Optimization Task Scheduling Algorithm in Cloud Computing

Somayeh Taherian Dehkordi, Vahid Khatibi Bardsiri

Department of Computer Engineering, kerman Branch, Islamic Azad University, Kerman, Iran
S.taherian2000@gmail.com

Department of Computer Engineering, Bardsir Branch, Islamic Azad University, Kerman, Iran
Kvahid2@live.utm.my

Abstract

Cloud computing refers to services that run in a distributed network and are accessible through common internet protocols. Presenting a proper scheduling method can lead to efficiency of resources by decreasing response time and costs. This research studies the existing approaches of task scheduling and resource allocation in cloud infrastructures and assessment of their advantages and disadvantages. Afterwards, a compound algorithm is presented in order to allocate tasks to resources properly and decrease runtime. In this paper we proposed a new method for task scheduling by learning automata (LA). This method where has named RAOLA is trained by historical information of task execution on the cloud, then divide task to many classes and evaluate them. Next, manage virtual machine for capture physical resources at any period based on rate of task classes, such that improve efficiency of cloud network.

Keywords: Resource allocation; cloud environment; learning automata.

I. Introduction

In cloud computing, IaaS approach is to increase the efficiency and utilization of resource, equipment and existing networks (Durao et al., 2014). In this approach, it is attempted to manage request executions so that operational costs such as energy consumption in datacenters, and costs of networks would be reduced (Mingxin, 2015). Cloud computing by numerous virtual machines on some data centers increase capability of response to requests. Here, managing virtual machines and physical resources, besides scheduling policies of tasks, is a significant issue. An inappropriate scheduling may involve numerous resource for a series of requests while, an optimized scheduling with less resources and better management give the same response. In cloud computing there are many serial request of users at the same times and conditions is causing similar condition for cloud (Buchert et al., 2015). Iteration of similar events indicates that a learning algorithm is able to provide suitable efficiency in such conditions. Applying intelligent methods based on learning to the cloud, is increasing in the field of optimization of tasks scheduling and resource allocation (Mathew et al., 2014). In this research, we will attempt to present a method based on scheduling information of the past of cloud so that it would be able to contribute performance optimization of the cloud scheduler. A method based on learning automata (LA) will be presented for realize optimum indicate of a behavioral cloud in different time period. We show, repeat of similar task in indicated time is suitable for equipment cloud such that throughput has been increased. So, we proposed a new method by learning automata (LA), where has named RAOLA, such that use of historical information of task execution for improve behavioral of cloud network. Firstly, RAOLA divide
task and evaluate execution of any class, and realize rate of any class for any period. Then manage virtual machine for capture physical resources at any period based on rate of task classes, such that improve efficiency of cloud.

For task scheduling and resource allocation in cloud environments is necessary to consider issues mentioned below, while investigating the methods of resource allocation in the cloud:

A) Resource allocation must be in accordance with tasks scheduling.
B) Proposed method must be dynamically result in system utilization
C) Resource allocation process must have little computation head.
D) Behavior of resource allocation method must be in accordance with cloud distribution. Then a non-concentrated scheduling method is required.

According to issues mentioned above, using capabilities of learning automata can help us achieve a method based on learning behavior of system. It is necessary to specifically consider pattern of user request in diverse time. Achieving high efficiency, low makespan and more task execution by exciting resources is goal here.

At continue, first we have review of previous learning method for task scheduling and resource allocation. Secondly, we define new formalization for scheduling of resource in cloud environment. Thirdly, we define learning automata (LA) as an appropriate model from cloud environment. Frothy, we propose RAOLA as a new algorithm for schedule tasks and resource based on LA. Fifthly, we extend the proposed algorithm for the online scheduling. Finally, we evaluate and present empirical results of proposed method.

