Applied Soft Computing 26 (2015) 285–298

Contents lists available at ScienceDirect

Applied Soft Computing

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

Using IDS fitted Q to develop a real-time adaptive controller for dynamic resource provisioning in Cloud's virtualized environment

Fouad Bahrpeyma*, Ali Zakerolhoseini, Hassan Haghighi

Faculty of Electrical and Computer Engineering, Shahid Beheshti University G.C., Tehran, Iran

ARTICLE INFO

Article history:
Received 12 November 2013
Received in revised form 3 October 2014
Accepted 5 October 2014
Available online 17 October 2014

Keywords:
Q-learning
Active learning method (ALM)
IDS fitted Q (IDSFQ)
Real-time reinforcement learning in continuous space

ABSTRACT

Reinforcement learning (RL) is a powerful solution to adaptive control when no explicit model exists for the system being controlled. To handle uncertainty along with the lack of explicit model for the Cloud's resource management systems, this paper utilizes continuous RL in order to provide an intelligent control scheme for dynamic resource provisioning in the spot market of the Cloud's computational resources. On the other hand, the spot market of computational resources inside Cloud is a real-time environment in which, from the RL point of view, the control task of dynamic resource provisioning requires defining continuous domains for (state, action) pairs. Commonly, function approximation is used in RL controllers to overcome continuous requirements of (state, action) pair remembrance and to provide estimates for unseen statuses. However, due to the computational complexities of approximation techniques like neural networks, RL is almost impractical for real-time applications. Thus, in this paper, Ink Drop Spread (IDS) modeling method, which is a solution to system modeling without dealing with heavy computational complexities, is used as the basis to develop an adaptive controller for dynamic resource provisioning in Cloud's virtualized environment. The performance of the proposed control mechanism is evaluated through measurement of job rejection rate and capacity waste. The results show that at the end of the training episodes, in 90 days, the controller learns to reduce job rejection rate down to 0% while capacity waste is optimized down to 11.9%.

1. Introduction

In current distributed computing world, Cloud's virtualized environment is known as the enabling technology of "on-demand", "pay as you go" and "resource pool" paradigms in the context of computational resources provisioning. Recently, adaptive resource provisioning mechanisms, which are assisted by intelligent systems, have gained wide popularity in the context of Cloud computing. However, in current Cloud's spot market, which is one of the most challenging environments to deal with, facing with the dynamic scale of requests for computational resources requires designing dynamic resource management scenarios [1–4]. Thus, Dynamic Resource Provisioning (DRP) is essential to assure efficient resource utilization in the Cloud's infrastructure systems. Elastic resource scaling, the ability of the Cloud to scale its resources to the volume of incoming demands, is an embeded concept for dealing with dynamicity in Cloud's virtualized environment. Elasticity allows for dynamic resource provisioning when facing with dynamic demands [5].

Dynamic resource provisioning in the Cloud's spot market can be formulated as an adaptive control problem. Adaptive control approaches have been applied successfully to provide dynamic resource allocation solutions in distributed physical servers [6–9]. Because of high amount of uncertainty, along with the lack of explicit model for Cloud's resource management systems, classical adaptive control mechanisms are not applicable to establish high performance dynamic resource provisioning. Nonetheless, artificial intelligence (AI) methods provide facilities to introduce model free solutions through intelligent approaches [10–12]. Model-free techniques like RL provide the adaptive controller with autonomous data driven solutions which are organized gradually to such uncertain environments [13].

Recently, RL methods have gained wide popularity in adaptive control problems with successful applications [14–19]. In engineering applications, RL is known as a bio-inspired machine learning technique used to solve sequential decision problems [20–22]. In RL, the major part of the efforts is concentrated on learning by interacting with the environment. An agent learns through evaluative feedback, as the environment pays for the action that the agent performs at each state [23]. Through an RL process, the

* Corresponding author. Tel.: +98 9362136997.
E-mail addresses: F.Bahrpeyma@Mail.sbu.ac.ir, Fouad_Bahrpeyma@yahoo.com (F. Bahrpeyma).

http://dx.doi.org/10.1016/j.asoc.2014.10.005
1568-4946/

agent learns the optimal policy Q for receiving better rewards (as the feedbacks of the environment). Among the RL methods, Q-Learning [24] has proved its capabilities when employed in real-time applications [25–27]; Q-learning is commonly used for solving finite Markov Decision Process (MDP) problems, one of the most promising solutions in engineering applications [20–22,24].

RL problems are commonly solved by dynamic programing (DP) techniques. However, in continuous domains, due to the curse of dimensionality [28] together with the continuity of the domain, the state space of the RL problem cannot be modeled by conventional DP or finite state space formulation techniques. Therefore, a solution is required to generalize the experiences of the RL algorithm to unseen conditions. In such situations, general approximation techniques, like artificial neural networks (ANNs), are used to approximate the reward function of the RL problem [29].

One of the most promising approximation-based approaches is the neural fitted Q (NFQ) technique [30]. Fitted Q provides ability to remember the experiences of interactions with the environment as beneficial knowledge revealed though exploration, provided that the agent can reflect on past experiences in the similar conditions to improve long term (cumulative) reward [23,31,32]. Therefore, convergence to the optimal policy of the RL algorithm can be guaranteed.

