CHAPTER 5

A DUAL-COST RESPONSIVE ON CLOUD MOBILE SERVICES

5.1 INTRODUCTION

Dynamic Prioritized Load Balanced Round Robin framework is explained to solve the optimization in different elasticity cloud applications. It generally directed on allocating the load to the CPU planning at balancing CPU load rate. Elasticity of the application is most important process in cloud mobile services for successful mapping on the cloud zone. Elasticity in cloud computing is the capability for the mobile services to change its cost conditions in terms of requirements in relation to operational conditions. Flexibility and cost factor plays the significant task to ability planning in cloud computing. Hence Dual-Cost Responsive on Cloud Mobile Services (DRCMS) is introduced. The elastic application is not efficient on the clear use of cloud mobile resources. Providing cost reduce factor on different cloud mobile devices. DRCMS method is implemented for increasing the cloud elasticity applications.

Cost-sensitive elastic scaling method in the cloud mobile devices detects and examines the multi-tier cloud-based bottleneck. Workload-adaptive technique in elastic scaling method reaches the effective cloud elasticity organization. However, fails to allocate and de-allocate resources in a cost effective fashion. Cost-effective Cloud HPC Resource Provisioning by Semi-Elastic Cluster (SEC) model combines all the resource strategies into a unique set on the public cloud. SEC produce lowers cost but deserves bigger wait time. However, on-demand cloud cluster provisioning is not latency-free.
The cost effective ability is the important task on the various cloud mobile services. Several techniques perform elastic application model on cloud mobile devices but fails to run cloud on different device with minimal service cost. To overcome this issues DRCMS model is introduced. In DRCMS mechanism, dual cost represents the leasing and shift cost in the elasticity application. In leasing-cost responsive work, calculation is based on the category of the mobile cloud Virtual Machine (VM) and the amount of virtual machine which is in active condition for processing.

Cloud virtual machine calculates the highest request charge and differentiates each VM individually in the elasticity cloud applications. The shift cost responsive work is introduced DRCE mechanism to compute the shift latency in moving elasticity application. Using Integer Linear Program (ILP) both leasing and shift cost is reduced. Finally, dual cost responsive algorithm is used to examine the two cost criteria for managing the profits up and down in the elasticity of cloud services. The DRCMS with long term result denotes the maximum elasticity rate in cloud mobile devices.

5.2. COST-AWARE MULTI-TIER APPLICATIONS & RESOURCE PROVISIONING ON CLOUD ENVIRONMENT

Cost management is the process of developing and managing the resources of a production. Cost organization is a type of organization accounting that tolerates a business to forecast future expenses to help decrease the change of available over budget. Every small scale to large scale business holder has an aim to reduce costs and increase profits on cloud services.
5.2.1 Enabling cost-aware and adaptive elasticity of multi-tier cloud applications

Rui Han Li et al. (2012) designed a cost-sensitive elastic scaling approach in the cloud mobile devices. Adopting lowers resource allocation costs by detecting the bottlenecks in a class of multi-tier applications. In cloud infrastructure services the users are minimize the charges employing adaptive scaling algorithm. It tolerates to balance their requests simply by bottleneck ties, and execute the construction of an intelligent platform that automates the scaling process. Elastic scaling approach in the cloud mobile devices (ESA-CMD) detects and analyzes the multi-tier cloud-based bottleneck.

The scaling approach contains four key elements for performing capable cloud elasticity management. Initially cost-aware criteria are established to examine the result of cost of resources on each element of response time. A flexible systematic model is developed to confine the performance of multi-tier applications. Then next, workload-adaptive scaling is used to design cost-aware scaling (CAS) algorithm is to hold varying loads of multi-tier requests by flexibly rating up and down bottlenecked tiers within applications.

The next key element is automation of application scaling in that case standard automation scaling requirement is established to describe the properties of the servers. Depending on this specification, the best cost-aware scaling approach essential for a request is naturally calculated and delivered. Finally CAS algorithm is implemented based on the intelligent platform that performs the scaling method of cloud requests on the IC-Cloud infrastructure using execution and experimental calculation element.
A multi-tier application consists of two parts. First part is the server set $S$, together with all servers of the application and second part is the demand set $D$, confining the requirements specified in the SLA. The multi-tier architecture promises the modularity of cloud applications and facilitates the control of their tiers. An application’s server set is separated into several subsets and each subset consists of servers belonging to the similar tier.

