CHAPTER 3

THREE DIMENSIONAL RESOURCE SCHEDULING ALGORITHM

3.1 INTRODUCTION

Khoo et al (2007) proposed a two dimensional algorithm by considering computational and Data requirements. This algorithm allocates job to resources (sites) once it matches computation and Data requirements. The issue here is sometimes higher capability resources (or resources with higher capability) are allocated to small jobs. In the proposed algorithm optimal resource allocation is done, at the time deadline is also considered as a major factor.

The three dimensional resource scheduling algorithm (3D algorithm) enables jobs with multiple resource requirements to run effectively on Grid Computing Environment. The three dimensions considered are Computation, Data and Deadline. These dimensions are the general classifications of resource requirements for a job. They are chosen to achieve faster computation through proper resource allocation. The various resource requirements of a job are CPU, hard disk space and bandwidth. Inter resource communication is also addressed by the concept of resource potential.

The available resources are aggregated and then combined into three major indices namely Computation, Data and Deadline based on the three dimensions. With these indices, a 3D plot is created and this plot
describes the virtual topology of a job’s resource requirements. Thus, it identifies the best allocation of resources for a job and this algorithm gives minimal waiting and execution time.

In the present chapter a model of the Grid environment is proposed along with certain assumptions to execute the 3D algorithm. Later sections describe the scheduling strategies adopted in this algorithm. Performance metrics are defined and simulation details are provided. Performance analysis of this algorithm with some commonly used algorithm is performed later in the chapter.

3.2 MODEL OF THE GRID ENVIRONMENT

The model of the Grid environment to execute the 3D algorithm is as follows,

- The grid environment consists of diverse machine types, disks/storage and networks.
- The resources in the grid environment are made of desktops, servers, clusters and multi-processor systems.
- All computation resources are connected through different bandwidths.
- Every resource is accessible to every other participating node in the grid.
- Changes in any shared resource at a site are known to all locations instantaneously throughout the grid environment.
- The importance of the resources with respect to each other is identical.
Figure 3.1 Network bandwidth connectivity

- Different amount of CPU, Hard disk space and Bandwidth are provided by the resources.

- A user from any node within the grid environment can submit a job. The execution results of the job are given back to the user.

- The grid scheduler determines how jobs should be scheduled and how the resources should be utilized.

- The CPU resource’s computational capability is represented in the form of MIPS (Million Instructions per second). This measure is used to standardize the performance of different CPU architectures in different sites.
Figure 3.1 illustrates the network bandwidth connectivity and this model is adopted for simulation and implementation. The bandwidth connectivity is essential in evaluating the Resource potential.

### 3.3 ASSUMPTIONS IN EXECUTION ENVIRONMENT

Following are the assumptions used in the execution environment, they are,

- Before execution, the resource requirements for a job can be divided.
- During execution, the resource requirements for a job do not change.
- The provided resource requirements for a job are the upper bound of resource usage.
- Every job has a data requirement in addition to computational requirement. In the data requirement, the data source and size is stated.
- The data source for a job can be located anywhere in the grid environment.
- A job can be scheduled anywhere in the grid environment for its execution.
- Once the distribution of resources start, the resources are locked for a job’s execution and reclaimed after use.
- The runtime of a job is calculated from the time the job is submitted till the end of its execution.
3.4 SCHEDULING STRATEGY

The three dimensional scheduling algorithm finds the best matching sites for a job to be submitted. This algorithm considers both job requirements and resource capabilities. The strategy adopted is to have an aggregation algorithm, which results in dimensionless indices, as the representation units for both user request and site resource are the same.

The resources are allocated such that the job’s requirements are satisfied in a single site to improve performance. Efficiency in execution is achieved by allocating a set of best fit resources.

3.5 SCHEDULING DIMENSIONS

The three dimensions used in this algorithm are,

- Computation
- Data
- Deadline

These dimensions are selected to achieve faster computation and higher throughput. Faster computation is achieved by proper resource allocation such as MIPS and disk space. The aggregation of available resources is combined into the three indices such as Computation, Data and Deadline based on the three dimensions.

