Automated Mechanism To Minimize Sla Violations In Cloud Environment

Gurpreet Singh, Manisha Malhotra, Ajay Sharma

Abstract: Cloud deployment is increasing day by day which will surge the demand of effective resource provisioning and consolidation of virtual machines in data centers. Automatically it also concerns with the quality of service (QoS) and service level agreement (SLA) defined by user. To reduce SLA violations provider has to ensure the appropriate usage of resources which will directly increase the QoS. Due to more usage of cloud, there is a need to address on these issues. This paper presents the mechanism to minimize the SLA violations in cloud environment.

Index Terms: Automated, Cloud, Resource Provisioning, SLA, QoS.

1 INTRODUCTION

This The service provisioning scenario is changing now days. The markets having economic constraints are shifting their business on cloud as on demand orientation. All the stakeholders who are adopting and providing on demand services have different expectations & objectives. To achieve all expectations & objectives of cloud consumers, providers, brokers and end users, there is only one fundamental aspect is responsible i.e. service level agreement (SLA). SLA is a negotiated and formal document which defines terms and conditions of service that is offered to a customer by provider as shown in figure 1.

![Fig. 1. SLA Scenario](image)

All the metrics involved in a SLA should be monitored on regular basis. This is big challenge in cloud computing to ensure QoS without any SLA violations. QoS expects high availability, minimum execution time, cost optimized, reliability and automated service composition. It also includes self-management, configuration, optimization and protecting capabilities which are based on QoS. This paper is mainly focus on two objectives i.e. SLA and QoS. The structure of paper is as follows: first section has described the introduction of paper. Second section will discuss the existing work which has been already done for the above said objectives. Followed by related work, the proposed work will depict in third section. After that experiment section will discuss the outcome of proposed work and finally the conclusion and future work will describe.

2 RELATED WORK

This section presents the work done by eminent researchers, academicians in this field. (Base Paper) the authors have focused VM consolidations which is based multi objectives including energy consumption, reducing VM migrations and ensuring QoS requirement. Double threshold and ant colony system (ACS) approaches used for virtual machine (VM) consolidations, so the proposed algorithm is named as DA-VMC. Double threshold was used for decision making and the mapping relation of ant colony has used for mapping of VMs and host. The experiment of this paper reduced the energy efficiency and guaranteed QoS. In (Beloglazov et al. 2012) discussed three VM selection policies. First policy discussed the minimum migration policy. The first policy is responsible for minimum number of migrations for overloaded host to decrease the CPU utilization. Second is responsible for SLA violations and last is for random selection based on uniformly distributed random variable until the overloaded host will eliminate. In (Singh & Channa, 2016) authors have described an automatic resource provisioning and scheduling algorithm to meet the QoS requirements which was described by user. The algorithm also described the unpredictable environment. Authors (Varalakshmi et al., 2011) have proposed an optimal workflow scheduling framework to meet the user defines constraints of QoS. This algorithm doesn’t focused on cost and energy. Authors (Herbst et al. 2016) proposed a mechanism to optimize the QoS parameters like relative error and SLA violations. But they have not covered the execution time. From the literature it has been observed that some of the authors have focused on SLA violations and some on QoS parameters. Hence there should be a framework which presents both of the issues. Next section will describe the mechanism which is mainly focusing on SLA violations and QoS.

3 PROPOSED WORK

The main objective of this paper is to develop an automated mechanism to minimize the Service Level Agreement (SLA) violations also ensures the Quality of Service (QoS) during virtual machine migration named as ASMQ. The proposed architecture of ASMQ is shown in figure 2. In the proposed architecture numbers of agents are used and description of all agents are describes in table 1.
Whenever the VM becomes overloaded on any host, the MAc (Singh et al. 2017) sends request to directory agent which keeps the information of all hosts and virtual machine of a data center. After checking the records DA forwards the list of destination VMs to MAc. When the cloud mobile agent has the information of source host and destination host, it initiates the migration agent to apply the migration process. While migrating any VM in a host it contains many parameters like amount of CPU processing in the source host, downtime of VM which is going to migrate and total migration time. Firstly, the main task is to select the VM by any selection policy and migration policy in order to minimize the total migration time.

The following parameters are required to apply the migration policy: total ping time, available bandwidth, total size of VM, utilization of VM. At the same time there is a need to check the SLA to ensure the quality. To ensure the SLA, the proposed mechanism contains trust agent which is responsible to meet SLA along with QoS. The mechanism also has the option if during run time there is need to negotiate SLA. Revised SLA again sends to trust agent for verification.

**TABLE - 1**

<table>
<thead>
<tr>
<th>Name of Agent</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Mobile Agent (MAc)</td>
<td>It has the list of VMs on which migration is required. It also keeps the list of destination VM after receiving from Directory Agent.</td>
</tr>
<tr>
<td>Directory Agent (DA)</td>
<td>It keeps the record of all available VMs within the data center and other data center.</td>
</tr>
<tr>
<td>Migration Agent (MA)</td>
<td>It is responsible to migrate all the data from host VM to destination VM with migration policy.</td>
</tr>
<tr>
<td>Trust Agent (TA)</td>
<td>TA ensures the user defined SLA and QoS parameters during the migration process. If not satisfied then it negotiate the SLA if required.</td>
</tr>
</tbody>
</table>

**Fig. 2. Architecture of ASMQ**
The following objective function is used to minimize the SLA violation which leads to ensure maximum level of QoS:

$$\text{Min } \sum \hat{\delta}_j$$  \hspace{1cm} (1)

Here $\hat{\delta}_j$ is SLA violation.

$$\hat{\delta}_j = \frac{1}{\alpha} \sum_{j=1}^{\alpha} \frac{\mu_j}{\nu_j} \times \frac{1}{\beta} \sum_{k=1}^{\beta} \frac{\tau_k}{\psi_k}$$  \hspace{1cm} (2)

$\alpha$ is the total number of hosts in a data center, $\mu_j$ is the time of utilization of resources in a host and $\nu_j$ is the running time of all virtual machines. Virtual machines in a host are indicated as $\beta$. $\tau_k$ is the amount of resources which is not fulfilled during the migration whereas $\psi_k$ indicates the total requirement of resource during the execution of a VM.

The algorithm of proposed mechanism is as below:

Next section discusses the result and experiment of proposed mechanism.

4 RESULTS AND DISCUSSIONS

The proposed algorithm is implemented in cloud sim which provide the simulated environment for cloud computing. For simulation of this algorithm, 50 hosts are taken and more than 200 VMs are deployed on it. The first experiment is setup to check SLA violation rate and calculated by said equation (2). For this different workload has been assigned on VMs. The proposed algorithm is also compared with existing algorithm DAVMC and it gives better results as shown in figure 3. Second experiment is done to calculate the execution time. Due to prior calculation of workload on every VM, the resource utilization becomes optimized. Therefore the waiting time of migration is reduced which also effects on the throughput of algorithm shown in below figure 4.

5 CONCLUSION

This paper has presented an algorithm to meet the SLA and required parameters to maintain the QoS level named as AMSQ. It has used number of agents which will help to reduce the waiting time. All the agents have beforehand calculation of available resources. During migration it is easy to migrate the hosts and virtual machines from source to destination. Hence it reduce the SLA violations and increase the QoS. It has been compared with existing algorithm DAVMC and gives prominent results.
REFERENCES


