes it leave the system and the cores it used become idle before they are reassigned to other virtual machines. These idle cores may increase the actual number of physical machine required to run all virtual machines.

We propose a virtual machine deployment methods, *Dynamic Round-Robin(DRR)* for virtual machine consolidation. The objective of these methods is to minimize the number of physical machines used to run all virtual machines. This goal is very important because the number of physical machines used strongly affects total power consumption.

A. Dynamic Round-Robin

Dynamic Round-Robin is proposed as an extension to the Round-Robin method. Dynamic Round-Robin uses two rules to help consolidate virtual machines. The first rule is that if a virtual machine has finished and there are still other virtual machines hosting on the same physical machine, this physical machine will accept no more new virtual machine. Such physical machines are referred to as is in "retirement" state, meaning that when the rest of the virtual machines finishes, this physical machine can be shutdown.

978-0-7695-4460-1/11 $26.00 © 2011 IEEE
DOI 10.1109/CLOUD.2011.94

The second rule of *Dynamic Round-Robin* is that if a physical machine is in the "retirement" state for a sufficiently long period of time, instead of waiting for the hosting virtual machines to finish by itself, the physical machine will be forced to migrate the rest of virtual machines to other physical machines, and shutdown after the migration finishes. This waiting time threshold is denoted as "retirement threshold". A physical machine is in the retirement state but could not finish all virtual machines after the retire threshold will be forced to migrate its virtual machines and shutdown.

Our Dynamic Round-Robin method uses these two rules in order to consolidate virtual machines deployed by the original Round-Robin method. The first rule avoid adding extra virtual machines to a retiring physical machine so it can be shut down. The second rule speeds up the consolidation process and enables dynamic round-robin method to shutdown physical machines, so that it can reduce the number of physical machine used to run all virtual machines.

III. Comparison with Eucalyptus

Eucalyptus is a widely used open-source Cloud middleware. There are three scheduling options in Eucalyptus: *GREEDY*, *ROUNDROBIN* and *POWERSAVE*. *GREEDY* and *ROUNDROBIN* do not shut down machines nor put machines to sleep. *POWERSAVE* is equivalent to the power-saving first-fit method.

Power consumptions of the three scheduling algorithms in Eucalyptus are measured, and the results are compared with *Dynamic Round-Robin*. The number of powered-on physical machines in *GREEDY* and *ROUNDROBIN* is set to the number of physical machines required to process all incoming virtual machines.

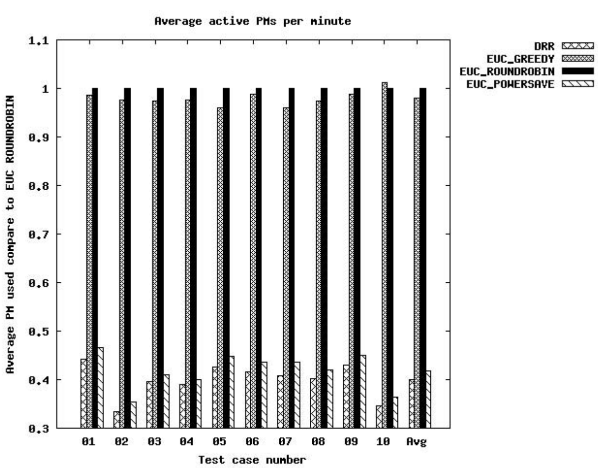

Figure 1. The normalized average percentage of powered-on physical machines with *ROUNDROBIN* as the baseline

Figure 1 and Figure 2 show average number of powered-on physical machines and average power consumption from six different scheduling methods, where *ROUNDROBIN*, the default method of Eucalyptus, is the baseline.

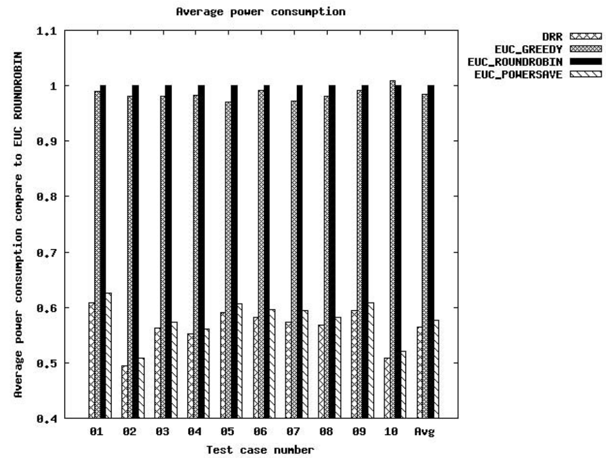

Figure 2. The normalized average percentage of power consumption with *ROUNDROBIN* as the baseline

From Figure 1 and Figure 2, it is clear that the number of powered-on physical machines and the power consumption reduce significantly using our algorithms, when compared with those from Eucalyptus. The Dynamic Round-Robin reduce 43.7% power consumption when compared with the power consumption required by the *ROUNDROBIN* policy. Furthermore, our algorithm saves 2% more power on average than the *POWERSAVE* scheduling policy in Eucalyptus.

IV. Conclusion

We propose a method, *Dynamic Round-Robin*, to deploy the virtual machines for power saving purpose. *Dynamic Round-Robin* is an extension from original Round-Robin, with two rules that help the virtual machines to consolidate. Test cases are generated, each with a bunch of virtual machines that have different arrival and execution time, as inputs. The average power consumption is used as performance metrics and the result of power-saving First-Fit is used as baseline.

Our algorithm is compared with the three scheduling algorithms in Eucalyptus. The experiment results show that our algorithm saves about 60% physical machines used and save about 43.7% power consumption when compared with the default scheduling algorithm ROUNDROBIN in Eucalyptus. When compared with the POWERSAVE scheduling in Eucalyptus, our algorithm requires 5% less physical machines used and save 2% more power consumption than POWERSAVE.