A three dimensional (3D) plot is constructed with the above three indices. The plot describes the virtual topology of the job’s resource requirements. The virtual map is computed every time a job is submitted to the grid environment and it is different for each job. The most suited resource provider for a job will be the sites located nearest to the origin in the map.
3.6 COMPUTATION DIMENSION

The entities considered as resources in the computation dimension are CPU in MIPS (C) and Disk space (F) in GB (Giga Bytes). These resources are needed for the effective computation of a job. Each resource is represented by a capability value and a requirement value. Communication between resources in various sites is addressed by the concept of resource potential. This parameter is also included in the calculation of computation index.

3.6.1 Resource Potential

The resource potential $P_i$ of a resource $R_i$ defines the network connectivity between itself and its neighboring sites. The potential value is computed as,

$$P_i = \sum B_{ij}$$

(3.1)

where, $B_{ij}$ is the bandwidth between two sites $i$ and $j$ and is calculated as min

$$\{B_{ij}^{\text{download}}, B_{ji}^{\text{upload}}\}.$$

(3.2)

For ease of implementation, the average of uplink and downlink is used in the implementation model as shown in Figure 3.1.

The resource environment is characterized as set of resources denoted by,

$$S = \{R_1, R_2, \ldots, R_n\}.$$

(3.3)

The allocable computational resources in a site $i$ are represented as a set,

$$S_c = \{R_i, t\} \text{ where } S_c \subseteq S.$$

(3.4)
R_i is represented by a 3-tuple of f_i ( <C,F,P_i>,t ) denoting the four resources in the allocation strategy. These three values C, F and P_i change dynamically with resource availability over time t. Similarly the job environment is denoted by a set J={A_1,…….,A_j} and the computational requirement of each job A_j is given by g_j ( <C,F,P_{src}>,t ).

3.6.2 Computation index calculation

The resource requirements of jobs and resource availability at sites are aggregated and evaluated, further inter-resource relationship is calculated. A single computation index is arrived by computing a ratio of provision for site i and job j, i.e., the ratio between the resource requested and the amount of resource to be provided. The three dimensional scheduling algorithm allocates the site that best fits a job i.e. the difference between the amount of resources required for a job and the amount of resources provided at a site should be minimum for the best fit.

3.6.2.1 Provision for CPU

For computational resources, the formula for provisioning for CPU is given by,

\[ R_{ij}(C) = 1 - \frac{f_i(C)}{g_j(C)} \]  \hspace{1cm} (3.5)

where \( f_i(C) \) is MIPS resource provided at site i

\( g_j(C) \) is MIPS resource required by job j

Only positive values are considered for drawing the virtual map and so the values for \( R_{ij} \) are truncated to 0 if the ratio value evaluates to be less than 0.
The general observations made in the calculation of ratio of provision is as follows,

- If the value is 0, it implies perfect ability to provision for a resource.
- If the value is >0, Inability to provide for a resource.
- If the value is <0, Over ability to provision resources, set the value as 0.

**3.6.2.2 Provision for Hard Disk**

Ratio of provision for Hard Disk is given by,

\[
R_{ij}(F) = 1 - \left( \frac{f_i(F)}{g_j(F)} \right)
\]  

(3.6)

where \( f_i(F) \) is Hard Disk capability provided at site i
\( g_j(F) \) is Hard Disk capability required by job j

Only positive values are considered for drawing the virtual map and so the values for \( R_{ij} \) are truncated to 0 if the ratio value evaluates to be less than 0. The general observation in the calculation of the ratio of provision for hard disk is the same as that made for CPU provision.

