Computers and Electrical Engineering 000 (2016) 1–11

Contents lists available at ScienceDirect

Computers and Electrical Engineering

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

Responsive and efficient provisioning for multimedia applications

Md. Mahfuzur Rahman*, Peter Graham

Deptartment of Computer Science, University of Manitoba, Canada

ARTICLE INFO

Article history:
Received 19 September 2015
Revised 10 April 2016
Accepted 11 April 2016
Available online xxx

Keywords:
Resource provisioning
eHealth application
Cloud
Optimization

ABSTRACT

Multimedia applications (including those in eHealth scenarios) can require on-demand and urgent resource provisioning in cloud environments. Provisioning in clouds, the virtual machines (VMs) assignment to physical machines (PMs), is critical to obtaining efficiency for the cloud provider and also to ensuring the overall satisfaction of cloud users. Provisioning algorithms are often divided into batch and online algorithms where the former gather VM allocation requests and then efficiently pack them as a group and the latter place VMs immediately upon receiving requests. The underlying tradeoff is one of efficiency in packing (leading to lower cost for the cloud provider) vs. greater responsiveness to requests (which is important to some cloud applications). In this paper, we propose and show the effectiveness of a new static hybrid algorithm for provisioning that groups and optimizes normal VM requests but immediately places more urgent requests.

1. Introduction

Many health care applications like radio image archiving, health system management, patient records, etc. have successfully been migrated into “the cloud” [1,2]. Considering the benefits offered by cloud computing and as the successes in this area grow a similar transition can easily be foreseen for more demanding eHealth applications (e.g. tele-surgery, emergency video surveillance [3,4], remote diagnosis, etc.) too. Such applications introduce further challenges to successful cloud-oriented implementation. The reason is primarily due to their load varying, on-demand characteristics. Unfortunately, these challenges are unlikely to be satisfied through changes at the application-level only. To effectively and efficiently host these types of applications in a cloud, we propose some enhancements to cloud resource provisioning itself.

Multimedia eHealth applications impose different provisioning requirements (for example, sometimes urgent, sometimes load-varying, etc.) [5]. A very large amount of hardware resources (memory, CPU, network capacity, etc.) are available in the data centers in a cloud. Those resources are managed in the form of compute nodes and using those connected computing nodes a massive computing platform is then made available to cloud clients (to run their applications). The challenge to hosting such applications in a cloud is that provisioning strategies are often inadequate to meet their requirements.

Much cloud-based work is still typified by long-running VMs executing OLTP (OnLine Transaction Processing) and other standard workloads but there are also an increasing number of shorter duration tasks for which clouds are now being used, “on-demand”. Packing well-characterized, long-running virtual machines (VMs) can be done very efficiently since more time may be devoted to the packing process (to optimize power consumption, resource costs and the like). As the VM mix

* Corresponding author. Tel.: +12042303851.
E-mail addresses: mahfuz@cs.umanitoba.ca (Md.M. Rahman), Peter.Graham@umanitoba.ca (P. Graham).

http://dx.doi.org/10.1016/j.compeleceng.2016.04.011
0045-7906/© 2016 Elsevier Ltd. All rights reserved.

becomes less regular and as VMs run for shorter duration, however, packing decisions need to be made quickly and provisioning must carefully consider resources (on physical machines) freed by the completion of VMs. In this paper we present a static VM provisioning strategy that addresses these challenges and which fills part of a spectrum of cloud provisioning techniques mixing static and dynamic (i.e. live migration-based) provisioning [6] that we believe will need to be used concurrently to ensure effective resource utilization as well as quick response to evolving cloud workloads.

The focus of the algorithm introduced in this paper is on better handling of VMs with short life times and on being responsive to urgent VM requests (those that require early/immediate start times which include a number of e-health related applications). Providing better support in these areas will increase the appeal of cloud environments for a broader range of user applications. As such the paper is primarily concerned with making provisioning decisions quickly and effectively. To accomplish this, we use a hybrid between online and batch provisioning techniques. While the ability to adjust placement decisions after VMs have begun execution through live migration (e.g. Clark et al. [7]) is undoubtedly important, it is not considered in this paper.