Each server is marked by a single tier id. In a multi-tier application, servers are classified into various tiers according to their functionalities. Each application has a set of demands and checks specified by the application owner in the form of a SLA. A performance requires are determined by the maximum end-to-end response time for a demand. The maximum user budget was not specified explicitly. However the condition was increasing users’ cost for meeting the performance target.

5.2.2. Cost-effective cloud hpc resource provisioning by building semi-elastic virtual clusters

Shuangcheng Niu et al. (2013) developed a Semi-Elastic Cluster (SEC) computing model which tolerates organizations to operate a set of cloud requests. Semi elastic model also maintains schedule jobs inside the current capacity, with the elasticity of dynamically correcting the capacity level. SEC can distribute considerably improved cost-effectiveness without trouble of user experience.

Semi-Elastic Cluster (SEC) model allows the capacity of a virtual cluster to rise by effectively securing/releasing cloud instances. In SEC model provisioning-enabled
scheduling algorithm faces the trade-off among receptiveness and cost (overall monetary cost of executing a batch of parallel jobs). Semi-Elastic Cluster present and calculate an approach that efforts to reduce the cost.

**Instance Acquisition:** related calculations are used to trigger the request acquisition need. It’s prompt that the checking for the possibility of transmitting a new job (i.e. job arrival or completion). Only these events can allow the queue condition and the expected transmit time of waiting jobs. After expected scheduling measures like job transmit, priority informs backfilling and queue modification.

**Instance Release:** different for acquisition, request release is confirmed periodically following job scheduling. Considering Amazon's hourly costing model, it creates additional economic sense to achieve delayed release. An idle request is freed simply at the stop of its hourly distribution term, when it is regarding to expire. Such episodic release confirms is achieved except when the job queue is not vacant (which implies that all inactive nodes are marked preserved for a waiting job).

The cost-responsiveness tradeoff in accepting SEC and evaluate it with replacement forms of cloud HPC. Most important simulated modes are described further. At first Individual mode, user independently acquires virtual clusters and run jobs, single user take improvement of the reserved requests.

Then next is no wait a naive workload gathering form, an inexpert workload collection mode, where the cluster directly request on-demand resources. Each time an entering job does not enclose sufficient nodes to start, virtually eliminating the job
queue. Finally SEC contains difference within SEC On-Demand using only on-demand requests and SEC-Hybrid using the entire reserved requests classes for calculating wait time and cost rate.

The hybrids produce higher cost saving compared to individual mode. Both SEC modes essentially decrease the average job wait time of individual through request reuse. No Wait has considerably lesser average job wait time, due to its combination of instant job transmit and instance reuse. Its cost is nearer to particular than SEC-Hybrid. However, SEC does not predict the overall instances within the time interval.

5.3 DUAL-COST RESPONSIVE ON CLOUD MOBILE SERVICES

An efficient DRCMS Mechanism is developed to minimize the cost factor on the different cloud mobile devices. The dual cost indicates the leasing cost and shift cost. In leasing-cost, the computation is based on mobile cloud Virtual Machine (VM) and it calculates the type of the virtual machine and which machine is in active condition for further processing. In shift cost work, the shift latency is reduced in moving elasticity application. Finally, the DRCMS algorithm analyzes the two cost principle for directing the profits up and down in the elasticity cloud services. The long term result in DRCMS algorithm represents the higher flexibility rate in cloud mobile devices. The overall diagram of DRCMS is shown in figure 5.1

The below figure 5.1 illustrates the architecture diagram of DRCMS. Figure 5.1 shows the elasticity application on various cloud mobile services. The two cost responsive model is explained for increasing the elasticity application. The shift cost responsive model is employed to reduce the latency.
Whereas, the leasing cost responsive model verify the category of virtual machine is used and which machine in an active condition. The cloud virtual machine also determines the highest request rate and significantly divides the each virtual machine in elasticity cloud applications.
5.3.1. Dual-cost responsive mechanism

A dual cost reactive method that offers the support for elasticity application in the cloud that reduces the cost factor. The two cost mechanism is described as following subsection.

