Computer Networks 91 (2015) 453–470

Contents lists available at ScienceDirect

Computer Networks

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

Modeling of the resource allocation in cloud computing centers

Shahin Vakilinia*, Mustafa Mehmet Ali, Dongyu Qiu

Department of Electrical and Computer Engineering, Concordia University, 7141 Rue Sherbrooke West, QC H3G 1M8, Canada

ARTICLE INFO

Article history:
Received 25 September 2014
Revised 12 July 2015
Accepted 21 August 2015
Available online 8 September 2015

Keywords:
Cloud computing
Queuing systems
Resource allocation
Markov process

ABSTRACT

Cloud computing offers on-demand network access to the computing resources through virtualization. This paradigm shifts the computer resources to the cloud, which results in cost savings as the users leasing instead of owning these resources. Clouds will also provide power constrained mobile users accessibility to the computing resources. In this paper, we develop performance models of these systems. We assume that jobs arrive to the system according to a Poisson process and they may have quite general service time distributions. Each job may consist of multiple numbers of tasks with each task requiring a virtual machine (VM) for its execution. The size of a job is determined by the number of its tasks, which may be a constant or a variable. The jobs with variable sizes may generate new tasks during their service times. In the case of constant job size, we allow different classes of jobs, with each class being determined through their arrival and service rates and number of tasks in a job. In the variable case a job generates randomly new tasks during its service time. The latter requires dynamic assignment of VMs to a job, which will be needed in providing service to mobile users. We model the systems with both constant and variable size jobs using birth–death processes. In the case of constant job size, we determined joint probability distribution of the number of jobs from each class in the system, job blocking probabilities and distribution of the utilization of resources for systems with both homogeneous and heterogeneous types of VMs. We have also analyzed tradeoffs for turning idle servers off for power saving. In the case of variable job sizes, we have determined distribution of the number of jobs in the system and average service time of a job for systems with both infinite and finite amount of resources. We have presented numerical results and any approximations are verified by simulation. The results of the paper may be used in the dimensioning of cloud computing centers.

1. Introduction

Reduced costs of processing and storage technologies brought about rapid growth of computing resources industry. Recently, a new computing paradigm called cloud computing emerged which provides on-demand network access to the computing resources through virtualization. This paradigm offers cost savings because users lease the computing resources from a service provider when needed instead of owning them. Further, clouds will provide mobile users access to computing resources, which is referred to as mobile cloud computing [1]. This is very important as mobile devices are becoming primary computing platform to many users and they have limited processing power and battery life. Cloud computing enables dynamic sharing of the computing resources among the users. A service level agreement (SLA) specifies the quality of service (QoS) to be provided to the user in terms of various performance parameters such as throughput, reliability, blocking probability and response time. Cloud computing services may be classified into three types as Infrastructure-as-a-service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS). IaaS refers to providing hardware equipment such as CPU,

* Corresponding author. Tel.: +15148365580.
E-mail addresses: s_vakili@ece.concordia.ca, shahin.vakilinia@gmail.com (S. Vakilinia), Mustafa@ece.concordia.ca (M.M. Ali), dongyu@ece.concordia.ca (D. Qiu).

http://dx.doi.org/10.1016/j.comnet.2015.08.030
1389-1286/

memory and storage as a service, PaaS refers to providing platforms such as software development frameworks, operating systems or multi-tenant application supports as a service and SaaS providing software and applications as a service. In this paper, we only consider cloud computing centers that provide the IaaS service through leasing of virtual machines (VMs) to the users [2].

In general, the topology of a cloud computing center is hierarchical with racks containing a fixed number of blade servers. A blade server contains a number of processors each one consisting of several processing cores. The processing cores, memory and storage space are configured into VMs. VMs may be homogeneous or heterogeneous. In the first case, VMs have the same number of CPUs, memory and storage sizes, while in the second case, there may be different VM types which may differ from each other in number of CPUs, memory and storage sizes [1].

