CHAPTER 3
AN EFFECTIVE ENERGY-POWER APPROXIMATION USING MACHINE FLOW ON ELASTIC CLOUD SERVICES

3.1 INTRODUCTION

Elasticity applications available in cloud computing environment presents cloud mobile services to process an effective mapping request on the cloud zone. Cloud mobile services guarantee flexible services for various ranges of business customer conditions and it allocated from different infrastructure forms. The cloud mobile computing policy describes the algorithms on different QoS expectations such as energy, power, CPU load, memory and cost.

Energy-aware resource allocation heuristics method offered a substantial cloud services for the resource providers and consumers. However, QoS probability and the procedure of power quality of the devices fail to improve the energy-efficient management on the elastic cloud computing environments. The energy efficient multi resource allocation of virtual machine improves energy efficiency. However, the virtual machine consumed high power due to the unstable performance of the device.

Machine Flow based Energy-Power Approximation (MFEPA) method is introduced to achieve energy efficient system for the cloud mobile services. Two objectives are attained for construction of MFEPA algorithm. Initially, with the application of Multi-grid approximation technique, the performance of energy consumption is reduced. Next, similarly power consumption is minimized by using look-ahead control on mobile cloud services. Here, coarser structure is decreased with the
explanation of Multiple grid (i.e.,) machines is mapped back to the inventive grid. The mapping technique decreases the energy usage and proves to be effective on the terminal mainframe mobile communications. The Look-ahead control in proposed MFEPA method decreases the power utilization on the wireless interface.

3.2 ENERGY AWARE RESOURCE ALLOCATION & MULTISOURCE ALLOCATION FOR ELASTIC CLOUD COMPUTING

The energy utilization of resources, mainly in a cloud environment, accounts for an extensive number of the certain energy use. Essentially, a resource allocation task that takes into account resource consumption would lead to better energy efficiency. This, in clouds expands further with virtualization technology in that tasks can be simply consolidated. Task consolidation is a useful method to increase resource utilization and in turn decrease energy consumption.

3.2.1 Energy aware resource allocation heuristics for efficient management of data centers

Anton Beloglazov Li et al. (2012) designed an architectural framework and principles for energy-efficient Cloud computing. The intention of energy resource allocation is manage the energy-aware resource and builds efficient plans and algorithms for virtualized data centers. This in turn improves the energy efficiency of the data center and reduces the cost of software engineering.
3.2.1.1 Green cloud architecture

Cloud architecture constructs the aim of the future generation data by building networks of virtual services. Hence users can access and organize operations against anyplace in the world on require at aggressive costs calculating on their QoS. Four main things are involved in green cloud architecture.

1. Consumers/Brokers:

Service requests are suggested from anyplace in the globe by Cloud consumers or brokers. Service requests is important to concern that can be a change between Cloud consumers and users of utilized services.

2. Green Service Allocator:

Perform the boundary between Cloud infrastructure and consumers. Green service allocator requires the contact of the consequent factors to maintain the energy-efficient resource management.

   (a) Green Negotiator: Green negotiator finalize the Service Level Agreement (SLAs) with particular prices and penalty (for violations of the SLAs) between the Cloud provider and consumer based on the consumer’s QoS requirements.

   (b) Service Analyzer: It understands and analyzes the service requirement of a submitted request before accepting it.
(c) **Consumer Profiler**: It collects specific characteristics of consumers. Hence important consumers can be granted special licenses and prioritized over other consumers.

(d) **Pricing**: Pricing is fully responsible for service requirements charged to maintain the supply and command of computing resources and assist in prioritizing service allocates efficiently.

(e) **Energy Monitor**: Energy monitor gives information to the VM commander to build energy-efficient resource allocation results.

(f) **Service Scheduler**: A Scheduler makes decision on resource power for the allocated virtual machines and distributes their request to other virtual machines.

(g) **VM Manager**: Keeps path of the accessibility of (Virtual Machines) VMs and their resource custom.

(h) **Accounting**: It checks the authentic usage of resources by VMs and accounts for the resource usage costs.