5.3.2. Leasing cost responsive model

Leasing cost responsive model in DRCMS regards the cost factor of all mobile service within the particular interval in order to calculate the cloud virtual machine. A lease is an arrangement calling for the person who’s right to employ the applications for leasing costs to where the mobile application is accessed from the cloud with efficient cost management. The internet cloud platform offers support for virtual machine (VM) operation and it also shared with elastic conditioning. In leasing cost responsive model, computation is based on number of the mobile cloud virtual machine examines which is an active condition for calculating the future requirement. The least cost model in DRCMS involves two steps. For each type of cloud server, compute the highest request rate that the server can service at a stage and also evaluates a leasing-cost arrangement of servers that consists of maximum ability. The virtual machine diagram is illustrates in figure 5.2.

Virtual machines are produced by hardware virtualization or software emulation methods. Virtualization software creates and handles number of virtual machines on one physical host. A virtual machine (VM) is the software performance of a machine is an
essential model and established by the improvements of resource sharing, cost efficiency on the different cloud services.

![Diagram of Virtual Machine](image)

**Figure 5.2 Diagram of Virtual Machine**

A VMM (virtual machine monitor) (i.e. hypervisor) is used as an intermediate to present concept from physical resources. VM operates as a physical computer, also consists of virtual RAM, CPU, NIC and hardware disk. The DRCMS utilizes a systems approach to conditionally derive highest request rate of different server categories by running the application on hardware arrangements. DRCMS placed to slowly rising simulated workload and determining the point where the server dropping the highest request.

Let us consider a cloud platform with N various categories of server. $R_i$ is the highest request rate and $q_i$ represents the leasing cost of the server type indicating (i). Let $\Delta$ represents the maximum workload request rate for which highest request rate requires to be arranged. The difficulty of lease cost is estimated by using the following equations.
\[ \sum_{i=1}^{N} M_i q_i \quad \text{......... Eqn (5.1)} \]

\[ \sum_{i=1}^{N} M_i R_i \geq \Delta \quad \text{......... Eqn (5.2)} \]

In equation (5.1) and (5.2) \( M_i = m_1, m_2, ..., m_n \) is the number of cloud server of each type that is preferred. Lease cost problem is optimized using ILP. The main objective of DRCMS mechanism is attained using the Leasing cost responsive model. In which the Cloud VM calculates the highest request rate and differentiates each VM independently in the elasticity cloud applications.

### 5.3.3 Shift cost responsive model

The shift cost responsive approach is designed to address the scenario where the shift latency gained in moving the application to new configuration is optimized. Hence responsive model is to calculate the latency using replication on the VM while the application is in implementation process. The Shift cost responsive method is to compute the latency of using various conditioning mechanisms such as replication, active moving and resizing. Local resizing uses the hypervisor on a machine for varying the resource distribution of a virtual machine. Local resizing is efficiently completed with least overheads.

Initiating the new request (replica) of an application involves replication the virtual machine picture of the application from essential storage to the disk on the original distributor. Starting up the OS and the application request and recreate the
application to construct it aware of the other request. The shift latency is estimated as follows,

\[ \frac{\text{DI}_{\text{size}}}{B} + t \]  

\[ \ldots \ldots \text{Eqn (5.3)} \]

In equation (5.3), \( \text{DI}_{\text{size}} \) denotes the size of the disk image, \( B \) denotes the network bandwidth accessible for the copying the virtual machine. Then, in equation ‘\( t \)’ is a constant representing the virtual machine and application startup time.

The active moving is passed out of a virtual machine from one machine to another absorbs copying the memory state of the VM to new server while the request is running in DRCMS. In general, the active moving methods imagine that the disk state of the VM is conserved on allotted data system. Hence, the latency of the active moving is calculated with the help of equation (5.4).

\[ \text{Latency} = \tau \frac{S}{B} \]  

\[ \ldots \ldots \text{Eqn (5.4)} \]

From above equation (5.4), latency is calculated, Here, \( S \) denotes the size of the virtual machine RAM, \( B \) represents the network bandwidth presented for the copying operation. Then, \( \tau \) is a constant that captures the mean number of times a memory page is shift over the network. Therefore, DRCMS decreases the copying overheads (i.e.) latencies.
Delaying the applications are used for performing movement in a public cloud, changing its disk state into a novel machine image, copying the machine image to a novel server and start over the image on the novel virtual machine.

Since the disk state can require to be copied two times, once to create a new machine image and then copy the machine image to other type of server. Then the latency of this approach is expressed as,

\[
\text{Latency} = 2 \times \frac{\text{Dsize}}{B} + t
\]

\[\text{........ Eqn (5.5)}\]

The objective of shift cost responsive method is to reduce latency through preferring mechanisms that acquire lesser copying overheads. Shift cost responsive method is prepared and solved as an ILP optimization problem. Both leasing and shift cost is optimized using ILP.

