International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622
NATIONAL CONFERENCE on Developments, Advances & Trends in Engineering Sciences
(NCDATES- 09th & 10th January 2015)

RESEARCH ARTICLE OPEN ACCESS

Dynamic Heterogeneity-Aware Resource Provisioning in the Cloud

* Ch.vishali, **M.Jyothirmai
M.Tech scholar, CMR Engineering College
Asst.Professor, Dept. of CSE, CMR Engineering College, Hydearbad,
Email: jyothicmr2013@gmail.com

Abstract
Data centers consume tremendous amounts of energy in terms of power distribution and cooling. Dynamic capacity provisioning is a promising approach for reducing energy consumption by dynamically adjusting the number of active machines to match resource demands. However, despite extensive studies of the problem, existing solutions have not fully considered the heterogeneity of both workload and machine hardware found in production environments. In particular, production data centers often comprise heterogeneous machines with different capacities and energy consumption characteristics. Meanwhile, the production cloud workloads typically consist of diverse applications with different priorities, performance and resource requirements. Failure to consider the heterogeneity of both machines and workloads will lead to both sub-optimal energy-savings and long scheduling delays, due to incompatibility between workload requirements and the resources offered by the provisioned machines. To address this limitation, we present Harmony, a Heterogeneity-Aware dynamic capacity provisioning scheme for cloud data centers. Specifically, we first use the K-means clustering algorithm to divide workload into distinct task classes with similar characteristics in terms of resource and performance requirements. Then we present a technique that dynamically adjusting the number of machines to minimize total energy consumption and scheduling delay. Simulations using traces from a Google's compute cluster demonstrate Harmony can reduce energy by 28 percent compared to heterogeneity-oblivious solutions.

Introduction

DATA centers have recently gained significant popularity as a cost-effective platform for hosting large-scale service applications. While large data \centers enjoy economies of scale by amortizing long-term capital investments over a large number of machines, they also incur tremendous energy costs in terms of power distribution and cooling. For instance it has been reported that energy-related costs account for approximately 12 percent of overall data center expenditures [5]. For large companies like Google, a 3 percent reduction in energy cost can translate to over a million dollars in cost savings [22]. At the same time, governmental agencies continue to implement and regulations to promote energy-efficient computing [2]. As a result, reducing energy consumption has become a primary concern for today's data center operators. In recent years, there has been extensive research on improving data center energy efficiency [24], [29]. One promising technique that has received significant attention is dynamic capacity provisioning (DCP). The goal of this technique is to dynamically adjust the number of active machines in a data center in order to reduce energy consumption while meeting the service level objectives (SLOs) of workloads In the context of workload scheduling in data centers, a metric of particular importance is scheduling delay [20], [23], [25], [27], which is the time a request waits in the scheduling queue before it is scheduled on a machine. Task scheduling delay is a primary concern in data center environments for several reasons: (1) A user may need to immediately scale up an application to accommodate a surge in demand and hence requires the resource request to be satisfied as soon as possible.

(2) Even for lower-priority requests (e.g., background applications), long scheduling delay can lead to starvation, which can significantly hurt the performance of these applications. In practice, however, there is often a tradeoff between energy savings and scheduling delay. Even though turning off a large number of machines can achieve high energy savings, at the same time, it reduces service capacity and hence leads to high scheduling delay. Finally, the heterogeneity- aware DCP scheme should also take into account the reconfiguration costs associated with switching on and off individual machines. This is because frequently turning on and off a machine can cause the "wear-and-tear" effect [12], [19] that reduces the machine lifetime.

Despite the fact that a large number of DCP schemes have been proposed in the literature in recent years, a key challenge that often has been overlooked or considered difficult to address is heterogeneity, which is prevalent in production cloud data centers [23]. We summarize the types of heterogeneity found in production environments as follows: Machine heterogeneity. Production data centers often comprise several types of machines from multiple generations[25]. They have heterogeneous processor architectures and speeds, hardware features, memory and disk capacities. Consequently, they have different runtime energy consumption rates.

Literature Survey

2.1 WORKLOAD ANALYSIS

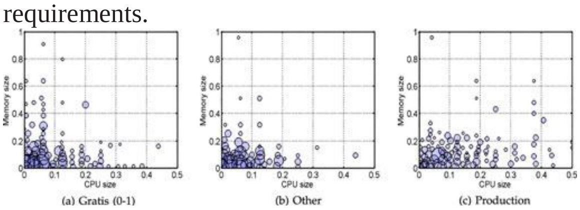
To understand the heterogeneity in production cloud data centers, we have conducted an analysis of workload traces for one of Google's production compute clusters [4]1 consisting of approximately 12;000 machines. The workload traces contain scheduling events, resource demand and usage records for a total of 672;003 jobs and 25;462;157 tasks over a time span of 29 days. Specifically, a job is an application that consists of one or more tasks. Each task is scheduled on a single physical machine. When a job is submitted, the user can specify the maximum allowed resource demand for each task in terms of required CPU and memory size.

2.2 Understanding Machine Heterogeneity

The traces also provide information about the types of machines used in the cluster. A machine is characterized by its capacity in terms of CPU, memory and disk size as well as a platform ID, which identifies the micro-architecture (e.g., vendor name and chipset version) and memory technology (e.g., DDR or DDR2) of the machine. Similar to tasks, machine capacities are normalized such that the largest machine has a capacity equal to 1. Fig. 5 shows the different types of machines and their characteristics (capacity and platform ID (PFID)). We found 10 types of machines where more than 50 and 30 percent of the machines belong to machine types 1 and 2, respectively. On the other hand, machine types 3 and 4 have around 1;000 machines each. The remaining machine types (5 to 10) constitute less than 100 machines. Unfortunately, the traces do not provide detailed information about hardware specifications, however, it is likely that this heterogeneity translates into different energy consumption models.