3. Virtual Machines:

According to incoming requests several VMs can be dynamically started and stopped on a particular physical machine. Several VMs can along through run requests depending on assorted operating system environments on a limited physical machine. By dynamically drifting VMs across physical machines, workloads can be combined and unused resources can be switched to a low-power mode, turned off or configured to control at low-performance levels (e.g. using DVFS) in order to save energy.
4. Physical Machines:

The basic physical computing servers present the hardware infrastructure for creating virtualized resources to gather service demands.

5. Power model

CPU, memory, disk storage and network interfaces are used to determine the power usage by computing nodes in data center. The CPU consumes the main part of energy when compared to other system resources, and hence power model construction is focus on maintain its power utilization and competent usage. Additionally the CPU expenditure is normally equal to the overall system load.

The application of DVFS on the CPU effects in approximately linear power-to-frequency relationship for a server. The limited numbers of states are placed in the CPU for considering the frequency and voltage level. Though, the system components are used for the separation of components from CPU.

In addition these studies have shown that on typical an idle server consumes more power by the server organized at the complete CPU speed. This reality validates the method of controlling idle servers to the sleep manner to decline the whole power usage.

3.2.1.2 Energy-aware allocation of data center resources

Modern growths in virtualization have resulted in its creation across data centers. By supporting the association of VMs among physical nodes, it enables dynamic migration of VMs according to the concert requirements. The VMs do not use all the provided resources. When server can be logically resized and combined to the smallest
amount of physical nodes. Even as idle nodes can be switched to the sleep mode to eliminate the idle power consumption and cut the total energy consumption by the data center. At present, resource allocation in a Cloud data center aims to supply high performance at the same time as meeting SLAs, without focusing on assigning VMs to minimize energy usage.

To investigate both performance and energy efficiency, three critical issues must be addressed. First, unnecessary power managing of a server may possibly cut its reliability. Second, turning resources off in a dynamic environment is hazardous from the QoS perspective. Due to the changeability of the workload an intrusive consolidation, a few VMs may not attain required resources under peak load, and unsuccessful to meet the desired QoS. Then next, ensuring SLAs brings challenges to precise application performance organization in virtualized environments. Above problems understand effective consolidation procedures that can minimize energy utilization without adjusting the user-specified QoS requirements. The energy-aware resource allocation algorithms control the energy consumption. However, QoS expectations and power usage characteristics of the devices fails to enhance the energy-efficient management on the elastic cloud computing environments.

3.2.2 Energy Efficient Multisource Resource Allocation of Virtual Machine Based on PSO in Cloud Data Center

An-ping Xiong Li et al. (2014) presents an energy efficient virtual machine allocation algorithm based on energy efficient multi resource allocation model and the particle swarm optimization (PSO) method. This in turn decreases the energy utilization
in cloud data center. The fitness function of PSO in Multi Resource Energy Efficient Allocation model based on Particle Swarm Optimization (MREE-PSO) algorithm is to determine the optimal point between resource utilization and energy consumption.

In cloud data center, several servers are allowed to be combined to one physical node as VMs by virtualization. A number of VMs are also run on a single physical node. Virtual machine allocation algorithm is a relationship between energy consumption and resource utilization which focuses on CPU and disk to minimize energy consumption. Here, the authors investigated the impact of workload consolidation on the energy-per-transaction based on both the CPU and disk exploitation.

Based on the model and the particle swarm optimization (PSO) the MREE-PSO algorithm can be separated into three parts. Initially particles are generated by FF (First Fit) algorithm. Then next, dependence functions of individual optimal solution and the global optimal elucidation of particles and then fitness function of PSO. Individual optimal solution avoids falling into local optimal results. In cloud data center it also creates the utilization of logical system resources.

PSO algorithm usually applied in different applications. Compared with other methods, it is established that PSO obtains improved solutions in a lower cost and quicker way. In PSO, the possible results are called particles which fly during the elucidation space by subsequent the current optimum particles. Particles maintain the part of their previous state because server have memory.
All particles preserve their individuality in any case, although server share the equal point in belief space with no limitation. Each particle has an original random velocity, and particle’s association is influenced by individuality and sociality weighted factors. Individuality is described as the tendency to arrival to the particle’s best previous position and sociality is defined as the tendency to progress towards the neighborhood’s best previous position.

