Computers and Electrical Engineering 67 (2018) 235–251

Contents lists available at ScienceDirect

Computers and Electrical Engineering

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

A hierarchical control framework of load balancing and resource allocation of cloud computing services☆

Nikolaos Leontiou[a], Dimitrios Dechouniotis[*,b], Spyros Denazis[a], Symeon Papavassiliou[b]

[a] Department of Electrical and Computer Engineering, University of Patras, Greece
[b] School of Electrical and Computer Engineering, National Technical University of Athens, Greece

ARTICLE INFO

Keywords:
Cloud computing
Load balancing
Application placement
Resource allocation
Admission control
Fuzzy modeling
Feedback control

ABSTRACT

Service providers must guarantee Quality of Service (QoS) requirements of the co-hosted applications in a data center and simultaneously achieve optimal utilization of their infrastructure under varying workload. This paper presents a hierarchical control framework that aims at compromising antagonistic objectives inside a data center. The local control level tackles simultaneously the problems of resource allocation and admission control of virtual machines while the upper level addresses together the load balancing of the incoming requests and placement of virtual machines into a cluster of physical servers. Numerical results show that the cooperation of the two control layers guarantees the satisfaction of the system's constraints and the user's requirements towards the fluctuations of incoming requests.

1. Introduction

Power consumption and energy efficiency are dominant in the design and management of cloud computing data centers. Due to the large number of servers and the increasing complexity of the network infrastructure, energy costs are quickly rising in large-scale service centers. A sustainable and power-aware management system should provide a trade-off between service performance and energy consumption. Virtualization provides a promising approach of consolidating multiple online services in fewer computing resources within a data center. This technology allows a single server to be shared among many performance-isolated platforms called virtual machines (VMs). Virtualization also allows the on-demand or utility computing a just–in–time resource provisioning model in which computing resources such as CPU, memory, and disk space are made available to applications only as needed and not allocated statically based on the peak workload demand. By dynamically provisioning VMs, consolidating the workload, and turning on only the necessary servers, data center operators can maintain the desired Quality of Service (QoS) while achieving higher server utilization and energy efficacy. However, as computing systems become larger and more complex, the task of tuning numerous control parameters, such as CPU and memory share and disk space, is becoming tedious, time-consuming, expensive and almost impossible for data center operators to do manually. Thus it is highly desirable for such systems to be largely self-managing or autonomic, only requiring high-level guidance by operators.

The present study aims at developing a scalable and optimal resource allocation framework for consolidating web applications in a single cloud data center. The goal is to provide services which achieve QoS requirements, while utilizing only the essential resources

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Associate Editor Dr.L. Bittencourt.

* Corresponding author.
E-mail addresses: nleontiou@ece.upatras.gr (N. Leontiou), ddechou@netmode.ntua.gr (D. Dechouniotis), sdena@upatras.gr (S. Denazis), papavass@mail.ntua.gr (S. Papavassiliou).

https://doi.org/10.1016/j.compeleceng.2018.03.035
Received 15 June 2017; Received in revised form 21 March 2018; Accepted 22 March 2018