**3.6.2.3 Site Potential**

Site Potential is calculated as

\[
R_{ij}(P) = 1 - \left( \frac{P_i}{P_{src}} \right)
\]  

(3.7)

where \( P_i \) is the potential value of the site i
\( P_{src} \) is the source file potential
The ratio between the potential value of the site $P_i$ and the source file potential $P_{src}$ is calculated. This evaluates whether the site connectivity is equal or better to where the source data file is located. The general observation made earlier in the calculation of the ratio of provision also holds well here.

To merge all the provisioning ratios to a single dimensionless Computation index, Euclidean distance between the provisioning ratios is calculated.

In general,

$$\text{Euclidean distance} = \sqrt{(x2-x1)^2+(y2-y1)^2+(z2-z1)^2} \quad (3.8)$$

In the construction of virtual map for each job, the Euclidean distance is computed between the origin and each of the provisioning ratios. These ratios are then aggregated to a dimensionless computational index ($x_{ij}$) as,

$$x_{ij} = \sqrt{\left( R_{ij} \{C\} - 0 \right)^2 + \left( R_{ij} \{F\} - 0 \right)^2 + \left( R_{ij} \{P\} - 0 \right)^2}$$

Hence the computation index evaluates to,

$$x_{ij} = \sqrt{\left( R_{ij} \{C\} \right)^2 + \left( R_{ij} \{F\} \right)^2 + \left( R_{ij} \{P\} \right)^2} \quad (3.9)$$

### 3.7 DATA DIMENSION AND INDEXING

In Data dimension, resources that affect the I/O of a job are interrelated. An index is evaluated that helps in determining a good resource site that would best execute a job. This is because a site capable of executing a job locally would incur minimal I/O time as compared to any other remote location. The time for I/O is found based on estimated data communications
required and the bandwidth between the source file location and the target job allocation site. The I/O time depends on the availability of bandwidth at a site. The available bandwidth also changes with time if a site is sharing any of its resources with other sites in the grid environment. In the worst case, the amount of data required for a job would be the amount of hard disk resource required at the site to store the data to be processed.

The data index is computed as the ratio between the I/O communication time to the estimated job runtime. Hence, the job is allocated to the target resource, which has the lowest ratio.

The data capabilities of the resource is denoted by $S_d\{R_i,t\}$. Each item in the set is represented by $d_i\{<B>,t\}$. The data requirement of a job $j$ is denoted by $e_j\{<F,A^{\text{runtime}},t\}$ where $A^{\text{runtime}}$ is the job’s estimated runtime. Thus, data index is evaluated as an example for inter-resource relation.

The ratio is written as,

$$y_{ij} = \left( \frac{e_j\{F\}}{d_i\{B_{ij}\}} \right) \cdot \left( \frac{1}{A^{\text{runtime}}} \right)$$

(3.10)

A smaller value for the ratio represents better site preference for a job.

### 3.8 DEADLINE

Deadline is the framework for regulating the supply and demand for resources. It allocates the resources to a job based on the users’ requirements. Calculation of deadline index optimizes time factor thereby helps to achieve a lower job completion time and better resource utilization.

Calculation of Deadline is the third index which is considered in drawing the virtual map. Similar to the other two indices namely Computation and Data, Deadline is also computed at each site in order to find the best resource at a site which efficiently executes a job within the user specified deadline ($U_d$).
The runtime of each job ($A_{\text{runtime}}$) is estimated by,

$$A_{\text{runtime}} = \frac{\text{job length}}{\text{resource MIPS}}.$$  \hfill (3.11)

The resource in the site which has minimum difference between the user specified deadline and the estimated runtime for a job is chosen as the best fit for that job. Thus, the deadline $D_{ij}$ for job $j$ in a site $i$ is computed as follows

$$D_{ij} = (1 - (A_{\text{runtime}} / U_d))$$  \hfill (3.12)

If the Computation index of a site is $>0$ (i.e., if that site has inability to provide for a resource) a constant $K$ is added to the above value in order to increase the Euclidean distance of that site from the origin, because the site which does not have the ability will never be chosen for the job as it is farther from the origin in the virtual map. Thus, deadline is calculated by the formula,

$$D_{ij} = K + (1 - (A_{\text{runtime}} / U_d))$$  \hfill (3.13)

### 3.9 VIRTUAL MAP

From the individual Computation, Data and Deadline indices described, the best allocated resources are the ones with low index values. These indices are used to construct a 3D plot which is termed as the Virtual map. The sites which position themselves closest to the origin are the best suited resources for a job. The Euclidean distance from the origin denotes the best resources that match the job’s resource requirements for a time instance.