At each time step, the particle swarm optimizations vary the velocity of each particle toward its $p_{best}$ and $g_{best}$ locations. Velocity is managed by a random time, with particular random values formed for acceleration against $p_{best}$ and $g_{best}$ locations. Additionally, particle swarm optimization is an approach which can be used for particular functions focused on a different requirement. Particle swarm optimization is generally used because there are few factors to adjust. However energy efficient multi resource allocation consumes high power when number of virtual machines is used.

### 3.3 MACHINE FLOW BASED ENERGY-POWER APPROXIMATION ON ELASTIC CLOUD SERVICES

The objective of MFEPA method is to provide energy-power saving mobile services on elastic cloud services. MFEPA method contains two phases. Initially the MFEPA method develop a Multi-grid approximation technique for decreases the energy consumption level on cloud mobile services. Next, it implements look ahead control to reduce the power consumption on mobile cloud services.
Figure 3.1 Architecture Diagram of Machine Flow based Energy-Power Approximation method

Figure 3.1 shows the architecture diagram of MFEPA method where the mobile applications are located in the internet cloud. Multi-grid approximation technique is designed on each mobile service for a certain particular time intervals to estimate the multi resources on mobile services. Multi-grid approximation technique obtains the resolution with minimal energy consumption by replacing the position of mobile devices. The solution of negligible energy utilization in multi-grid approximation technique constructs a result of restoring the position of mobile devices.

Then, next is used to minimize the power utilization on mobile cloud services by using look-ahead control in MFEPA system. The look-ahead control in MFEPA method is used for reduces the power usage and improves the profit margin rate through
weighting factor to all the mobile cloud users. In MFEPA method, the elastic cloud service contains mobile cloud devices and internet cloud applications where the mobile application is accessed from the cloud with successful resource management. The construction of multi-grid approximation in MFEPA method decreases the coarser construction. In addition to that, the multi-grid controls the time period continuously to optimize the system performance with the minimal energy utilization.

In MFEPA method workload of the mobile services changes on the basis of time period with the application of look-ahead control (i.e. varies quickly) that makes high performance with optimized power control. Look-ahead control is combined with multi-grid approximation technique to attain efficient machine flow with least power and energy consumption. The reduction of the energy and power rate reaches higher profit margin on the Infrastructure as a Service (IaaS) depends on business providers.

### 3.3.1 Multi-grid Approximation Technique

The multi-grid approximation in MFEPA method performs resource approximation on mobile services with the consideration of workload of each mobile service in particular time interval. Let us consider the set of mobile services in MFEPA method with ‘n’ active cloud servers in the internet cloud zone. The mobile services in the internet cloud zone assigns the resource depending on the request count with the request count denoted in matrix set.

In MFEPA method, let the matrix $M_{i,j}$ represents the resources allocated from services ‘i’ to server ‘j’. Here the service ‘i’ indicates the service request path whereas ‘j’
denotes the server path which allow the request points. Using MFEP method in the form of matrix set with multiple grids to decrease the coarser construction and to describe the resource capability of each server. The resources needed for mobile services ‘i’ for the time interval $T_i$ considers the basic resource used for efficient energy-power saving in the cloud mobile devices. In MFEP method, resource requirement of mobile services satisfies the resource allocation on the internet cloud zone.

![Figure 3.2 Multiple Grid Reduction Structure in Cloud](image)

Above figure 3.2 illustrate the structure of multiple grid reduction in the internet cloud zone. Hence, the grid reduced to a coarser construction decreases the energy consumption rate. Multi-grid approximation using MFEP method for ‘n’ resources $R_1, R_2, R_3…R_n$ calculates the mathematical as given below equation (3.1).

$$\text{Energy level} = \min \sum E (R_1, R_2, R_3… R_n)$$

\[ \ldots \ldots \text{Eqn (3.1)} \]

The structure of grid in cloud is minimized by producing the logical results and energy level of each resource is evaluated in proposed MFEP method. Resource management in MFEP method dynamically accepts the range with easy planning of the unique grid in the cloud infrastructure. MFEP method considers CPU to easily point
the multi dimension resources and decrease the energy for unnecessary computing load.

