Sustainable Computing: Informatics and Systems 4 (2014) 1–9

Contents lists available at ScienceDirect

Sustainable Computing: Informatics and Systems

journal homepage: www.elsevier.com/locate/suscom

Performance bounded energy efficient virtual machine allocation in the global cloud

Patrick Raycroft[a], Ryan Jansen[a], Mateusz Jarus[b], Paul R. Brenner[a,*]

[a] *University of Notre Dame Center for Research Computing, Notre Dame, IN, USA*
[b] *Applications Department Poznań Supercomputing and Networking Center, Poznań, Poland*

ARTICLE INFO

Article history:
Received 8 February 2013
Received in revised form 26 June 2013
Accepted 1 July 2013

Keywords:
Cloud
Virtual machine
Dynamic allocation
Network performance

ABSTRACT

Reducing energy consumption is a critical step in lowering data center operating costs for various institutions. As such, with the growing popularity of cloud computing, it is necessary to examine various methods by which energy consumption in cloud environments can be reduced. We analyze the effects of global virtual machine allocation on energy consumption, using a variety of real-world policies and a realistic testing scenario. We found that by using an allocation policy designed to minimize energy, total energy consumption could be reduced by up to 14%, and total monetary energy costs could be reduced by up to 26%. Further, we have begun performance qualification of our energy cost driven allocation policies through network capability tests. Our results indicate that performance and IaaS provider implementation costs have a significant influence on selection of optimal virtual machine allocation policies.

1. Introduction

As adoption of virtualization services increases, cloud computing platforms are becoming increasingly popular. Demand for existing cloud infrastructures such as Amazon's Elastic Compute Cloud [1] is steadily rising [2], and the use of private compute clouds is becoming more popular among a variety of institutions [3].

While virtualization has in many cases facilitated IT infrastructure consolidation for individual organizations, expanding demand for IT services via cloud technology drives growth. As virtualization on various cloud platforms becomes more prevalent, the rising number of virtual machine requests in any given cloud necessitates a proportionally increasing number of physical host servers to fulfill them. As such, data center sizes are expanding, generating a growing concern over energy consumption and electricity costs among businesses and hosting providers alike.

The energy cost incurred in running a data center has been steadily rising for years. Data center energy costs in the United States accounted for nearly 1.5% of all electricity costs in 2006, reaching up to approximately $4.5 billion per year, and trend data estimates that this cost will jump to an annual cost of $7.6 billion for 2011 [4]. According to recent report by Koomey [5], the growth in electricity used by data centers worldwide was in fact lower than previously predicted. It was mainly affected by two factors–slowdown of growth in the installed base of servers because of virtualization and the financial crisis of 2008 with its associated economic inhibition. Nevertheless, the energy issue is still an important concern as the high density computing facilities keep expanding and, as a result, require more and more power.

With this in mind, it is worthwhile to attempt to minimize energy consumption through any means available. In this paper, we will examine various cloud allocation policies used to match virtual machines to physical hosts in a cloud environment. We will simulate each policy independently and analyze its effectiveness in a number of categories, with a focus on energy consumption.

Related work by Garg et al. [6] focuses on deploying high-performance computing (HPC) services across a cluster while minimizing carbon emissions, using a combination of minimization algorithms and CPU voltage scaling. Additionally, Kim et al. [7] has focused on developing a similar system that combines scheduling and CPU voltage scaling to achieve reduced energy consumption across a cluster. Such methods reduce energy costs, but their focus is on power consumption at the CPU level as opposed to the cluster level. More akin to our scheduling analysis is a system developed by Chase et al. [8] in which services bid on host machines and are scheduled to minimize energy costs, while properly allocating services to handle varying web loads. Unlike our research, their approach relies on an outside scheduling framework. Further, work by Mazzuco et al. [9] develops dynamic scheduling of servers local to one data center to maximize user experience while minimizing the energy costs of cloud providers. Their work differs from ours in

* Corresponding author at: 111 Information Technology Center, Notre Dame, IN 46556, USA. Tel.: +1 574 631 9286.
E-mail addresses: praycrof@nd.edu (P. Raycroft), ryan.alan.jansen@gmail.com (R. Jansen), jarus@man.poznan.pl (M. Jarus), pbrenne1@nd.edu, paul.raymond.brenner@gmail.com (P.R. Brenner).

2210-5379/$ – see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.suscom.2013.07.001

that it addresses the scheduling policies of local physical servers as opposed to globally distributed VMs. In addition, Beloglazov et al. [10] focus on virtual machine reallocation by taking into account Quality of Service and cost minimization. However, their work concentrates solely on a single data center, contrasting our global methodology. Finally, work by Aikema et al. [11] takes a global migration approach, similar to our migration policies originally discussed by Jansen et al. [12]. However, we differ by considering multiple experimental network performance tests as well as energy cost optimization.