A Virtual map is drawn for each job because the job requirements differ for each submitted job.
3.10 PERFORMANCE METRICS

The metrics like Average waiting time and Queue Completion time are used to compare the three dimensional algorithm with other commonly used algorithms.

3.10.1 Average wait time

Average waiting time (AWT) is the average amount of time a job waits in the queue before being executed. This starts from the point of submission to the point when the job begins its execution,

\[
AWT = \frac{\sum (e_j - s_j)}{J}
\]

(3.14)

for all J jobs ,where, \(e_j\) is the time the job begins execution and \(s_j\) is the job’s submission time. This metric is a measure of responsiveness of the scheduling mechanism.

3.10.2 Queue Completion Time

Queue Completion time (QCT) is the amount of time it takes for the scheduling algorithm to be able to process all the jobs in the queue. This is calculated by tracking the time when the first job enters the scheduler until the time the last job exits the scheduler.

\[
QCT = e_{J-1} + E_{J-1} - s_0
\]

(3.15)

where \(e_{J-1}\) is the time the job begins execution , \(E_{J-1}\) is the execution time of the last job and \(s_0\) is the first job’s submission time.
Figure 3.2 depicts the job allocation strategy

Job announces itself to grid environment with its requirements

For all resources, calculate

Computation index, Data index, Deadline Index

Construct a virtual map for the job

Select the resource which is closest to the origin

Submit the job to the selected resource

Figure 3.2 Job Allocation
Figure 3.3 depicts the process of calculating the Computation Index.

1. Calculate the ratio of provision for CPU.
2. Calculate the ratio of provision for hard disk space.
3. Calculate the ratio of provision for resource potential.
4. Check the ratio value.
5. If perfect fit, go to step 6. If over ability, set ratio=0. If inability, go to step 7.
6. Calculate computation index.

Figure 3.3 Computation index
Figure 3.4 depicts the process of calculating the Data Index

1. Determine the source file location
2. Retrieve the size of source file
3. Get the bandwidth between source file location and target job allocation site
4. Find ratio between retrieved file size and bandwidth
5. Estimate the runtime of the job
6. Determine the data index
Figure 3.5 depicts the process of calculating the Deadline Index.

1. Calculate the difference between the user deadline and estimated run time.
2. Calculate the ratio between the above difference and the user deadline.
3. Is computation index > 0?
   - Yes: Add constant, K to the calculated ratio.
   - No: Not a suitable site for job execution.
4. Is 0 < index < 1?
   - No: Not a suitable site for job execution.
   - Yes: Suitable site for job execution.
3.12 THREE DIMENSIONAL SCHEDULING ALGORITHM

3.12.1 Job allocation

Following are the steps involved in submission of job using 3D Algorithm

1. Job announces itself to grid environment with its requirements.

2. For each site, calculate,
   a. Computation Index
   b. Data Index
   c. Deadline Index

3. Construct a Virtual map for the job using the indices calculated in Step2.

4. Select the site which is closest to the origin in the Virtual map.

5. Submit the job to the available resource in the site.

3.12.2 Computation index

Computational index can be calculated as follows,

1. Calculate the ratio of provision between the available CPU resource and required CPU resource for the execution of the job.

2. Calculate the ratio of provision between the available hard disk space of the resource and the required hard disk space for the job.