II. Related Works

By using existing capabilities in genetic algorithms and soft computations, learning methods follow a heuristic approach to reach an optimized scheduling. In (Xu et al., 2014) the genetic algorithm (GA) as a heuristic method for search an optimized solution in a large space of solutions, has proposed. At first step, random initializing of a chromosome population is performed for a certain duty. Each chromosome has a fitness value (makespan). Results of scheduling Task for machines are saved in a chromosome. After first population is produced, all chromosomes in the population evaluate themselves with the case based on the fitness value. In this evaluation having little makespan is a better mapping.

In (Ramezani et al., 2013), another scheduling algorithm is introduced based on combining methods for scheduling tasks in the cloud. In order to reach utilization and efficiency of maximum results, this algorithm simultaneously concentrates on tasks and resources to reach a general optimization. In this method, independent and dividable tasks which require different computation and different memories are scheduled by a genetic algorithm efficiently. In this method, it is assumed that the cloud system is heterogeneous. In other words, all resources of processes (Including CPUs) and communications are performed heterogeneously. Therefore, by considering memorial limitations and the most requests for cloud computations efficiency, the proposed method is attempting to provide a scheduling method based on GA.

The conceptual model is explained below:

A) Tasks are dynamic: so arrival time of the tasks is not obvious in advance. Each task has arrival time (ai), worst state of computation time (ei) and its duration (di). Ready time of the task is its arrival time.
B) Tasks don't have preemptive and each of them is independent.
C) Each task has shared access ways to a process unit. In this case it can share its resources with other tasks (Other tasks here must have their resources shared).

In (Elayaraja and Dhanasekar, 2014), another method to schedule resources of virtual machines (VMs) is introduced based on genetic algorithm. In order to schedule the system's variety and data of the past, behavior of the data which cause imbalance in the system load must be considered. Therefore, the object of provided scheduling is load balancing of the selected system. Here, genetic algorithm approach can perform scheduling of VMs located in the system by using data of the past and the current situation so that it solves the problem of load imbalance and travel cost. Generally, in the cloud environments, in order to schedule virtual machines (VMs), only data of current situation are used and previous situations of the system which have caused load imbalance in the system are not considered.

When task load balance occurs, the number of travels of virtual machines increases and it causes head costs for the cloud. Current resources of VM are allocated to each physical node and this choice has the least load on the system, this subject can create a context to introduce a genetic algorithm which provides a suitable scheduling for the resources of the system by using data of the past and current situation of the system.

III. Cloud Environment

In this section we present the architecture of the cloud network based on its components and elements for task scheduling and resource allocation. Assume that a cloud computing system is consisted of several heterogeneous process units including "m" units (m>1). The template of the tasks in this system is explained by figure 1.

![Figure 1. Task scheduling in cloud environment.](image)

If "T" is the task queue T= {T1, T2 ...Tm} including "m" numbers of users' requests, then "T" always changes dynamically. Therefore, in the time of "i", the characteristics of the user's request are Ti= {T1i, T2i ...Tim}. If physical resources of R (processor and space) are obvious in advance R= {R1, R2 ...Rn}, virtualization to the required number of Vi, are allocated on physical resources. An appropriate solution of "S" is consisting of allocation Vis to task queue. Other words, by Si as an optimized solution in the time "i", the appropriate response of allocation of VMs is provided. Therefore, the set of S = {S1, S2 ...Sr} indicates the series of scheduling resources to VMs and allocation of VMs to tasks.

Figure 1, shows the activity cycle of the cloud. The series of V= {V1, V2... VP} shows the number of virtual machines (VM). Here, presenting an optimized scheduling can be achieved by finding a possible solution Si.
Tasks are dynamic, so ready time of tasks (arrival) to "T" is not obvious in before. Each Ti task has an arrival time (ai), the worst execution (ei), and duration (di). Also, the basis of required space (si) involves the number of processors (ci) besides a part of bandwidth of virtual machine of Vj which will be applied by Ti (BW (i, j)) to execute itself. Some issues related to Ti are considered here. Therefore, data of T are in the form of an m×6 matrix. If T(i,j) refers to the resources of Ti when Vj is allocated to it, then the factor of utilization extent "Vj" is equal to the tasks which are responded by the jth virtual machine in the unit of time:

\[ U(V_j) = \sum_{i=1}^{m} T(i, j) / b_j \]  

(1)

Now the total utilization is:

\[ U(V) = \sum_{j=1}^{n} \sum_{i=1}^{m} T(i, j) / b_j \]  

(2)

According to the above definition, utilization of V depends on the number of tasks which have been able to possess an appropriate virtual machine which supplies the required resources of Ti before expiration of di from entrance time of ai. The scheduler model is shown in figure 2.