Let as consider \(N\) denotes the number of cloud server types continued by the cloud platform. Let \(q_i\) indicate the leasing cost for server type \(i\) and \(R_i\) denotes its highest request rate and \(\Delta\) represents the peak workload request for which the application involves to be provisioned. Then \(T\) indicates the maximum amount of servers that could be necessary to satisfy \(\Delta\). Let \(V\) denote the number of the conditioning mechanisms maintained by the active moving.
Then the objective function for reducing leasing cost is calculated as,

\[
\min \sum_{i=1}^{N} \sum_{j=1}^{T} \sum_{k=1}^{V} q_i y_{ijk}
\]

\[\text{......... Eqn (5.6)}\]

\[
\sum_{i=1}^{N} \sum_{j=1}^{T} \sum_{k=1}^{V} y_{ijk} R_i \geq \Delta
\]

\[\text{......... Eqn (5.7)}\]

\[
\sum_{i=1}^{T} \sum_{j=1}^{N} \sum_{k=1}^{V} y_{ijk} = 1 \text{ for all } i
\]

\[\text{......... Eqn (5.8)}\]

The terms \(y_{ijk}\) are an integer variable that can get values of 0 or 1. A value of 1 denotes that server ‘i’ is changed into ‘j’ using a responsive mechanism k (e.g., moving and copying). A value of 0 specifies that that choice was not preferred by the integer linear program.

The production of the integer linear program is set of value \(y_{ijk}\) that denotes which server categories are preferred and also specifies a plan for shifting for each server \(i\) to the new server type \(j\) using method \(k\).
The shift responsive designed for integer linear program for the optimization criteria which must minimize the shift cost and thus Equation (5.5) changes to following equation.

\[
\min \sum_{i=1}^{T} \sum_{j=1}^{N} \sum_{k=1}^{V} m_{ijk} y_{ijk}
\]

………… Eqn (5.8)

In equation (5.8) \(m_{ijk}\) is the rate of converting distributor \(i\) to distributor \(j\) using mechanism \(k\). The shift responsive is determined using the above equations that the overhead of replication and active moving is minimized. The algorithmic step of the dual cost responsive is described in below algorithm 5.1.

```plaintext
Input: Set of servers ‘S’, shift latency,
Output: cost aware mobile services to cloud infrastructure
Step 1 : Select the set of servers which perform the mobile services on cloud zone
Step 2: Select the resources
Step 3: cloud virtual machine calculates the maximum request
Step 4: If satisfy the maximum workload request rate then
Step 5: The problem of leasing cost is calculated using equation (5.1) and (5.2)
Step 6: Evaluate the shift latency using equation (5.3)
Step 7: The latency of the active moving is calculated using equation (5.4)
Step 8: Perform minimum leasing cost using equation (5.6)
Step 9: Minimize the shift cost using equation (5.9)
End if
End
```

Algorithm 5.1 Dual Cost Responsive algorithm

112
Dual Cost Responsive algorithm examines the two cost criteria for guiding the profits up and down in the elasticity cloud services. At first, the numbers of server are preferred and it achieves the mobile services on cloud zone and selects the cloud resources. Based on the category of the mobile cloud Virtual Machine (VM) computation is derived in least cost work.

The cloud VM decides the highest request rate. The maximum rate is satisfying the peak workload maximum condition, and then the leasing cost is evaluated. In addition, active moving latency of the cloud server and shift cost latency is calculated. Finally, the shift cost and leasing cost is reduced using integer linear program. This aids to reduce the shift cost and leasing cost responsive ratio in DRCMS. The Dual Cost Responsive algorithm with long term result provides higher flexibility rate in cloud mobile devices.