3. Calculate the ratio of provision between the potential value of the resource and the source file potential.
4. For each calculated ratio of provision,
   a. If the value is 0, perfect ability to provision for a resource.
   b. If the value is >0, Inability to provision for a resource.
   c. If the value is <0
      i. Over ability to provision for a resource.
      ii. Set the value as 0.

5. Calculate the Computation Index by aggregating the above calculated ratios.

3.12.3 Data index

Data index can be calculated as follows,

i) Determine the source file location.

ii) Retrieve the size of the Source file.

iii) Get the Bandwidth between the source file location and the target job allocation site.

iv) Calculate the ratio between the File size and Bandwidth.

v) Estimate the runtime of the job.

vi) Calculate the ratio between the values calculated in Step 4 & 5 to obtain the data index.
3.12.4 Deadline index

Following are the steps involved in the calculation of Deadline index.

1. Estimate the runtime of the job.
2. Calculate the ratio between the estimated runtime of the job and User deadline.
3. The deadline is computed in such a way that the site which has the minimum difference between the user specified deadline and estimated runtime is chosen for the job to execute.
4. If computation index is greater than 0 (if it is an inability site), then a constant K is added to the above computed value to increase the Euclidean distance of that site from the origin in the virtual map.
5. If the ratio for deadline
   a. lies between 0 and 1, the resource is suitable for job allocation.
   b. Is greater than 1, the resource is not suitable for job allocation.

3.13 SIMULATION AND SYSTEM IMPLEMENTATION

In the simulation, five sites of different configurations are considered. Each site is connected to each other through different bandwidths (Mbps). The capacity for computation in a CPU resource is provided in the form of MIPS. The capacity for storage in each site is expressed in GB.
Table 3.1 Site Configuration

<table>
<thead>
<tr>
<th>Site</th>
<th>Site configuration</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CPU (MIPS)</td>
<td>HDD (GB)</td>
</tr>
<tr>
<td>A</td>
<td>1885</td>
<td>100</td>
</tr>
<tr>
<td>B</td>
<td>4270</td>
<td>200</td>
</tr>
<tr>
<td>C</td>
<td>7155</td>
<td>300</td>
</tr>
<tr>
<td>D</td>
<td>10540</td>
<td>400</td>
</tr>
<tr>
<td>E</td>
<td>14425</td>
<td>500</td>
</tr>
</tbody>
</table>

* Potential is calculated as \( \sum B_{ij} \) (as given in Figure 3.1) where \( i \) is the site and \( j \) is the job.

3.13.1 Job 1 specification

10000 MI, 400GB HDD, 6 sec Runtime, Required file-TestFile1

3.13.1.1 Calculation of Computation Index

i) The Ratio of provision for CPU is given as,

\[ R_{ij}\{C\} = 1 - \left( \frac{f_i\{C\}}{g_j\{C\}} \right) \]

and \( R_{ij}\{C\} = 0 \) if \( R_{ij}\{C\} < 0 \)

<table>
<thead>
<tr>
<th>Sites</th>
<th>( R_{ij}{C} )</th>
<th>Provision</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>( 1 - (1885/10000) )</td>
<td>0.8115</td>
</tr>
<tr>
<td>B</td>
<td>( 1 - (4270/10000) )</td>
<td>0.5730</td>
</tr>
<tr>
<td>C</td>
<td>( 1 - (7155/10000) )</td>
<td>0.2845</td>
</tr>
<tr>
<td>D</td>
<td>( 1 - (10540/10000) )</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>( 1 - (14425/10000) )</td>
<td>0</td>
</tr>
</tbody>
</table>

ii) The Ratio of provision for Hard disk is calculated as,

\[ R_{ij}\{F\} = 1 - \left( \frac{f_i\{F\}}{g_j\{F\}} \right) \]

and \( R_{ij}\{F\} = 0 \) if \( R_{ij}\{F\} < 0 \)
### iii) Site Potential is calculated as,