Using MFEPA method, the simple approach of terminal mainframe mobile communications is carried out with multiple grid reduction. Here, the values of i and j ranges from 1, 2, 3… n in MFEPA method and explained using equation (3.2).

\[
\text{Grid Reduction Form} = \min \sum \sum |M_{i,j} - M_{i,j}^*|
\]

\[
\ldots \ldots \text{Eqn (3.2)}
\]

Here in equation (3.2), \( M_{i,j}^* \) represents the mobile services of the current grid where \( M_i \) denotes the older matrix grid and the result is mapped with the \( M_{i,j}^* \) grid. The main objective of MFEPA method is attained using the multi-grid approximation that decreases the energy utilization in the cloud infrastructure. The design of multi-grid approximation technique is carried out easy way and it is explained as given below.

**Step 1:** Gather information continuously from cloud mobile services

Step 1.1: Includes mobile service workload, QoS and Resource energy consumption level

**Step 2:** calculate workload on each mobile service with \( T_i \) interval

Step 2.1: calculated workload located in the matrix grid

**Step 3:** Examine matrix grid and predict the grid to decrease the matrix (i.e.,) grid form from multiple grid structure

**Step 4:** Implements the cloud resources by satisfying mobile services and minimum energy level

**Algorithm 3.1 Multi-Grid Approximation Technique**
The first aim in MFEPA method used to reduce energy consumption by eliminate too much of mapping time in cloud internet zone. Multiple grid (i.e.,) machines is decreased to an unrefined construction, and the result is mapped back to the original grid. Mapping decrease the energy on the excessive computing load with efficient communication on mobile services. Finally, resource allocating solution is recognized which minimizes the entire energy using the multiple grid form.

3.3.2 Look-ahead control Mechanism

Next intention of MFEPA method is to decrease the power consumption level on the grid connections where the power of the system is handled using the look-ahead control. A correct decision on grid connection is attained by using look-ahead control mechanism along with weighted factor, thus decreasing the power consumption on the cloud zone. Mobile services various quickly with workload in the cloud zone, which easily cover the total user request with enhanced true positive ratio. The look-ahead control mechanism consists of weighting factor to manage the unwanted switching resources using the mobile services. Therefore, weighting factor monitors the workload variability as well as the power consumption rate.

The look-ahead control estimates the comparative significance of each users request and groups identical requests. Identical requests include the similar weighting factor as a result the waiting time for the resource utilizations is reduced. This minimization of waiting time decrease the power consumption and the weighting factors is evaluate periodically in the look-ahead control and adjusted depends on the user needs.
Weighting factor is measured in look-ahead control mechanism with the help of resource utilization level and number of request user count.

\[
\text{Weighted Factor} = \text{RUL weight} + \text{Request count}
\]

\[\ldots\ldots\text{Eqn (3.3)}\]

The weighting factor of ‘RUL’ represents the Resource utilization level on the IaaS point. The higher priority is given to higher weighting factor value, and this performs provided to utilize the mobile services. Finally, MFEPA method calculates an accurate overall performance rating and obtained first result to the higher priority users.

Look-ahead control based power model in MFEPA method is formularized as,

\[
\text{Weighting Factor } P(u) = K \times P_{\text{max}} + (1 - K) \times P_{\text{max}} \times u
\]

\[\ldots\ldots\text{Eqn (3.4)}\]

From equation (3.4), parameter Weighting factor \(P(u)\) represents the current power utilization with the weighting factor in MFEPA method. Here, \(P_{\text{max}}\) represents the maximum power consumed on the operation for the resource consumption, ‘u’ on the mobile services where ‘k’ is the portion of power consumed by server in cloud computing.