However, when the decision time is important and the number of experiences increases, approximation techniques, which are employed as fitted Q techniques, commonly fail to provide acceptable response time. In essence, popular methods used for approximation in fitted Q approaches like ANNs, Takagi–Sugeno–Kang Fuzzy Inference System (TSK-FIS), Radial Basis Function Networks (RBFNs) and Neuro-Fyzzy (NFs) suffer from considerable computational complexities which impose a meaningful delay to the RL process. On the other hand, incremental methods like recurrent neural networks (RNNs) face with the possibility of divergence.

In this paper, a novel use of IDS modeling method [33], which is a solution to modeling multi-dimensional systems without dealing with computational complexities, is proposed to model the Q-values in real-time constrains. IDS has shown its real-time capabilities for modeling multiple-inputs, single-output (MISO) approximation problems [34]. Thus, IDS properly suits the Q-learning problem that is an MISO problem (there is only Q-value as the output of approximation procedure). The IDS model is trained via a recursive partitioning algorithm, called the active learning method (ALM) [35].

This paper proposes a novel fitted Q method to accelerate the modeling procedure inside Q-learning for continuous RL problems. Since Q-learning is one of the most practical solutions to adaptive control in real-time applications, this paper studies employment of Q-learning to provide an adaptive control solution to DRP problem in Cloud's virtualized environment. The proposed method is supposed to learn how to avoid job rejection while gradually leads the management system to optimal resource usage (which is tightly geared with energy consumption in Cloud's environment).

The rest of the paper is organized as follows: the next section describes the Q-learning method as the engine of the adaptive controller proposed in this paper. Section 3 is an overview on the DRP problem. In Section 4, the proposed method is explored to design an adaptive controller for the DRP problem in Cloud's virtualized environments. Experimental results are presented in Section 5. And finally, Section 6 concludes the paper.

2. An overview on *Q*-learning

Due to the fact that an explicit model of the environment is required for obtaining optimal policies in DP solutions, classical DP methods commonly face with difficulties for providing explicit model in addition to corresponding computational costs [20]. Nonetheless, in continuous domains, RL methods are addressed as approximate DP (ADP), regarding less computational costs and no assumption on having explicit models of the environment [36]. ADP is a perspective of RL which introduces approximation solutions to deal with the curse of dimensionality, continuous domain requirements and uncertainty in the environment of the problem.

Among model-free ADP methods, Q-learning is guaranteed to converge [37]. Q-learning is adaptive, can be real-time and works based on a single value reflected on pairs (state, action), denoted as cumulative reward, so as to derive the optimal policy. Q-learning starts from random initial Q-values and updates values online via the observed transitions $(s_t, a_t, s_{t+1}, r_{t+1})$ [24,38] (where s_t is the current state, a_t is the chosen action, s_{t+1} is the state after performing a_t at s_t, and r_{t+1} is the reward reflected on performing a_t at s_t); after each transaction, the Q-value of the corresponding (state, action) pair is updated with:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_{t+1} + \gamma \max_a Q(s_t, a) - Q(s_t, a_t)), \tag{1}$$

where α is the learning rate, γ is the discount factor that determines the impact of future reward, r_{t+1} is the immediate reward reflected on (s_t, a_t, s_{t+1}) tuple, and $\max_a Q(s_t, a)$ is the maximum cumulative reward which can be received through choosing an action in state s_t.

For each experiment, when the agent performs an action a_t in state s_t, the Q-entry belonging to (s_t,a_t) is updated depending on the utility of the chosen action in the corresponding state. As mentioned earlier, Q-values represent the optimal policy for achieving the goal of the learning process [22,24].

3. DRP in Cloud

This paper offers that the problem of DRP in Cloud's virtualized environments can be solved via adaptive control on dynamic scaling of resources to maximize the revenue function. Optimal resource provisioning in the Cloud's environment is a crucial task for Cloud infrastructure providers; they should provide hosts according to the Service Level Agreements (SLAs) at the proper level of Quality of Service (QoS) for client satisfaction. Cloud providers should also optimize the DRP task so as to reduce extra costs usually imposed by wastage in energy consumption in active servers. Therefore, regarding both client satisfaction and resource optimization is required to gain better revenue for Cloud providers.

Regarding client satisfaction, such solutions should avoid job rejection, which is expected to persuade the clients to submit their requests to other Cloud providers. In this paper, two major reasons are considered for job rejection, including long response time and inadequate capacity. The controller should avoid job rejection by providing enough capacity for the given jobs as well as reducing the response time. Due to the fact that the response time is more crucial in spot markets, jobs are considered to be presented to the Cloud as spot requests which are required to be responded in real-time manner; otherwise, the job is rejected. To guarantee client satisfaction, the proposed adaptive controller should consider higher priority for job rejection accusation in comparison to resource usage optimization.

4. The proposed RL based adaptive controller

The DRP problem is a proper suit for RL as an optimal control mechanism [39]. Optimal controllers are commonly addressed as adaptive controllers. In the literature, the most practical architecture for adaptive control is the Model Reference Adaptive Control (MRAC) mechanism [40]. Trying to formulate the DRP problem as an adaptive control problem, from the MRAC controllers' point of