5.4. EXPERIMENTAL EVALUATION DRCMS MECHANISM

Dual-Cost Responsive on Cloud Mobile Services (DRCMS) is executed in java language with CloudSim simulator environment. The CloudSim simulator performed on Cloud environment offers different resource configurations for a number of virtual machines. Each virtual machine is configured with an amount of memory, CPUs and cost. The particular toolkit is chosen as a simulation platform as it is a present simulation structure in Cloud environments. The CloudSim requires minimum cost and instance to perform Cloud-based request provisioning test environment.
The DRCMS with organized data calculation and information sharing using Amazon. The familiar and generally recognized cloud service provider simulates the dynamic benchmarking approach on Amazon using on-demand cloud. It is compared with existing methods namely, ESA-CMD developed by Rui Han Li et al. (2012) and Semi Elastic Cluster method introduced by Shuangcheng Niu et al. (2013). The performance of the DRCMS mechanism is measured in terms of:

i) Leasing Cost Ratio

ii) Throughput

iii) Latency Time

iv) Flexibility Rate

5.5 PERFORMANCE ANALYSIS OF DRCMS MECHANISM

The proposed DRCMS mechanism is compared against with the two existing methods. The compared existing methods are namely, ESA-CMD developed by Rui Han Li et al. (2012) and Semi-Elastic Cluster (SEC) computing model developed by Shuangcheng Niu Li et al. (2013). The experimental effects using CloudSim simulator are related and evaluated through table and graph structure given below. To evaluate the DRCMS mechanism, the following metrics are used.

5.5.1. Performance of Leasing Cost Ratio

The DRCMS considers the cost factor of each mobile device within the particular period to calculate the cloud virtual machine. The leasing cost ratio is determined based
on the number of cloud server increased with number of highest request. It can calculated as follows,

\[
\text{LCR} = \sum_{i=1}^{N} \text{cloud server}_i \ast \text{request}_i
\]

\[\text{Eqn (5.9)}\]

From equation (5.9), LCR represents the leasing cost ratio. Lower the leasing cost ratio more efficient in DRCMS method. It is measured in terms of percentage (%).

Table 5.1 Tabulation of Leasing Cost Ratio

<table>
<thead>
<tr>
<th>Number of cloud server</th>
<th>Leasing cost ratio (%)</th>
<th>Existing ESA-CMD</th>
<th>Existing SEC</th>
<th>Proposed DRCMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>32.56</td>
<td>29.3</td>
<td>25.36</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>35.64</td>
<td>32.32</td>
<td>28.45</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>38.69</td>
<td>34.68</td>
<td>29.32</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>39.75</td>
<td>35.21</td>
<td>31.48</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>41.32</td>
<td>37.45</td>
<td>33.47</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>43.17</td>
<td>39.58</td>
<td>34.23</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>44.98</td>
<td>41.21</td>
<td>36.84</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>46.74</td>
<td>43.56</td>
<td>38.46</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>47.93</td>
<td>44.12</td>
<td>39.14</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>49.28</td>
<td>45.36</td>
<td>41.25</td>
<td></td>
</tr>
</tbody>
</table>
Table 5.1 explains the experimental values of Leasing Cost Ratio using DRCMS mechanism, ESA-CMD method and SEC model. The proposed DRCMS mechanism considers different number of cloud servers for experimental evaluation that ranges from 5 to 50 cloud servers.

The performance of proposed DRCMS mechanism is compared with existing two methods namely, ESA-CMD developed by Rui Han Li et al. (2012) and Semi-Elastic Cluster (SEC) computing model developed by Shuangcheng Niu Li et al. (2013). From the table value, it is descriptive that the leasing cost ratio using proposed DRCMS Method is reduced as compared to other existing methods.

![Figure 5.3 Measure of Leasing Cost Ratio](image)

**Figure 5.3 Measure of Leasing Cost Ratio**

Figure 5.3 shows the presentation values of least cost ratio with respect to number of cloud server. As shown in figure, the proposed DRCMS Method using leasing
cost ratio provides better performance as compared to other existing methods namely ESA-CMD developed by Rui Han Li et al. (2012) and Semi-Elastic Cluster (SEC) computing model developed by Shuangcheng Niu Li et al. (2013). While growing the number of cloud server, the leasing cost ratio is increased in all three methods. But reasonably the leasing cost ratio is minimized using DRCMS Mechanism.

This is because DRCMS mechanism using the Leasing cost reactive model in which the Cloud VM calculates the highest request rate and separate each VM independently in the elasticity cloud applications. By increasing the cloud server and maximum customer request, the leasing cost ratio is decreased by 25% compared with the ESA-CMD developed by Rui Han Li et al. (2012).

In addition, the cloud VM calculates the highest request rate and it is assuring the peak workload maximum condition. Due to this condition, DRCMS mechanism reduces the leasing cost ratio by 13% compared with the Semi-Elastic Cluster (SEC) computing model developed by Shuangcheng Niu Li et al. (2013).