3.3.2.1 Combining Energy and Power Aware System in MFEPA method

The integration of energy and power simply classify the profit margin of the cloud users while using the Infrastructure as a Service (IaaS). The timing period of
MFEPA method describes the model with power and energy consumption rate with an integral factor variety between t1 and t2. It is formulated as given in below equation (3.5).

\[
\text{Timing period} = \int \text{Weighting Factor } P((u) t) * dt
\]

........... Eqn (3.5)

The weighting factor is describes as the time period with which the mobile services differ from the time position ‘t1’ to ‘t2’ and \(P((u)t)\) represents the unit of power consumed for particular mobile services on the cloud zone. The algorithmic step of the MFEPA method is explain as,

```
Input: Set of servers ‘S’, Time Interval T, Resources ‘R’
Output: Energy and power aware mobile services to cloud infrastructure

// Machine Flow based Energy-Power Approximation Algorithm
Step 1: Select the set of servers which perform the mobile services on cloud zone
Step 2: While true
Step 3: Select the resources for R1, R2, R3,…..Rn
Step 4: If satisfy the user request, then
Step 5: Computes min min \(\sum \sum |M_{i,j} – M'_{i,j}|\) with i and j values ranging between 1 and n for reducing the grid forms
Step 6: Multi-grid Approximation steps from Figure 4 calculate energy rate
Step 7: Optimize the solution based on the minimal energy rate
Step 8: End if
Step 9: Look-ahead control cover the entire user request with weighting factors
Step 10: Weighting Factor P (u) = K * P_{max} + (1 – K) * P_{max} * u provides the obtained
```

58
first result to the top priority users.

Step 11: Time period of the each cloud users defines the energy and power aware mobile service system

Step 12: End While

**Algorithm 3.2 Machine Flow based Energy-Power Approximation Algorithm**

Cloud server executes the resource allocation on mobile services by choosing the ‘S₁’, ‘S₂’, up to ‘Sₙ’. The server allows the service requests, and uses the multi-grid approximation technique to condense the grid points and achieve effective mapping in sensor network. The reduction of grid points compresses the energy consumption in cloud infrastructure. The look-ahead control with the weighting factor reduces the power consumption rate at special time interval.

### 3.4. EXPERIMENTAL EVALUATION OF MFEPA APPROACH

The MFEPA approach is implemented using JAVA language. The proposed MFEPA approach uses statlogheart dataset extracted from UCI for conducting the experiment. In Cloud environments the special toolkit is selected as a simulation platform. The user’s current needs for successful communication provisioned mixed Virtual Machine (VM) pack. Experimental device is simulated with cloud data center containing of 4 GB RAM of storage in consumer machines.

Every virtual machine runs a web-application with variable business based query request from client, which is modeled to produce the operation of bandwidth according
to consistently distributed random variable. It is compared with existing Energy-aware Resource Allocation Heuristics (ERAH) method developed by Anton Beloglazov Li et al. (2012) and MREE-PSO model developed by An-ping Xiong Li et al. (2014). The performance of the MFEPA approach is measured in terms of:

i) True Positive Rate
ii) Energy Consumption
iii) Grid Mapping Efficiency
iv) Profit Margin Rate
v) Grid Mapping Time

3.5. PERFORMANCE ANALYSIS OF MFEPA APPROACH

The performance of the MFEPA approach is compared with the two existing methods. The compared existing methods are namely, ERAH method developed by Anton Beloglazov Li et al. (2012) and MREE-PSO model developed by An-ping Xiong Li et al. (2014). To evaluate the MFEPA approach, the following metrics are used.

3.5.1. Performance analysis of True Positive Rate

The true positive rate is calculated according to the true positive and false negative conditions that describe the capacity to check the mobile services depending on the users require. The test result with the high positive rate provides higher sensitivity rate. The true positive rate is measured in terms of percentage (%).
True Positive Rate = \[(TPMS) / (TP + FN) * 100\]

\[\ldots \ldots \text{Eqn (3.6)}\]

From above equation (3.6), TPMS, TP, FN characterize the number of true positive mobile services and effects of true positive and false negative correspondingly. In order to reduce the energy consumption rate, a multi-grid approximation method is granted that assigns the resources based on the demand count indicated through matrix set.