$$R_{ij}^P = 1 - \left( \frac{P_i}{P_{src}} \right)$$

and $$R_{ij}^P = 0$$ if $$R_{ij}^P < 0$$

<table>
<thead>
<tr>
<th>Sites</th>
<th>$$R_{ij}^F$$</th>
<th>Provision</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1-(100/400)</td>
<td>= 0.75</td>
</tr>
<tr>
<td>B</td>
<td>1-(200/400)</td>
<td>= 0.5</td>
</tr>
<tr>
<td>C</td>
<td>1-(300/400)</td>
<td>= 0.25</td>
</tr>
<tr>
<td>D</td>
<td>1-(400/400)</td>
<td>= 0</td>
</tr>
<tr>
<td>E</td>
<td>1-(500/400)</td>
<td>= 0</td>
</tr>
</tbody>
</table>

### iv) Finally Computation index ($x_{ij}$) is calculated as,

$$x_{ij} = \sqrt{(R_{ij}^C)^2 + (R_{ij}^F)^2 + (R_{ij}^P)^2}$$

<table>
<thead>
<tr>
<th>Site</th>
<th>Computation index</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>sqrt[ 0.8115^2+0.75^2+0] = 1.105</td>
</tr>
<tr>
<td>B</td>
<td>sqrt[ 0.573^2+0.5^2 +0] = 0.76048</td>
</tr>
<tr>
<td>C</td>
<td>sqrt[ 0.2845^2+0.25^2+0] = 0.37874</td>
</tr>
<tr>
<td>D</td>
<td>sqrt[ 0+0+0] = 0</td>
</tr>
<tr>
<td>E</td>
<td>sqrt[ 0+0+0] = 0</td>
</tr>
</tbody>
</table>
3.13.1.2 Calculation of Data index

Data index is calculated as \( y_{ij} = (F/B_{ij}) \times (1/A) \)

Source file size - 2430 MB

Source file location is assumed at site E

<table>
<thead>
<tr>
<th>Site</th>
<th>Data index</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>((2430<em>1885) / (1</em>10^6 * 10000)) = 0.00046</td>
</tr>
<tr>
<td>B</td>
<td>((2430<em>4270) / (1</em>10^6 * 10000)) = 0.00104</td>
</tr>
<tr>
<td>C</td>
<td>((2430<em>7155) / (4</em>10^6 * 10000)) = 0.00043</td>
</tr>
<tr>
<td>D</td>
<td>((2430<em>10540) / (100</em>10^6 * 10000)) = 0.00003</td>
</tr>
<tr>
<td>E</td>
<td>Same Location            = 0</td>
</tr>
</tbody>
</table>

3.13.1.3 Calculation of Deadline index

Deadline index is calculated as,

\[ \text{Deadline Index} = K + \left[ 1 - \frac{A^{\text{runtime}}}{\text{User deadline}} \right] \]

<table>
<thead>
<tr>
<th>Site</th>
<th>Deadline Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1+1-[(10000/1885)/4] = 2</td>
</tr>
<tr>
<td>B</td>
<td>1+1-[(10000/4270)/4] = 1.41452</td>
</tr>
<tr>
<td>C</td>
<td>1+1-[(10000/7155)/4] = 1.65059</td>
</tr>
<tr>
<td>D</td>
<td>1-[(10000/10540)/4] = 0.76281</td>
</tr>
<tr>
<td>E</td>
<td>1-[(10000/14425)/4] = 0.82669</td>
</tr>
</tbody>
</table>

The distance between the job and the sites is calculated by using Euclidean distance as shown below and in Table 3.2.
After the calculation of Computation Index, Data index, Deadline index and further applying these values to find Euclidean distance it can be inferred from the Table 3.2 that the suitable site is Site D. This Site has distance of 0.76281 which is the least and thus becomes the most appropriate site for job allocation after applying all the indices.