In this paper we assume that physical resources in the cloud environment are constant during scheduling and only task queue are working dynamically. These resources include cases below:

- **Space**: Disc = {Disc1, ..., Discz}
- **Host**: Host = { Host1, ..., Hostu }
- **Processor**: CPU = {CPU1, ..., CPUv}
- **Datacenters**: DC = {DC1, ..., DCw}
- **Cost of network**: Cost = {Cost1, ..., Costn}
- **Bandwidth**: Bandwidth = {BW1, ..., BWs}

Cloud environment scheduler dynamically faces a queue of tasks in each moment. When resources are provided by a virtual machine for at least one of the existing tasks in the queue, allocation of VMs to the tasks are performed based on the duration di. If Ve is a part of involved
virtual machines, VF = V - Ve would be the free part of resources in the cloud. Therefore, a list of all tasks on T queue which their required resources exist in VF, construct T' queue. Now scheduling approach must be able to allocate VMs in a form to maximize resource utilization. One response to scheduling and giving S response is that it sorts tasks based on least execution time or shortest task first (STF). Now, T" is the sorted queue of existing tasks and the scheduler allocates tasks to VMs existing in VF as the time it is possible.

\[ S = T \cap VF \]  

A learning algorithm can be presented for S optimization when users' requests in different task load times, consist similar tasks.

IV. Learning Automata

Learning automata (LA) is one of the types of graph-based methods for system modeling, which include a finite series of actions. An LA can be applied to define a finite discrete environment. A special feature provided by LA is when our specific system has a probabilistic environment of events. Here, a series of possible actions are evaluated by this probabilistic environment and its result are given to the automata through a response template (positive of negative) and automata selects this response in its next action selection. The final object is that Learning Automata learns to choose the best action among its actions (Meybodi et al., 2004). The general scheme learning automata is shown in figure 4.

![Learning automata and cloud environment.](image)

Learning automata can be classified as two sets of constant structure learning automata and variable structure learning automata (VSLA) (Thathachar and Sastry, 2004). One of the most important issues in the field of learning automata is to determine what is the best action? And how it is identifiable by LA? Generally, an action that can maximize the optimization probability of a function such as index of environmental utilization or can minimize a function such as system cost, can provide the concept of optimized behavior of LA or optimized appropriate action selection. Learning automata is a method to optimize system based on previous condition of the problem and interaction of its members. In the following, a general definition of LA is provided.

**Definition 1: Learning Automata**

A learning automata is quintuple \( LA = < F, G, \emptyset, \{a_1 \ldots a_n\} \) which \( \{a_1 \ldots a_n\} \) is the action series of the automata having "n" numbers. Also \( \{b_1 \ldots b_m\} \) is the input series of automata having "m"
numbers, and $\emptyset \,(t) = \{\emptyset 1, \emptyset 2... \emptyset k\}$ refers to data related to the system situation in time "t", and the function of producing new situation is $F: \emptyset \times \emptyset \rightarrow \emptyset$. The $G: \emptyset \rightarrow \emptyset$ is an output function which maps current situation to the next situation (Hyukho et al., 2010).