**Table 3.1 Tabulation of True Positive Rate**

<table>
<thead>
<tr>
<th>Number of Cloud Users</th>
<th>True Positive Rate ( % )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Existing ERAH</td>
</tr>
<tr>
<td>10</td>
<td>59.32</td>
</tr>
<tr>
<td>20</td>
<td>61.14</td>
</tr>
<tr>
<td>30</td>
<td>61.98</td>
</tr>
<tr>
<td>40</td>
<td>63.47</td>
</tr>
<tr>
<td>50</td>
<td>66.45</td>
</tr>
<tr>
<td>60</td>
<td>67.11</td>
</tr>
<tr>
<td>70</td>
<td>68.97</td>
</tr>
<tr>
<td>80</td>
<td>70.54</td>
</tr>
<tr>
<td>90</td>
<td>73.47</td>
</tr>
<tr>
<td>100</td>
<td>73.55</td>
</tr>
</tbody>
</table>

Table 3.1 explains the true positive rate using MFEP Method, ERAH Algorithm and MREE-PSO model. For experimental evaluation, different number of cloud users is considered that ranges between 10 to 100 users. From the table value, it is descriptive
that the true positive rate using proposed MFEPAM Method is developed as compared to other existing methods.

Figure 3.4 shows the impact of true positive rate with respect to different number of cloud users. The proposed MFEPAM approach executes relatively well when compared with ERAH method developed by Anton Beloglazov Li et al. (2012) and MREE-PSO model developed by An-ping Xiong Li et al. (2014). With the application of weighting factor in proposed MFEPAM approach, different mobile services changes the workload conditions in the cloud zones. Thus, it improves the true positive rate with the help of total user request in cloud zones.

![Figure 3.3 Measure of True Positive Rate](image)

MFEPAM method easily covers the entire user request with improved true positive rate by 20% when compared with the ERAH method by Anton Beloglazov Li et al. (2012). The weighting factor monitors the workload variability based on the resource
utilization level, and amount of user request. Thus improves the true positive rate by 10% when compared with the MREE-PSO model developed by An-ping Xiong Li et al. (2014).

### 3.5.2. Performance analysis of Energy Consumption

The energy consumption in proposed MFEPA approach is given by the amount of energy provided in grid supplied from mobile service to cloud environment which is measured in terms of Joules (J). In MFEPA method, mapping shrinks the energy consumption during the avoidable computing load.

| Number of mobile services | Energy Consumption (J) |   |   |
|---------------------------|------------------------|---|---|---|
|                           | Existing ERAH          | Existing MREE-PSO | Proposed MFEPA |
| 20                        | 0.36                   | 0.32            | 0.26           |
| 40                        | 0.37                   | 0.33            | 0.27           |
| 60                        | 0.39                   | 0.36            | 0.31           |
| 80                        | 0.41                   | 0.37            | 0.33           |
| 100                       | 0.43                   | 0.38            | 0.33           |
| 120                       | 0.44                   | 0.41            | 0.34           |
| 140                       | 0.46                   | 0.43            | 0.37           |
| 160                       | 0.48                   | 0.45            | 0.41           |
| 180                       | 0.52                   | 0.48            | 0.43           |
| 200                       | 0.53                   | 0.49            | 0.45           |
Above table 3.2 explains the experimental values of energy consumption using proposed MFEP A Method with exiting ERAH method and MREE-PSO model. The energy consumption is measured with respect to number of mobile services ranges from 20 to 200. From the table value, it is descriptive that the energy consumption rate using proposed MFEP A Method is developed as compared to other existing methods.

The energy consumption using MFEP A method compared with ERAH method by Anton Beloglazov Li et al. (2012) and MREE-PSO model by An-ping Xiong Li et al. (2014) in figure 3.5 is presented for visual comparison based on the mobile services. The MFEP A reaches equivalent performance to ERAH method and MREE-PSO method. The energy consumption is reduced by 27% when compared with the ERAH method developed by Anton Beloglazov Li et al. (2012).

Figure 3.4 Measure of Energy Consumption
The terminal mainframe mobile communications is carried out in the MFEPA method with 16% lesser energy consumption when compared with the MREE-PSO method by An-ping Xiong Li et al. (2014).