of the cloud infrastructure. In this context, QoS requirements, formally stipulated by Service Level Agreement (SLA) contracts, are difficult to satisfy due to the high variability of request workload, which may vary by orders of magnitude. Thus the effective resource allocation and the implicit reduction of energy consumption are still open and challengingresearch problems. Cloud computing is the recent adopted paradigm for service delivery that hosts applications in virtualized environments. Physical resources are partitioned into multiple virtual ones, creating isolated VMs that run at a fraction of the physical system capacity. Autonomic self-managing techniques are implemented by network controllers which can establish the set of applications executed by each server (i.e., application placement problem), the request volumes at various servers (i.e., load balancing problem), and the capacity devoted to the execution of each application at each server (i.e., capacity allocation problem). In this paper, we propose a two-layer control architecture that addresses cooperatively the problems of load balancing, application placement, capacity allocation and admission control with respect to satisfaction of QoS metrics and optimal recource allocation (or in equivalent minimization of energy consumption). In detail, the local (lower) level includes distributed individual controllers based on fuzzy Takagi–Sugeno modeling for each application [1]. A set of feasible operation points is determined based on the derived models by solving a dynamic programming proble, e.g., for an agreed response time of 1 s and a VM with CPU share of 20% ican serve a workload of 40 requests/s, together with stabilizing control strategies for each point. Inputs of local controllers are the allocated capacity (CPU share) and the admitted load while the regulated output is the response time of each application's VMs (vertical scaling). The global (upper) level supervisory controller, considering the available set of VM's operating points implied by the local controllers, determines the number and the size of necessary VMs for meeting the total incoming request rate (horizontal scaling). The distribution of the workload among the activated VMs and the placement of them in the cluster of servers are made in such a way that the minimum number of active servers. This indirectly leads to the reduction of the energy consumption of the data center. Regarding to studies in [2] and [3], the further improvements of the proposed solution are summarized as follows:

- A fuzzy Takagi–Sugeno state space modeling captures more accurately the dynamic complex behavior of each VM than the Linear Parameter Varying (LPV) modeling in [2].
- The global level controller consider the load balancing problem as an unbounded knapsack problem, which needs less time to be solved than the greedy approach in [3].

The rest of the paper is structured as follows; Section 2 discusses related work. Section 3 contains an analytical description of problem. Section 4 presents fuzzy modeling and the determination of feasible operating points. Sections 5 and 6 describe global and local level controllers respectively. In Section 7, the implementation and evaluation of the approach are presented. Conclusions are drawn in Section 8.

2. Related work

Various management techniques or combinations of them have been proposed to automate the performance and power management of computing systems such as control theory, optimization theory, intelligent control and machine learning. This problem is mainly approached by solving the subproblems of resource allocation, admission control, load balancing and application placement separately or a combination of them. In this section the general guidelines of them are presented and some of the most representative studies are analyzed. A main distinction can be made by the use of models or not.

In model free approaches are divided in the following categories. Threshold based rules, i.e., [4], are easy to implement, but the determination of the rules is a difficult problem and as it depends on the workload trends, neither stability nor performance is guaranteed. Also they normally suffer from overprovision problems. Model free optimization techniques, such as [5], suffer from stability issues. The optimal solution may be feasible but its stability is questioned, which means that a small disturbance can lead the system to instability or performance oscillations. Reinforcement learning techniques, like [6], suffer from long periods of training. Rule based approaches, such as fuzzy control, i.e., [7] have scalability issues as more applications and servers become available. All of the above model free approaches lack stability analysis which may lead to performance degradation.

Model based approaches require more domain knowledge. They are more complicated and provide more insight into the original system and capture the dynamics of the system. Model based approaches divide into two phases, analysis and planning. In the analysis phase a model of the under study system is exported. Currently, most model based approaches are based on control theory [8] and queuing theory [9]. In the control theory literature, the dominant type of modeling is the state-space models that are accurate in an operation region of a nominal operation point. They efficiently capture dynamic behavior of the system and stability analysis is provided. On the contrary, the queuing theory models, such as M/M/1 and M/G/1 queue models, assume specific distribution for the incoming request and service rate and determine the operating point only for steady-state conditions. They cannot model transitive phenomena of the system's operation and the stability of the steady-state operation point provided by queue analysis is at stake when the system is operating in a different region. In the planning phase an algorithm for performance management is scheduled for its solution. Dynamic state-space models are combined with feedback control techniques to lead the trajectory of the system on the desired operation point and provide stability guarantee. Queuing models combined with an optimization method are usually designed for an steady-state operating point.

In [2], there is an extensive description and categorization of the studies on resource allocation and admission control of services deployed in cloud computing. In the following paragraphs, these relative studies are enriched with the most interesting works that are close to the approach proposed here. In [9], the authors proposed a holistic approach for application placement, admission control, resource allocation. They used queuing theory models and a greedy algorithm to solve the application placement problem and