Parallel Computing 44 (2015) 1–17

Contents lists available at ScienceDirect

Parallel Computing

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

Cost-efficient coordinated scheduling for leasing cloud resources on hybrid workloads

Jian Li [a], Sen Su [b,*], Xiang Cheng [b], Meina Song [b], Liyu Ma [c], Jie Wang [d]

[a] School of Software and TNList, Tsinghua University, Beijing, China
[b] Beijing University of Posts and Telecommunications, Beijing, China
[c] School of Computer Science, Carnegie Mellon University, USA
[d] Department of Computer Science, University of Massachusetts, Lowell, USA

ARTICLE INFO

Article history:
Received 17 March 2014
Received in revised form 13 November 2014
Accepted 9 February 2015
Available online 14 February 2015

Keywords:
Cloud computing
Cost efficient
Interactive services
Batch jobs

ABSTRACT

Cloud service providers, leasing resources from cloud vendors under the pay-per-use service model, would want to minimize rental costs while meeting users' computing needs. They typically serve the following two types of workloads: interactive service requests and batch jobs. Early algorithms were devised to deal with either type of workloads, but not both. In the presence of a mixture of both types of workloads, we observe that these algorithms would often overproduce virtual-machine (VM) instances, resulting in much higher rental costs than necessary. In particular, we show that the VM instances generated by these algorithms for interactive services tend to incur significant resources unused. We present a coordinated scheduling algorithm to solve this problem. First, we use a priority function to handle interactive services, meet stringent service response time, and in the same time collect residual resources needed for batch jobs. Second, we use queueing analysis to adjust resource allocations based on predictions of resource requests for interactive services. Third, we schedule batch jobs according to the dynamics of residual capacity and spot instance pricing. Using traces from real-world interactive services and a library of batch jobs in practical applications, we demonstrate using numerical analysis that our coordinated scheduling is superior to the existing algorithms on cost efficiency designed for either type of workloads.

1. Introduction

Typical cloud vendors, such as Amazon EC2 [1], Windows Azure [2], and UltiCloud [3], allow commercial service providers to lease computing resources under a pay-per-use pricing model. For example, Amazon, arguably the most popular cloud vendor on the market today, allows service providers to lease different types of virtual machine (VM) instances [4–7]. Under this model, additional VM instances would be generated to meet new demands of workloads. These VM instances, however, may be under utilized. Service providers would want to minimize resource rental costs while still meeting workload demands.

* Corresponding author.
E-mail addresses: upton@tsinghua.edu.cn (J. Li), susen@bupt.edu.cn (S. Su), chengxiang@bupt.edu.cn (X. Cheng), mnsong@bupt.edu.cn (M. Song), west@gmail.com (L. Ma), wang@cs.uml.edu (J. Wang).

http://dx.doi.org/10.1016/j.parco.2015.02.003
0167-8191/

While a wide range of workloads may be served, they can be characterized into the following two types: interactive services and batch jobs [8,9]. Early algorithms were focused on cost-conscious scheduling for either type of workloads [10–15], but not on a mixture of both. Kurowski et al. 's seminal work devises novel data structures and online algorithms for scheduling both computational batch tasks and tasks with advance reservation requests[16]. We follow their methodology to further devise cost-efficient scheduling approach for service providers to process hybrid workloads. In this work, we devise coordinated scheduling approach based on the following observations:

1. Interactive services (e.g., web search, online video, and business transactions) would need to process user requests within strict time constraints. Long response time directly impacts business revenue (e.g., Amazon estimates that every 100 *ms* of latency would cost 1% more in sales [17]). In practice, we serve long-term interactive services using VM instances charged by hourly rates. We note that the request-arrival rate for an interactive service is highly dynamic, and so VM instances that were scheduled for this workload may contain substantial resources unused in the rental period. These unused resources can be used to serve batch jobs with flexible response time.
2. Batch jobs (e.g., recommendation calculations and financial analysis) would usually incur higher computational complexity. In general, a batch job can be divided into subtasks that may be scheduled to run at anytime. The time constraint imposed on processing the batch job is much more flexible than that of interactive services, thus providing opportunities to split batch jobs across a large number of time slots.

We observe that unused resources after serving interactive services are often scattered across a large number of VM instances and time slots, and the unused resources may be insufficient to execute batch jobs. We present a coordinated scheduling algorithm to address these issues. First, we devise an online algorithm called Online Cost-efficient Scheduling (OCS) that does the following: OCS uses a priority function to measure the urgency of requests, where the value of the priority function is based on the expected VM speed to complete a request, and assigns faster VM instances to requests with higher priorities. When the VM instances scheduled for interactive services contain unused resources, OCS uses a migration mechanism to collect residual capacities. We show that this strategy can improve cost-efficiency of interactive services while meeting the time constraints. Since job migration in a cloud incurs small overhead [18], OCS offers significant benefits.

Second, we devise an algorithm called Dynamic Resource Planning (DRP) to terminate unneeded VM instances before the next billing cycle starts based on prediction of daily patterns of resource usage from interactive services using queueing analysis. On the other hand, when the remaining available resource capacity within an hour is insufficient to handle batch workloads, we will use Amazon's spot instances [1] priced on actual resource consumptions. Since the spot instances' prices are generally much less than the normal VM instances, we can lower computing costs for time-flexible batch jobs.

Third, we devise an algorithm called Cost-conscious Scheduling algorithm (CCS) to dispatch batch jobs. The core of CCS is workload partitioning, which splits batch jobs across a large number of time slots according to the remaining resource capacity and spot instance pricing. The algorithm consists of three components: (1) abstraction refinement, (2) deadline distribution, and (3) planner. The abstraction-refinement component groups interdependent tasks according to the sizes of unused intervals of VMs scheduled for interactive services. The deadline-distribution component assigns proper deadlines to each task of a batch job on critical paths and computes sub-deadlines for all of its direct predecessors in the job. The planner component sorts the tasks by deadlines, and generates a schedule with minimum rental costs. One may worry about overly "segmented" schedules, for context switch incurs overhead and cache pressure. However, the context switch overhead, typically in the range of a couple of seconds, is much smaller compared to the deadline that is often a few hour long for batch jobs [19]. In addition, local storage devices (e.g., disk or tape drives) can be used to store the state of a process.

To evaluate our coordinated scheduling, we implemented a simulator based on CloudSim, which includes a job dispatcher, a resource planner, a cloud, and VM instances. We used interactive workload traces [20,21] and a variety of realistic batch workloads [22] to analyze the practicality of our design. Evaluation using real-world workloads reveals significant reduction of resource rental costs using OCS, DRP, and CBS individually and in combinations. In particular, our coordinated scheduling can reduce between 12% and 39% rental costs while meeting the performance-based requirements for interactive services and batch jobs.

The structure of this paper is outlined below: We describe the system model for service providers in Section 2 and we formulate the resource rental planning problem, including the cloud resource model, the application model, and the objective function in Section 3. We then describe our coordinated scheduling algorithms for interactive and batch workloads in Section 4. In Section 5 we present numerical simulations for performance evaluation. We review related work in Section 6. We conclude the paper in Section 7 and discuss future work.

2. Related work

In the realm of interactive services, early results provided important insights on how to apply adaptive control theory to dispatching interactive service requests for timely responses. Sha et al. [23] developed a queueing model predictor with a feedback loop to achieve response time regulation. Liu et al. [24] devised a queueing-model-based adaptive control