A variable structure learning automata consists of quadruple \{ $\alpha$ , $\beta$ , $\rho$ , $\tau$ \} which $\alpha$ are the series of actions of automata, $\beta$ are the inputs of the automata, $\rho$={$p_1$... $p_r$} is selection probability vector of each actions and $p(n+1) = T [ \alpha (n), \beta(n), P(n)]$ is the learning algorithm. There is an example of learning algorithm of "T" which is counted as a linear learning algorithm below. The appropriate response from the environment is linearly determined through the response of the previous step.

$$p_i(n+1) = p_i(n) + a[1 - p_i(n)]$$
$$p_j(n+1) = (1 - a) p_j(n) \quad \forall j \neq i \quad (4)$$

V. Resource Allocation and Task Scheduling By Learning Automata

In this section, a new method for scheduling task and resources in cloud environment based on learning automata (LA) has been proposed. The proposed method manages scheduling with pre-determined resources and dynamic tasks queue. This method has named RAOLA (Resource Allocation Optimization by Learning Automata). Execution results of RAOLA on a data set of a dynamic queue of $T$, set of task, by LA can optimize tasks resource allocation and scheduling through identifying behavior pattern of cloud performance.

We present this proposed method for extended online scheduling by ORAOLA algorithm. Also here, according to periodic evaluations of the extent of resource utilization, cloud scheduler results in performance optimization of the broker and the difference is that in this case it faces with computational complexities. Generally, in the cloud environment, scheduling virtual machines of $V_i$ is performed regardless of previous situations of the system and is merely performed based on current data of the system. Analysis of scheduling behavior shows the effect of previous situation of the system on the load imbalance of the system (Ramezani et al., 2013). We have assumed that existing resources are constant in the physical machines. Therefore, no travel is performed by virtual machines to balance task load and there is no head costs related to $V_i$s on physical machines. Here, scheduling must have the least load on the system by appropriate selection of $V_i$s for the duties. This prepares a context to provide an optimized algorithm using data of the past. Learning automata can model systems with discrete components that their behaviors have known effects on each other. The most important thing is to record experiences of the past and considering them in the future behavior of a system. Therefore, a correlation must be made between two capabilities (A, B) of recording experiences of the past, and optimizing the current behavior, which are provided by LA. When $T$ consists of numerous similar duties, or the
current situation of T has been iterated for many times in the past, a scheduling template based on scheduling behavior can be resulted by considering tasks loads. If LA would be a learning automata so that \( \beta(n) \) is the tasks added to \( T_{\text{new}} \) queue in the time "n", and \( V_f \) would be the free resources of virtual machines, then the function is defined as \( F: \emptyset \{T_{\text{new}}, V_F\} \rightarrow \emptyset \).

Scheduling function is defined as \( G: \{T", V_F\} \). In order to use data of the past to optimize scheduling, score function of \( p \) is defined in the following section. Assume that \( |T_i| \) is the weight of the required resources for the duty of \( T_i \). Then if the threshold \( TH = \max \{|T_i| \text{ } Ti \text{ is a task} \} \) is constant, we would define a parameter named \( H \) with the length of \( TH \). While \( G \) function schedules the tasks based on \( T \) we have:

\[
H[|S (i)|] = \max \{H(|S (i)|), \frac{U (V_f)}{U (V) \times |T_i|} \} \tag{5}
\]

Here, we have \( S (i) = T_i \), and all elements of the \( H \) parameter are 0 at first. Now, an effective method is proposed to optimize \( S \), based on structure of scoring the tasks when the iteration of tasks are high. Now, we present a learning automata scheduling algorithm. This algorithm which we have called RAOLA, by recording advantages resulted from scheduling tasks with different specifications, shows better performance when task load conditions of the cloud encounters similar situations.