Underlying cloud provisioning is a fundamental packing problem – assigning VMs with certain requirements to physical machines (PMs) with certain resources to minimize wasted resources within PMs and to minimize the required number of PMs. With a single resource type, provisioning becomes an instance of the bin packing problem. VM provisioning is therefore NP-hard so heuristics methods are required that will give solutions that are hopefully good but without any guarantee of optimality. Considering multiple resource types (e.g. memory, CPU, and storage requirements) add complexity to the problem as it is harder to concurrently optimize the packing for multiple criteria. Further, packing decisions typically need to be done reasonably quickly so elaborate heuristics designed to get very close to the optimal packing in many cases are commonly impractical. Thus, development of new provisioning algorithms needs to be a process of refining techniques to address new requirements and/or to quickly provide better packing solutions in specific cases.

In this paper, Section 2 briefly reviews related work. The problem to be solved is described clearly in Section 3. Section 4 presents the proposed algorithm to solve the problem. The algorithm is assessed in Section 5. Finally, our conclusions and a discussion of our future work directions are presented in Section 6.

2. Related work

From the cloud provider's perspective, a key requirement is to ensure efficient resource utilization and to enable efficient resource provisioning [8–11]. Based on current/expected load, cloud clients need to allocate the memory space, computational resources, network bandwidth, disk storage, etc. and pay accordingly. Based on the clients' requirements, the necessary resources are allocated by the cloud provider and access is provided to the client in the form of VMs with associated resources. VM provisioning ensures necessary application execution, proactive failure handling, and efficient load management.

The cloud client requests VMs based on agreed-to Service Level Agreements (SLAs) and the cloud provider allocates required hardware resources accordingly using static VM provisioning. Such static provisioning includes simple techniques such as Greedy, which packs VMs onto the fewest number of physical machines, or Round Robin which spreads VMs evenly over the physical machines. In a cloud, VMs may also be migrated to other physical machines using dynamic provisioning techniques on the fly, if needed, though dynamic provisioning is not discussed in this paper.

The goal of static algorithms is usually to pack the VMs into a minimum number of physical machines (a form of bin packing). Such static provisioning algorithms are commonly sub-divided into online and batch algorithms. In online algorithms, requests arrive one at time and are immediately placed without considering subsequent requests. With batch algorithms, requests are collected and packed as a group. Simple online algorithms besides Greedy and Round Robin include First Fit (FF), Best Fit (BF), etc. When batches of requests may be collected, significant improvements in packing quality can be achieved. For example, the First Fit Decreasing (FFD) and Best Fit Decreasing (BFD) algorithms offer denser packing of VMs into PMs than their counterparts (FF and BF, respectively). In FFD, VMs are sorted according to their resource requirements and are placed in physical machines following the sorted order. FFD starts with a new physical machine and places the VMs, in order, into that machine and as many additional physical machines as are needed.

To determine the hardware resources needed for static provisioning, "sizing" of the required hardware resources is important. *Joint* VM sizing, introduced by Meng et al. [12], can aggregate the total required capacity of multiple VMs to be co-hosted. When a physical machine hosts *multiple* VMs, joint VM sizing becomes important to select and co-host the appropriate VMs in a particular physical machine. In static VM provisioning, a single VM will normally not fully utilize the allocated resources and in that case the utilization of the hardware resources can be improved using VM mutliplexing. Joint VM sizing can also help to allocate the unused resources of one VM to other VM(s). To determine compatible VMs where the VM's SLAs are not violated and to consolidate them on a single physical machine, a workload forecasting model was also developed by Meng et al. as part of their work. A model to monitor cloud client's satisfaction on SLAs and other related criteria was developed by Fito et al. [13].

Resource requirements are not the only characteristic that can be used to determine packing. Calcavecchia et al. [14] describe an algorithm where demand risk scores are used in making placement or migration decisions. The idea is to minimize the risk of placing a VM on a physical machine where it will perform poorly in the future (e.g. due to anticipated variance in host load). VM requests are therefore allocated to hosts with low risk scores. This is done in an online fashion. Other