5.5.2. Performance analysis of Throughput

The cloud network throughput is described as the average rate of information distribute through a cloud environment. Through put a rate the users accept for their request. It is measured in terms of kilo bits per second (KB/sec).
Table 5.2 Tabulation for throughput

<table>
<thead>
<tr>
<th>User Requests</th>
<th>Throughput (KB/sec)</th>
<th>Existing ESA-CMD</th>
<th>Existing SEC</th>
<th>Proposed DRCMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>136</td>
<td>110</td>
<td>162</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>168</td>
<td>132</td>
<td>198</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>196</td>
<td>156</td>
<td>245</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>231</td>
<td>178</td>
<td>264</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>248</td>
<td>185</td>
<td>298</td>
<td></td>
</tr>
<tr>
<td>120</td>
<td>256</td>
<td>212</td>
<td>323</td>
<td></td>
</tr>
<tr>
<td>140</td>
<td>287</td>
<td>254</td>
<td>336</td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>314</td>
<td>263</td>
<td>356</td>
<td></td>
</tr>
<tr>
<td>180</td>
<td>345</td>
<td>289</td>
<td>387</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>367</td>
<td>332</td>
<td>396</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.2 describes the network throughput measurement depended on the number user request ranges from 20 to 200. The throughput presented for an elasticity application gets increase when the user request for particular application also increases. From the table value, it is descriptive that the throughput value using proposed DRCMS method is higher, when compared to other existing methods.
Figure 5.4 illustrate the throughput rate based on the number of customer demand in cloud zone. Our proposed DRCMS mechanism performs relatively well when compared with the existing ESA-CMD developed by Rui Han Li et al. (2012) and Semi-Elastic Cluster (SEC) computing model developed by Shuangcheng Niu Li et al. (2013). DRCMS mechanism had better changes where the workload in the cloud zone with mobile services changes quickly that aids to simply cover the total customer demand with improved throughput rate.

Moreover, the leasing and shift cost is reduced using integer linear program in proposed DRCMS mechanism. This helps to increases the network throughput by 15% when compared with the ESA-CMD method by Rui Han Li et al. (2012) and 30% when compared with SEC model by Shuangcheng Niu Li et al. (2013) respectively.
5.5.3. Performance analysis of Latency Time

The performance of latency time in DRCMS mechanism is time for a virtual machine from one device to another involves copying the memory state of the virtual machine. It involves time taken for user request and response from the users. Latency time is measured in terms of milliseconds (ms).

\[
\text{Latency Time} = \sum_{i=1}^{n} \text{Time (Req}_i\text{)} \ast \text{Time (response)}
\]

…… Eqn (5.10)

From equation (5.10) the latency time ‘LT’, is measured by considering the requests being made ‘Req\_i’ and the response ‘response’ for the corresponding request.

**Table 5.3 Tabulation for Latency Time**

<table>
<thead>
<tr>
<th>User requests</th>
<th>Latency Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Existing ESA-CMD</td>
</tr>
<tr>
<td>20</td>
<td>0.49</td>
</tr>
<tr>
<td>40</td>
<td>0.58</td>
</tr>
<tr>
<td>60</td>
<td>0.74</td>
</tr>
<tr>
<td>80</td>
<td>0.88</td>
</tr>
<tr>
<td>100</td>
<td>0.96</td>
</tr>
<tr>
<td>120</td>
<td>1.14</td>
</tr>
<tr>
<td>140</td>
<td>1.23</td>
</tr>
<tr>
<td>160</td>
<td>1.36</td>
</tr>
<tr>
<td>180</td>
<td>1.52</td>
</tr>
<tr>
<td>200</td>
<td>1.68</td>
</tr>
</tbody>
</table>
Table 5.3 illustrate the latency time using DRCMS Method, ESA-CMD method and SEC model. In order to measures the latency time, user request is considered in the range 20 to 200. The performance of proposed DRCMS Method is compared with existing two methods namely, ESA-CMD developed by Rui Han Li et al. (2012) and Semi-Elastic Cluster (SEC) computing model developed by Shuangcheng Niu Li et al. (2013). From the table value, it is descriptive that the latency time using proposed DRCMS Method is reduced as compared to other existing methods.