2.3 Understanding Task Heterogeneity

In order to analyze the workload heterogeneity, we plotted tasks' requirements and their durations for the three priority groups. the CPU and memory size of tasks belonging to each priority group. The coordinates of each point in these figures correspond to a combination of CPU and memory requirements. Radius of each circle is logarithmic in number of tasks within its proximity. It can be seen that most of the tasks have low resource requirements.



Task size analysis

System Design

3.1 Existing System

Despite extensive studies of the problem, existing solutions have not fully considered the heterogeneity of both workload and machine hardware found in production environments. it can be easily integrated with existing scheduling algorithms, variants of first-fit and best-fit algorithms and Open source platforms such as Eucalyptus can adopt this mechanism by changing the scheduling policy to weight round-robin first fit and weight round-robin best fit, respectively. The main benefit of CBP is its simplicity and practicality for deployment in existing systems . The existing work on this topic has not addressed a key challenge, which is the heterogeneity of workloads and physical machines.

3.2 Proposed System

The fact that a large number of DCP schemes have been proposed in the literature in recent years, a key challenge that often has been overlooked or considered difficult to address is heterogeneity, which is prevalent in production cloud data centers. present the formulation of the heterogeneity-aware DCP, and provide two technical solutions . Discusses the deployment of Harmony in practice. Finally, we evaluate our proposed system using Google workload traces. as both scheduler design and capacity upgrade require a careful understanding of the workload characteristics in terms of arrival rate, requirements, and duration .We have analyzed the workload of a Google compute cluster, and proposed an approach to task classification using k-means clustering .

SOLUTION TECHNIQUES

Realizing that directly solving DCP is not possible, in this section we present two fast heuristics for solving DCP. Both techniques rely on solving the integer-relaxation of DCP (i.e., relaxing the constraints that variables must take integer values) called DCP _ RELAX, which is much easier to solve than DCP. Once the solution for DCP _ RELAX is obtained, one of our solution techniques called containerbased provisioning (CBP) directly rounds the numbers of machines to the nearest integer values and use these values for capacity provisioning. On the other hand, the containerbased scheduling (CBS) technique attempts to find a feasible placement of containers in physical machines, and use containers for run-task scheduling. In both cases, the capacity provisioning module first adjusts the number of active machines, and informs the scheduler about how tasks should be assigned to each type of machines.

Summary

The above analysis suggests that while the benefit of dynamic capacity provisioning is apparent for production data center environments, designing an effective and dynamic capacity provisioning scheme is challenging, as it involves finding a satisfactory compromise between energy savings and scheduling delay with consideration to the heterogeneous characteristics of both machines and workload.

Conclusion

Dynamic capacity provisioning has become a promising solution for reducing energy consumption in data centers in recent years. However, existing work on this topic has not addressed a key challenge, which is the heterogeneity of workloads and physical machines. In this paper, we first provide a characterization of both workload and machine heterogeneity found in one of Google's production compute clusters. Then we present Harmony, a heterogeneity-aware framework that dynamically adjusts the number of machines to strike a balance between energy savings and scheduling delay, while considering the reconfiguration cost. Through experiments using Google workload traces, we found Harmony yields large energy savings while significantly improving task scheduling delay

ACKNOWLEDGMENTS

This work was supported in part by the Natural Science and Engineering Council of Canada (NSERC) under the Smart Applications on Virtual Infrastructure(SAVI) Research Network, and in part by a Google Faculty ResearchAward.

REFERENCES

[1] Amazon Elastic Computing Cloud, http://aws.amazon.com/ ec2/, 2013.
[2] Energy Star Computer Server Qualified Product List, energystar. gov/ia/products/prod_lists/enterprise_servers_ prod_list.xls,2014.
[3] Eucalyptus community, http://open.eucalyptus.com/, 2014.
[4] Googleclusterdata - traces of google workloads, http://code. google.com/p/googleclusterdata/, 2014.
[5] Technology research - Gartner Inc, www.gartner.com, 2014.
[6] U.S. Energy Information Adminstration, http://www.eia.gov/,2014.
[7] G. Ananthanarayanan, A. Ghodsi, S. Shenker, and I. Stoica, "Effective Straggler Mitigation: Attack of the Clones," Proc. 10th USENIX Conf. Networked Systems Design and Implementation (NSDI), 2013.
[8] R. Boutaba, L. Cheng, and Q. Zhang, "On Cloud Computational Models and the Heterogeneity Challenge," J. Internet Services and Applications, vol. 3, pp. 77-86, 2012.
[9] G.E.P. Box, G.M. Jenkins, and G.C. Reinsel, Time Series Analysis, Forecasting, and Control. Third ed., Prentice-Hall, 1994.
[10] S. Boyd et al., Convex Optimization. Cambridge Univ. Press, 2004.
[11] C. Chekuri and S. Khanna, "On Multi-Dimensional Packing Problems," Proc. Symp. Discrete Algorithms, 1999.
[12] Y. Chen, A. Das, W. Qin, A. Sivasubramaniam, Q. Wang, and N. Gautam, "Managing Server Energy and Operational Costs in Hosting Centers," ACM SIGMETRICS Performance Evaluation Rev., vol. 33, pp. 303-314, 2005.
[13] Y. Chen et al., "Analysis and Lessons from a Publicly Available Google Cluster Trace," Technical Report UCB/EECS-2010-95, 2010.
[14] J. Diaz et al., "A Guide to Concentration Bounds," Handbook on Randomized Computing. Spring