Jobs entering the system may demand different types of services. However most of them require parallel data analysis [3]. This is the main reason for the recent development of MapReduce Programming model [4]. This model relies on parallel processing with a sequential functional approach. Job fragments are executed in parallel to speed up processing of the jobs. MapReduce has usually three phases as fan out, map and reduce. Applications such as Apache Hadoop [5] and platforms such as Pig [5] implement the MapReduce programming model.

This model also applies to bag-of-tasks (BoTs) where a job consists of parallel and sequential tasks. Number of tasks executing in mapping phase will be larger than fan out and reduce phases, thus dynamic resource allocation will benefit this programming model.

Mobile devices such as smartphones and tablet PC are increasingly becoming part of everyday life. These devices provide many capabilities such as GPS, WiFi and cameras. As a result, developers are building more and more complex mobile applications such as gaming, navigation, video editing, etc. for these devices. Though hardware of these devices is becoming more powerful, they are not able to keep up with the computational, storage and energy demands of more complex applications and they have short battery life [6]. Mobile cloud computing (MCC) is a derivative of cloud computing and its goal is to serve mobile users [7]. The MCC is expected to provide on-demand processing power and storage for mobile users in the cloud. This will enable mobile devices to offload their work to the cloud at a finer granularity [8]. Khan et al. [9] provides a survey of the proposed application models for mobile cloud computing. The various application models differ from each other in terms of design and objectives. Depending on the workload of the mobile device, number of VMs assigned to it will be dynamically changing. In [10], a method level offloading to the cloud has been proposed in order to take advantage of the parallelism in the application. In the experiments reported in [10], the average time to resume a VM from the pause state is around 300 ms while from the powered-off state is 32 s.

In this paper, we will consider various cloud computing models that may be used in the dimensioning of these systems. We will consider systems with both homogenous and heterogeneous types of VMs.

We assume that the job arrivals will be according to a Poisson process. A job may consist of multiple numbers of tasks and execution of each task requires a VM. The size of a job in number of tasks may be a constant or may vary dynamically during its service time. In the case of constant job size, the size is chosen from a discrete probability distribution. For this case, we consider two service types, which are simultaneous and individual completion of the tasks. In the simultaneous subcase a job is assigned a service time at the end of which all its tasks terminate. In the second subcase, tasks of a job receive independent and identically distributed service times, which results in individual task service completions.

In the case of variable job size, the size of a job varies during the time that it is in the system. A job initially has a single task, however, it generates new tasks according to a Poisson process during its service time. The service times of the tasks are independent and identically distributed and each one requires a VM for its execution. Thus the number of tasks belonging to a job during its service time will be a random variable. A job is completed when all the tasks belonging to that job complete their service times. A job with variable size may be appropriate for modeling of service demands of mobile devices.

In the following sections of the paper we present performance analysis of the cloud computing models described in the above. Main contributions of this paper are as follows:

- We have considered systems with multiple classes of jobs with constant job sizes in number of tasks with homogeneous VMs. Assuming Poisson arrival of jobs with arbitrary service distributions, we have determined job blocking probabilities of each class and distribution of the utilization of resources under single server, multiple-server and multiple-server pool cases. In multiple-server case, we have determined fragmentation probability of a job's service among multiple servers. We have shown applicability of our results to study a power management algorithm that reduces the power consumption while maintaining a plausible job blocking probability under time-varying traffic load.
- We also derived job blocking probabilities and distribution of the utilization of resources with multiple classes of jobs with heterogeneous VMs.
- We determined probability distribution of the service time and average number of jobs for a system with constant job sizes and independent task completion times.
- We considered a system with jobs arriving to the system according to a Poisson process with variable job size in number of tasks. It is assumed that a job will generate new tasks randomly during its service time in the system. We have derived service time distribution of a job, distribution of the number of jobs and total number of tasks in the system.

The remainder of this paper is organized as follows. Related work is discussed in Section 2. In Section 3, we study systems with homogeneous VMs with constant job sizes and simultaneous task release times. Sections 4 and 5 extend the analysis of Section 3 to systems with heterogeneous VMs and jobs with independent task release times respectively. In Section 6, we present modeling of a system with variable