Figure 5.5 shows the latency measurement of planned DRCMS mechanism compared with the existing ESA-CMD developed by Rui Han Li et al. (2012) and Semi-Elastic Cluster (SEC) computing model developed by Shuangcheng Niu Li et al. (2013). In DRCMS mechanism, the active moving is achieved of a virtual machine from one mechanism to another absorbs copying the memory state of the VM to other type of server through the application process. As a result, the latency time is reduced by 51% when compared with the ESA-CMD method by Rui Han Li et al. (2012).
In addition, the effective dual cost responsive mechanism significantly minimizes the latency time by 26% when compared with SEC model by Shuangcheng Niu Li et al. (2013) respectively.

5.5.4. Performance analysis of Flexibility Rate

Flexibility rate is measured based on shift cost and leasing cost of user request. Flexibility rate calculated in terms of percentage (%). If higher flexibility rate occur the method is said to be more efficient.

Table 5.4 Tabulation for Flexibility rate

<table>
<thead>
<tr>
<th>User requests</th>
<th>Flexibility rate (%)</th>
<th>Existing ESA-CMD</th>
<th>Existing SEC</th>
<th>Proposed DRCMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td></td>
<td>63.21</td>
<td>69.47</td>
<td>75.89</td>
</tr>
<tr>
<td>40</td>
<td></td>
<td>65.38</td>
<td>71.24</td>
<td>77.28</td>
</tr>
<tr>
<td>60</td>
<td></td>
<td>67.02</td>
<td>73.21</td>
<td>79.46</td>
</tr>
<tr>
<td>80</td>
<td></td>
<td>69.77</td>
<td>75.64</td>
<td>81.23</td>
</tr>
<tr>
<td>100</td>
<td></td>
<td>71.54</td>
<td>77.32</td>
<td>82.64</td>
</tr>
<tr>
<td>120</td>
<td></td>
<td>72.13</td>
<td>78.12</td>
<td>84.21</td>
</tr>
<tr>
<td>140</td>
<td></td>
<td>72.89</td>
<td>79.34</td>
<td>85.77</td>
</tr>
<tr>
<td>160</td>
<td></td>
<td>74.32</td>
<td>80.54</td>
<td>86.94</td>
</tr>
<tr>
<td>180</td>
<td></td>
<td>76.45</td>
<td>81.47</td>
<td>87.64</td>
</tr>
<tr>
<td>200</td>
<td></td>
<td>78.31</td>
<td>83.21</td>
<td>88.56</td>
</tr>
</tbody>
</table>
Table 5.4 shows the flexibility rate using DRCMS Method, ESA-CMD method and SEC model. From the table value, it is descriptive that the flexibility rate using proposed DRCMS Method is increased as compared to other existing methods.

![Figure 5.6 Measure of flexibility rate]

Figure 5.6 illustrates the Flexibility rate of DRCMS mechanism as compared with the existing ESA-CMD developed by Rui Han Li et al. (2012) and Semi-Elastic Cluster (SEC) computing model developed by Shuangcheng Niu Li et al. (2013). From the figure it is clear that the most considerable improvements arise in the three methods and comparatively the flexibility rate is higher in our DRCMS mechanism. By applying the Dual-Cost Responsive algorithm in proposed DRCMS mechanism, the shift cost and leasing cost are reduced. The DRCMS mechanism with extensive result provides improved flexibility rate in cloud mobile devices. The flexibility rate of DRCMS
mechanism is increased by 14% when compared with the ESA-CMD method by Rui Han Li et al. (2012). In addition, the dual mechanism such as leasing cost and shift cost responsive model provides Cost-aware Elasticity application in the Cloud with higher flexibility rate by 7% when compared with SEC model by Shuangcheng Niu Li et al. (2013) respectively.

5.6. SUMMARY

An effective DRCMS Mechanism is implemented for improving the elasticity on cloud services. The DRCMS Mechanism executing various categories of requirements from different cloud mobile environments. In DRCMS technique there are two types of cost provisioning method is developed to minimize the cost factor on different cloud mobile devices. In leasing-cost responsive work, calculation is based on the type of the mobile cloud Virtual Machine (VM) and the amount of virtual machine which is in active condition for processing. Cloud VM computes the highest request rate and differentiates each VM independently in the elasticity cloud applications. In shift-cost responsive work, shift latency in dynamic elasticity application is reduced. Both shift and leasing cost is reduced using integer linear program. The dual cost responsive algorithm with extensive results verifies the higher flexibility rate in cloud mobile devices.