A different approach, focusing on optimizing the allocation process, is described by Srikantaiah et al. [13]. Their strategy involves modeling the cloud as a bin packing problem, with physical hosts as bins and virtual machines as objects to fit inside of them. Using this model, they attempt to consolidate the virtual machines to as few hosts as possible, in an effort to minimize overall energy usage. As we will see later on, this approach is akin to (although much more advanced than) the *Packing* allocation policy defined in our simulation.

Zheng et al. [14] created an optimal energy-aware load dispatching model to minimize the electricity and network costs for Online Service Providers. They selected end-to-end response time as the metric of performance, which consists of network delay and response time inside an Internet Data Center. Geographic distance was used as a rough measure of network latency. The round trip time for a request from user group to data center is a linear function of its distance. However, this approach is not accurate. As our experiments indicate, the latency and throughput of a network connection cannot be always calculated in this simple manner. Moreover, network bandwidth changes throughout the day, being affected by network congestion and load on servers. For this reason our simulation is based on real results from experiments measuring network efficiency between selected data centers.

The allocation policies presented in this paper are either already available on two popular cloud platforms (OpenNebula [15] and Eucalyptus [16]), or they are straightforward to implement using either platform's scheduling policy syntax. In this paper, we will attempt to analyze how various allocation policies affect energy consumption, as well as CPU load and overall energy costs, in a realistic environment based on dynamic website loads.

Of the seven allocation policies we tested, four are currently available by default in existing open-source cloud platforms. The four existing policies tested include Round Robin, Striping, Packing, and free-CPU-count-based Load Balancing. One of the remaining three, ratio-based Load Balancing is a variation on the original count-based load balancing, and the other two, the Watts per Core and Cost per Core policies, are experimental, intended to minimize overall data center energy consumption and energy costs respectively. These policies are described in depth by Jansen et al. [12] and are summarized in Section 2.

2. Simulation scenario

Via our simulation, we tested seven different cloud allocation policies: Round Robin, Striping, Packing, Load Balancing (free CPU count), Load Balancing (free CPU ratio), Watts per Core, and Cost per Core.

- *Round Robin*: This allocation policy iterates sequentially through available hosts. When a host is found that has sufficient resources, the VM is matched to the host. On the next iteration, the policy starts its iterations where it previously left off.
- *Striping*: This policy first discards all hosts that do not have sufficient available resources to host the machine. It then selects from the remaining hosts the one that is currently hosting the fewest number of VMs and matches the virtual machine to that host.
- *Packing*: The Packing policy is the opposite of Striping. After discarding all hosts similarly to Striping, it selects from the remaining hosts the one that is currently hosting the greatest number of VMs and matches the virtual machine to that host.
- *Load Balancing (free CPU count)*: Similarly to the other policies, the count-based Load Balancing policy first discards all hosts that do not have sufficient available resources. From the remaining hosts, it then selects the one with the greatest number of free CPU cores and matches the virtual machine to that host.
- *Load Balancing (free CPU ratio)*: This policy is similar to the count-based Load Balancing; however, it instead selects the host with the greatest ratio of free CPU cores to allocated CPU cores and matches the virtual machine to that host.
- *Watts per Core*: From the pool of hosts that have sufficient available resources, this policy selects the host that would result in using the least additional wattage per CPU core if chosen, based on each host's power supply, and matches the virtual machine to that host.
- *Cost per Core*: This policy is similar to the Watts per Core policy above; however, it instead selects the host that would result in using the least additional cost per CPU core if chosen, based on each host's power supply and electricity costs, and matches the virtual machine to that host.

Table 1
Physical host specifications.

	Cluster			
Physical server	US East server counts	US West server counts	Asia server counts	Europe server counts
server.A1	0	24	0	0
server.A2	0	8	0	0
server.B1	24	0	0	0
server.B2	16	0	0	0
server.C1	0	0	8	0
server.C2	0	0	8	0
server.D1	0	0	0	16
server.D2	0	0	0	16
server.D3	0	0	0	16

Our simulation scenario attempts to accurately simulate a large-scale website–the social media site Reddit.com. Reddit.com [17] shifted their entire infrastructure to Amazon EC2 virtual machine instances, and, as of February, 2011, the site serves up to 1 billion users monthly [18].

In the scenario, we define four clusters of physical hosts, each representing a geographical location around the world as well as an existing Amazon EC2 data center [19]. As mentioned above, the website is hosted entirely on a set of virtual machines, which will be distributed among the clusters as necessary to deal with dynamic server loads. The load structure was chosen specifically to imitate typical web server loads based on the time of day at the different geographical locations.

In this section, we will state the specification of each of our physical hosts, the requirements of each of our virtual machines, and the website load scheme used in our simulation.

2.1. Physical hosts

The physical hosts in our simulation are based off of commodity servers available from IBM. Server specifications are based off of the specifications and power requirements provided by IBM's Power Configurator tool [20]. The different servers used in our simulation are defined in Table 1.