Computers and Electrical Engineering 000 (2016) 1–14

Contents lists available at ScienceDirect

Computers and Electrical Engineering

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

A cost-effective strategy for Cloud system maintenance

Xinyi Li*, Yong Qi, Pengfei Chen, Yang Fan

Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an, Shaanxi Province 710049, China

ARTICLE INFO

Article history:
Received 31 July 2015
Revised 5 September 2016
Accepted 6 September 2016
Available online xxx

Keywords:
Cloud computing
Availability
Software maintenance
Restless multi-armed bandit

ABSTRACT

To guarantee the high availability of modern Cloud-based applications, service backup is commonly used to maintain the failed services. However, an ineffective maintenance strategy based on virtual machine (VM) backup will cause a high budget and low resource utilization. This paper proposes a cost-effective multi-VM maintenance strategy to minimize maintenance cost with limited backup VMs in the Cloud. Different from the single-system maintenance strategies that pursue local optima by considering only one system each time, the proposed strategy considers all the running systems simultaneously from a global perspective. We formalize the multi-system maintenance as a restless multi-armed bandit problem, propose the concept of Healthy Indices of VMs with Lagrangian relaxation and develop an index based policy for maintenance scheduling. Given the limited backup VMs, the experiment results show that the proposed cost-effective multi-VM maintenance strategy significantly outperforms single-system maintenance policies in terms of total maintenance cost and system availability.

1. Introduction

The ever-growing Cloud-based systems are suffering performance degradation and even failures due to internal factors such as software bugs and external factors like misconfiguration [1]. According to [2], the cost of downtime per hour can go from 100,000 for online stores up to 6 million dollars for online brokerage services. To settle these problems, a preventive approach called software rejuvenation has been proposed [3]. Software rejuvenation recovers running applications proactively to prevent future failures. Recover activities involve occasionally terminating a software system, cleaning the internal state and/or runtime environment, and rebooting the system [3]. By combining the software rejuvenation with virtualization technique, the availability and performance of applications in the Cloud can be significantly improved. Several works dig into the optimal rejuvenation scheduling with backup virtual machines (VMs) [4]. However, an ineffective maintenance strategy based on backup VMs leads to a high maintenance cost and low resource utilization.

Given limited backup VMs, high resource utilization and low substantial cost can be achieved by sharing one backup VM among multiple service VMs. To effectively share the limited backup VMs, the important issue is to determine a cost-effective software rejuvenation strategy. It has been shown that simultaneous failures of multiple service VMs become commonplace in large-scale distributed systems [5]. When the number of simultaneously failed service VMs is larger than that of backup VMs, the single-system maintenance strategies [6, 7] will cause serious impacts on the system availability in the Cloud because they mainly focus on optimizing the maintenance cost of each individual system and choose the rejuvenated

* Corresponding author at: Department of Computer Science and Technology, Xi'an Jiaotong University, Room 1304#, Building 23, Jinhua South Road 168, Xi'an, Shaanxi 710048 China. Tel.: +1 9842699980, +86 15991622931.
E-mail addresses: xingga.li@stu.xjtu.edu.cn, xinyili@outlook.com (X. Li).

http://dx.doi.org/10.1016/j.compeleceng.2016.09.009
0045-7906/

service VMs locally without considering the cost of the whole distributed system. As for multi-system maintenance, only a few works can be found in literatures. Yang [8] investigated software maintenance policies in cluster systems, but only one node was allowed to be rejuvenated at a time epoch and the simultaneous multiple failures could not be tackled effectively, which will eventually result in low system availability. The work in [9] minimized the performance degradation of multi-tier applications by employing limited backup VMs in the Cloud, but the maintenance cost is not taken into account.

To minimize the maintenance cost in the Cloud, a cost-effective strategy should be capable of detecting and predicting the performance degradation or impending downtime of multiple service VMs, so that the planned rejuvenation can be initiated at the appropriate time and Cloud providers can avoid huge economic loss.