A. RAOLA Scheduling Algorithm

If we consider components existing in the cloud such as VMs or cloudlets as our system components, the situation of these components is completely random in different times. If the performance of scheduling unit of existing tasks in \( T \) queue, and the unit of resources allocation management would always have the same priorities based on tasks and resources, their optimization function in similar conditions is verified by a same behavior. Therefore, learning automata, by comparing average application execution (AAE) of tasks as the factor of cloud utilization with the spent finance, can decide whether to allocate tasks to virtual machines and allocating center data to virtual machine. This decision can always be optimized based on past experiences. The function of \( \emptyset \) in LA is the situation data of the system which consists data of matrixes \( X \) and \( Y \). Briefly we have; \( \emptyset = T \times V \) and \( T_{\text{new}} = \{t'1 \ldots t's\} \). If \( T" \) would be a part of \( T" \) which has exited the queue for execution and \( \text{a VM}_{j} \) is allocated to it, \( F \) function creates a new situation by removing scheduled tasks and adding new tasks, while removing resources allocated of free resources and adding new free resources.

\[
T = ((T \ T")) \ T_{\text{new}} \tag{7}
\]

\[
V_F = ((V_F \ V (T"))) \ V_{\text{new}} \tag{8}
\]

Operator is row summing of two matrixes with the same number of columns, and operator indicates their subtraction. \( X \ Y \) means that it is equal to a \( Z \) matrix having columns equal to \( X \) matrix, and its rows consist of that part of \( X \) matrix that does not consist components of rows of \( Y \) matrix. For instance, for these two matrixes below we have:
Management of raw data scheduling behavior of cloud environment through learning automata presents an optimum way for resource allocation in the cloud based on learning system behavior in different task points. In cloud processing, similar events in the same time and conditions are iterating due to specific request of the users. Therefore, identification of behavior pattern of a cloud by the proposed LA, provides high efficiency in increasing system utilization.

B. RAOLA Scheduling Algorithm ORAOLA Algorithm for Online Scheduling
As we introduced the non-linear method for simultaneous online scheduling of tasks and resources, in this part we attempt to present an extension of a proposed online scheduling algorithm its online status and present ORAOLA algorithm. The ORAOLA scheduling algorithm has considered as a developed version of RAOLA for online scheduling. If resource allocation is managed by broker in the cloud, proposed scheduler behavior in RAOLA algorithm can be optimized for online scheduling by help of data transmitted to the broker dynamically from tasks existing in queue in the cloud in each moment.

VI. Algorithm Implementation
A simulator named Cloudsim is used to execute this algorithm in Java language environment. We executed resource allocation management of the algorithm in its simplest case which is responding the memorial request. At first, we conduct scheduling merely based on the smallest duty (STF). A threshold is defined for the requests (TH) and we evaluate the extent of utilizations based on this threshold. Step by step we update scheduling behavior and the effect of tasks with different extents in the utilization extent of the H parameter of the cloud and we set the selection priority based on utilization extent of tasks in cloud behavior. Learning automata gets trained step by step by comparing the results. Output results of algorithm execution are indicated in the figure below. As indicated in this figure, normal shape is ascending as utilization makespan increases. Our result is for the returning times close to the predicted extent, but its growth is more intense as period increases.
Among important indicators existing for scheduling, makespan factor which is equal to subtraction of end time from start time, is very effective in evaluating the method. Here, we compare makespan resulted from existing methods. As indicated explicitly, makespans of non-linear methods increase as the number of cloudlets increase, while proposed methods based on LA are completely declining.

The debt is reduced in this method, since utilization increase is in interaction with debt and exists as a negative index in $|T_i|$. Therefore, utilization increase is accompanied by debts of responses to the tasks in the cloud.
VII. Conclusion

In this research we have proposed an optimized algorithm for scheduling and allocation of virtual machines by using a scheduler learning approach in cloud environment when we don’t consider effects due to dynamic change of cloud resources. This algorithm attempts to identify an optimum behavior for scheduling based on learning automata and its optimization capability to investigate the effect of similar tasks in utilization and especially in times with different task loads. The proposed algorithm called RAOLA increases resource utilization explicitly. On the other hand it relatively optimizes the make span but completely optimizes the debt index of the cloud. Since online scheduling in cloud is of high importance specially in providing existing services in the cloud for users of SLA, we extended the proposed method for online scheduling. The resulted outputs verify the efficiency of the proposed method. Appendixes, if needed, appear before the acknowledgment.

References


