Effective Data Center Selection Algorithm for a Federated Cloud

Amol Jaikar\textsuperscript{1,2}, Gyeong-Ryoon Kim\textsuperscript{2}, Seo-Young Noh\textsuperscript{2}

\textsuperscript{1} University of Science and Technology, Daejeon, South Korea.
\textsuperscript{2} Korea Institute of Science and Technology Information, Daejeon, South Korea
{amol, grkim, ryoung}@kisti.re.kr

Abstract. Cloud computing becomes a key factor in the market day by day. It provides the services like infrastructure, platform and software as a service. Therefore, many companies are investing or going to invest in this sector for the development of the data center. These data centers are not only consuming more energy but also producing greenhouse gases. Data center providers are unable to establish data centers all over the world due to the financial limitation. In order to provide better Quality of Service (QoS) and maintain Service Level Agreement (SLA), infrastructure providers come together to provide services. In this federated cloud environment, whenever a new job come, then the central entity selects the data center to execute that job. Selection of the data center plays an important role in a federated cloud environment to improve the throughput as well as performance.

Keywords: Cloud Computing, Data center, Federated Cloud, Energy efficiency.

1 Introduction

Cloud computing offers services for IT as well as non-IT industries. Using infrastructure, platform and software as a service, users can utilize these resources for complex computation, data storage and software execution [4]. The cloud computing services are provided by using virtualization technology like KVM to run virtual machines. This virtual machine executes on a physical machine and tries to complete the job.

There is a possibility of under and over provisioning because researchers do not use these resources all the time which are allocated statically. The dynamic allocation policy with respect to resources can help to avoid over and under provisioning problem. In a federated cloud environment, centralized and distributed management can serve the user effectively. In centralized management, the central or global entity is responsible for the management of all the clusters. This central entity decides the location of the data center as well as the location of the physical machine where the virtual machine is going to deploy. In a distributed management, the central management system selects the data center dynamically but selection of physical machine for a virtual machine is selected by the local entity from the cloud.
The paper is organized as follows. Second section gives the previous related work. Third section describes centralized and distributed techniques. Fourth section explains the proposed methodology. Last section gives the conclusion.

2 Related Work

This section discusses the existing work. The network affects the performance of the distributed application. Network delay, due to long distance and network bandwidth, both can reduce the throughput of the system. The proximity based routing algorithm selects the closest data center based on the network latency. If there are many data centers in the same region then the algorithm selects the data center randomly without considering the performance as well as the cost [1].

Cost effective selection algorithm extends the service proximity based routing by selecting cost efficient data center. This algorithm does not consider performance and availability [2]. Random selection of the algorithm can lead to another problem like high cost required for the execution of these jobs and high response time to complete the jobs. It will also increase power consumption because of always active state.

Round-robin based data center selection algorithm in a single region for Cloud Analyst is proposed [3]. Round-robin technique is used for the selection of the data center as well as a physical machine. This algorithm leads to the problems like under utilization and power consumption.

3 Distributed and Centralized Technique

Two techniques are used to choose the data center in a federated cloud environment in order to support the service. In this section, distributed and centralized technique describes the methodology for selection of a data center. Fig. 1 shows the architecture for centralized as well as distributed technique. The architecture consists of entities like user, global entity, local entity, virtual machine (VM) and cloud. The user is responsible for submission of jobs to the federated cloud.

In the distributed technique, two levels of decisions are taken. First one is to decide the data center. This decision is taken by the global entity. Global entity passes this decision to the local entity. This local entity is responsible for the management of the cloud. This entity decides the location or the physical machine for the deployment of the virtual machine. In this distributed technique, the infrastructure provider can use the resources for another purpose if it is underutilized by central management. The resources are not allocated statically to the central system. Dynamic allocation of resources can be done by the local entity. Disadvantage of this technique is an absence of dynamic policies to improve the performance. Due to the distributed approach, the global entity is unable to make the efficient dynamical policies for better throughput.
In the centralized technique, the global entity is responsible for the allocation of the virtual machine to a physical machine of the data center using centralized policies. The global entity decides the data center as well as the physical machine in a data center. These machines statically allocated to this system. The advantage of this system is that, it can create the dynamic policies to improve the throughput because everything is managed by global entity. Disadvantage is the bottleneck which can reduce the performance.

4 Matrix-based Method

As per observation, centralized system shows many advantages than distributed system. Centralized technique can create dynamic policies which can improve the throughput. The existing algorithms consider only cost or network delay for the data center selection. In this proposed method, the data center is selected based on the matrix values. The matrix is assigned to each region. This matrix contains the information of the cost required for the resources of a data center and distance from the request location. The cost vs location matrix is as shown in Table 1.

<table>
<thead>
<tr>
<th>Region</th>
<th>DC1</th>
<th>DC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance (km)</td>
<td>&lt;50</td>
<td>&lt;100</td>
</tr>
<tr>
<td>Cost</td>
<td>0.1$</td>
<td>0.15$</td>
</tr>
</tbody>
</table>

Performance vs Availability matrix shown in Table 2 contains information about performance and availability of the data center. Performance parameter states the number of jobs completed per unit time while availability parameter states the percentage of availability of resources. The centralized technique has two steps to select the location of the virtual machine. First one is to select the best data center and the second one is to select the physical machine to deploy the virtual machine. Algorithm 1 selects the best data center to execute the job. Algorithm 1 can be used for centralized as well as distributed system approach.
First for loop selects the cost vs distance matrix related to that specific region. Then select some of the data centers from that matrix based on the policy. Second for loop selects the performance vs availability matrix for the same region. Then it again selects data centers from that matrix. Second last step is to select the common data center. Last step is to select the data center from the list of common data centers.

Algorithm 1: Matrix-based Data Center Selection Algorithm

```plaintext
1 while(need) //If the data center selection is required
2 {
3   for i = 0 to n //list of all matrices for each region
4     RegCvL[ ][ ] ← Select the Cost vs Distance matrix related to that region
5   end
6   Set1[ ] ← insert all the candidate data centers from RegCvL[ ][ ]
7   for i = 0 to n //list of all matrices for each region
8     RegPvA[ ][ ] ← Select the Performance vs Availability matrix
9   end
10  Set2[ ] ← insert all the candidate data centers from RegPvA[ ][ ]
11  Set3[ ] ← Set1[ ] ∩ Set2[ ] //get the common elements
12  DC ← Select the best data center from Set3[ ]
13 } //end while loop
```

5 Conclusion and Future work

The proposed algorithm gives the capability to design dynamic policies for performance improvement. The proposed methodology gives the data center with low cost as well as distance and high performance as well as availability. Wrong selection of the data center may lead to increase the cost of the job execution as well as response time due to unavailability of resources or network delay.

Acknowledgement. This work was supported by the National Research Foundation (NRF) of Korea through contract N-13-NM-IR04.

Reference