Designing such a desirable strategy is **challenging**, due to the following four reasons.

- There are thousands of service VMs in a large-scale Cloud system. Formalizing each service VM as a variable will eventually result in an intractable super high dimensional optimization problem.
- In the Cloud, the performance degradation processes of service VMs are quite different from each other. Various runtime behaviors need to be handled to ensure effective maintenance scheduling of service VMs from a global perspective.
- Since the rejuvenation overhead is non-neglectable, both the downtime cost and the rejuvenation cost should be taken into account when performing rejuvenation. The optimal maintenance cost is difficult to calculate when there are a lot of service VMs with variety of downtime costs and rejuvenation costs.
- Due to the dynamic variations of service VM performance, the maintenance strategy should work in real-time with a low overhead.

Aimed at solving the above difficulties, this paper presents a dynamic maintenance solution to the performance degradation of the virtualized Cloud platform with limited backup VMs, which we call the Cost-effective Multi-VM Maintenance Strategy (CMMS). We formulate the multi-VM maintenance as a Restless Multi-armed Bandit (RMAB) problem [10], derive explicit Healthy Indices of service VMs with Lagrangian relaxation, and develop an index based policy for maintenance scheduling. Compared with the traditional dynamic programming approaches which suffer from dimension explosion and iterative computation, the proposed index based policy has a low computational complexity. In simulation experiments, we comprehensively evaluate CMMS and validate its effectiveness. It is shown that CMMS has superiority in reducing maintenance cost and improving system availability when there are limited backup VMs.

The **contributions** of this paper include formulating the multi-VM maintenance as a RMAB problem for the first time, and proposing a novel Healthy Index based policy to schedule the multi-VM maintenance when there are limited backup VMs in the Cloud.

The remainder of this paper is structured as follows. The multi-VM maintenance is formalized as a RMAB problem in Section 2. Section 3 presents the index based solution to the formalized multi-VM maintenance problem with an index based policy. Experimental evaluation is shown in Section 4. Section 5 concludes this paper.

2. Problem formulation

To design an efficient software maintenance policy, it's necessary to know the running behaviors of service VMs. Therefore, we first leverage a fine-granularity multi-degradation-state model to depict the running behaviors of service VMs.

2.1. Multi-degradation-state model

In this paper, service performance means the quality of services (i.e., QoS such as response time, throughput or task execution time) hosted in the service VMs. QoS can directly impact the end users' experience that is critical in Cloud-based systems. In the life time of a running service VM, its performance deteriorates over time, i.e., response time increases, throughput decreases and task execution time increases gradually due to accumulated internal errors, exhausted available resources and varying workload. Software performance degradation can be found in various types of software systems, and several recent works deal with optimizing VM maintenance on virtualized clusters which is an emerging research area. During the performance degradation, the state of a service VM changes from state i to state $+1$ ($i=0, 1, \ldots, L-2$), eventually to the system failure state L if without any intervention. Moreover, it is possible that any degradation state i can directly reach the failure state L. The stochastic state deterioration process of a service VM can be described by a Markov chain model. As time goes on, the system service capability is decreasing monotonically, and the service performance of state j is lower than state i ($i < j$). When a failure occurs, the service VM crashes and transits to state L. Then a recovery operation is required. After the recovery, the service VM returns to the initial state (state 0), in which the service VM has the best capability and performance. Suppose that the state transition of a service VM is described by a right-skip free Markov chain and the transition probability from state i to state j is given by $P(i, j)$ ($i, j=0, 1, \ldots, L$), the multi-degradation-state model of a service VM is shown in Fig. 1.

2.2. Multi-VM maintenance problem

In a real-world enterprise level production environment, many Cloud users prefer deploying a single service on a dedicated VM, called ServiceVM, as the service performance can be seriously decreased due to the performance interference

Please cite this article as: X. Li et al., A cost-effective strategy for Cloud system maintenance, Computers and Electrical Engineering (2016), http://dx.doi.org/10.1016/j.compeleceng.2016.09.009