### 3.5.3 Performance analysis of Grid Mapping Efficiency

The grid mapping efficiency is defined as the measure of energy usage and proves to be effective on the terminal mainframe mobile communications. The multi-grid approximation technique is used to condense the grid points and achieve effective mapping in cloud zone. It measures the grid mapping efficiency rate in terms of percentage (%).

**Table 3.3 Tabulation of Grid Mapping Efficiency**

<table>
<thead>
<tr>
<th>Grid Count</th>
<th>Grid Mapping Efficiency (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Existing ERAH</td>
</tr>
<tr>
<td>50</td>
<td>53</td>
</tr>
<tr>
<td>100</td>
<td>58</td>
</tr>
<tr>
<td>150</td>
<td>59</td>
</tr>
<tr>
<td>200</td>
<td>61</td>
</tr>
<tr>
<td>250</td>
<td>64</td>
</tr>
<tr>
<td>300</td>
<td>67</td>
</tr>
<tr>
<td>350</td>
<td>69</td>
</tr>
<tr>
<td>400</td>
<td>71</td>
</tr>
<tr>
<td>450</td>
<td>72</td>
</tr>
<tr>
<td>500</td>
<td>75</td>
</tr>
</tbody>
</table>
Table 3.3 shows the grid mapping efficiency rate using MFEPA Method, ERAH method and MREE-PSO model. Different grid count is considered in cloud zone for mapping the user request and the grid count is ranges from 50 to 500. From the table value, it is descriptive that the grid mapping efficiency rate using proposed MFEPA Method is developed as compared to other existing methods.

Figure 3.6 shows the difference of grid mapping efficiency of proposed MFEPA method greater than unlike number of grid count. All the results provided in figure 3.6, the proposed MFEPA approach considerably outperforms compared with ERAH method by Anton Beloglazov Li et al. (2012) and MREE-PSO model by An-ping Xiong Li et al. (2014). The better performance of MFEPA algorithm is reached due to the fact that it provides a capable method to provide multi-grid approximation for different grid count values.
Therefore MFEPA method executes the valuable result to decrease the grid structure. The reduction of grid structure develops the efficiency rate by 18% when compared with ERAH method by Anton Beloglazov Li et al. (2012). MFEPA method implements the range with the simple mapping of the unique grid in the cloud infrastructure by developing the mapping efficiency rate by 10% when compared with the MREE-PSO method by An-ping Xiong Li et al. (2014).

3.5.4. Performance analysis of Profit Margin Rate

The profit margin rate is estimated for various numbers of iterations that determine the commercial rate reached while using the IaaS information from the cloud zone which varies as per the user resources acquired from the cloud zone. The profit margin rate is defined in terms as the percentage (%).

Table 3.4 Tabulation of Profit Margin Rate

<table>
<thead>
<tr>
<th>Number of Iterations</th>
<th>Profit Margin Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Existing ERAH</td>
</tr>
<tr>
<td>1</td>
<td>62.38</td>
</tr>
<tr>
<td>2</td>
<td>64.25</td>
</tr>
<tr>
<td>3</td>
<td>66.13</td>
</tr>
<tr>
<td>4</td>
<td>66.87</td>
</tr>
<tr>
<td>5</td>
<td>68.14</td>
</tr>
<tr>
<td>6</td>
<td>69.76</td>
</tr>
<tr>
<td>7</td>
<td>71.24</td>
</tr>
<tr>
<td>8</td>
<td>72.3</td>
</tr>
<tr>
<td>9</td>
<td>73.48</td>
</tr>
<tr>
<td>10</td>
<td>74.37</td>
</tr>
</tbody>
</table>
The above table 3.4 illustrates the profit margin rate of the proposed MFEPA method using various iterations. The experiments were conducted using the different information from different users which is measured in terms of percentage (%). In the experimental setup, the number of iterations ranges from 1 to 10 users. The profit margin rate obtained by using proposed MFEPA method offer comparable values than the state-of-the-art methods.

The proposed MFEPA method extremely describes the profit margin rate as shown in figure 3.7 and it is produced depends on various iteration count.

![Figure 3.6 Measure of Profit Margin Rate](image)

**Figure 3.6 Measure of Profit Margin Rate**

From the figure, it is clarifying that the proposed MFEPA approach probably gives up important results compared with ERAH method by Anton Beloglazov Li et al.
(2012) and MREE-PSO model by An-ping Xiong Li et al. (2014). Then, the results are presented for various values of iterations considering the profit margin rate. The results reported above confirm that with the increase in the number of iteration being sent to the cloud zones by cloud users, the profit margin rate also increases. MFEPA method with the introduction of the weighting factor, describes the time period with which the mobile services changed from the time positions. Thus, profit margin rate gets improved by 14% when compared with ERAH method by Anton Beloglazov Li et al. (2012) and 8% improved when compared with the MREE-PSO method by An-ping Xiong Li et al. (2014).

3.5.5. Performance analysis of Grid Mapping Time

Grid mapping time is defined as the measure of amount of time used to carry out the operation in the mobile services. The grid mapping time is estimated using the variation among the starting time to map the grid and ending time after mapping the grid with the mapping time measured in terms of milliseconds (ms).

\[
\text{Processing (i.e., mapping) Time} = PT_1 - PT_2 \\
\text{......... Eqn (3.7)}
\]
Table 3.5 Tabulation of Grid Mapping Time

<table>
<thead>
<tr>
<th>Grid Size (KB)</th>
<th>Grid Mapping Time (ms)</th>
<th>Existing ERAH</th>
<th>Existing MREE-PSO</th>
<th>Proposed MFEPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>214</td>
<td>156</td>
<td>83</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>285</td>
<td>198</td>
<td>156</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>324</td>
<td>257</td>
<td>198</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>399</td>
<td>312</td>
<td>256</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>484</td>
<td>387</td>
<td>304</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>532</td>
<td>458</td>
<td>387</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>587</td>
<td>532</td>
<td>432</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>632</td>
<td>568</td>
<td>498</td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>687</td>
<td>621</td>
<td>534</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>734</td>
<td>687</td>
<td>588</td>
<td></td>
</tr>
</tbody>
</table>

The above table 3.5 illustrates the grid mapping time of the proposed MFEPA method using various grid sizes. The experiments were conducted using the different grid size which is measured in terms of percentage (%). In the experimental setup, the size of grid ranges from 10KB to 100KB. The grid mapping time obtained by using proposed MFEPA method offer comparable values than the state-of-the-art methods.

The proposed MFEPA method extremely describes the grid mapping time as shows in figure 3.8 and it is produced depends on size of grid. From the above table, the comparison is made with existing methods are namely, ERAH method developed by
Anton Beloglazov Li et al. (2012) and MREE-PSO model developed by An-ping Xiong Li et al. (2014).

![Figure 3.7 Measure of Grid Mapping Time](image)

The performance of grid mapping time is evaluated using a grid size of 10 KB to 100 KB. The significant results attained using MFEP algorithm is because the mapping time with look-ahead control estimates the relative importance of all users request and accordingly sets parallel requirements. Thus, the multiple grid (i.e.,) machines is reduced to a coarser construction, and the solution is mapped back to the original grid with minimal time. The mapping time is reduced by 57 % when compared with the ERAH method by Anton Beloglazov Li et al. (2012) and 28 % when compared with the MREE-PSO method by An-ping Xiong Li et al. (2014).
3.6 SUMMARY

Machine Flow based Energy-Power Approximation algorithm is presented for elastic cloud services. MFEPA algorithm uses multi-grid approximation technique, the method decreases the coarser arrangement and the solution is mapped back to the original grid. The mapping reduces the energy on the excessive computing load and works effectively on the terminal mainframe mobile services. Thus the proposed MFEPA algorithm attains comparable services reducing the energy on the unnecessary computing load and is very effective on energy-power saving. In MFEPA algorithm, Look-ahead control allocates weights to all the expected users and performs the operations to decrease the power usage on the wireless interface. MFEPA Algorithm decreases the memory utilization, so that the profit margin of the business providers with Infrastructure as a Service (IaaS) is also increased. Minimization of the power increases the performance on the elasticity cloud applications. Future, the dynamic prioritized load balance on cloud services is explained to reduce the memory utilization and solve the optimization difficulties in the elasticity cloud services.