Once the job is submitted the Virtual 3D map is constructed. Figure 3.6 indicates the virtual map for Job1 created once the computations are done. Figure 3.6 clearly indicates that the site D is near to the origin and thus it is the site for job allocation.
On comparing 3D allocation to 2D allocation, Figure 3.7 shows that Site E is near to the origin in 2D strategy and in 3D, Site D is near to the origin. This shows some effect on the overall performance of the Grid environment, which is later discussed in section 3.14.
Figure 3.7  2D Virtual map for job 1

3.13.2  Job 2 Specification

13000 MI, 450 GB HDD, 5 sec Runtime, Required file-TestFile2

Table 3.3 Indices for job 2

<table>
<thead>
<tr>
<th>Site</th>
<th>Computation Index</th>
<th>Data Index</th>
<th>Deadline</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1.15584</td>
<td>0.00004</td>
<td>2</td>
<td>2.30997</td>
</tr>
<tr>
<td>B</td>
<td>0.87155</td>
<td>0.00009</td>
<td>1.3911</td>
<td>1.64157</td>
</tr>
<tr>
<td>C</td>
<td>0.55970</td>
<td>0</td>
<td>1.63662</td>
<td>1.72968</td>
</tr>
<tr>
<td>D</td>
<td>1.41421</td>
<td>0</td>
<td>2</td>
<td>2.44949</td>
</tr>
<tr>
<td>E</td>
<td>0</td>
<td>0.00007</td>
<td>0.81976</td>
<td>0.81976</td>
</tr>
</tbody>
</table>
Using the Three Dimension algorithm, after the calculation of Computation index, Data index and Deadline index and further applying the values to the Euclidean distance equation as performed in job 1, it is found that the site E is the appropriate site for processing with a minimum distance value as shown in Table 3.3.

Figure 3.8 shows the virtual map for Job 2 using 3D and the map indicates that site E is the appropriate site for job 2.

![3D Virtual Map for Job 2](image)

**Figure 3.8** 3D Virtual map for job 2

In 2D allocation strategy for the same specification of jobs it is seen that Site C is being allocated as compared to site E by 3D allocation strategy. Figure 3.9 indicated 2D allocation scheme for job 2.
3.13.3 Job 3 Specification

4000 MI, 120 GB HDD, 4 sec Runtime, Required File- TestFile3

Table 3.4 Indices for job 3

<table>
<thead>
<tr>
<th>Site</th>
<th>Computation Index</th>
<th>Data Index</th>
<th>Deadline</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.5544</td>
<td>0.00038</td>
<td>1.29266</td>
<td>1.40653</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>0.00086</td>
<td>0.68774</td>
<td>0.68774</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0.00036</td>
<td>0.81365</td>
<td>0.81365</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>0.8735</td>
<td>0.87350</td>
</tr>
<tr>
<td>E</td>
<td>1.1577</td>
<td>0</td>
<td>2</td>
<td>2.31090</td>
</tr>
</tbody>
</table>
Using the Three Dimension algorithm, after the calculation of Computation index, Data index and Deadline index and further applying the values to the Euclidean distance equation as performed in job1 and job2 it is found that the site B is the appropriate site for processing with a minimum distance value as shown in Table 3.4.

Figure 3.10 shows the virtual map for Job 2 using 3D and the map indicates that site E is the appropriate site for job 2 to be processed.

![Figure 3.10 3D Virtual map for Job3](image)

In 2D allocation strategy for the same Job3 specification, it is seen that Site D is being allocated as compared to site B by 3D allocation strategy. Figure 3.9 indicated 2D allocation scheme for job 3.
Performance analysis of 3D algorithm against 2D strategy and Backfill algorithm is performed. Table 3.5 compares these three algorithms with respect to two metrics namely AWT and QCT

Table 3.5 Comparison of Backfill, 2D, 3D

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>AWT (time units)</th>
<th>QCT (time units)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKFILL</td>
<td>22</td>
<td>79.05</td>
</tr>
<tr>
<td>2D</td>
<td>13.67</td>
<td>43.55</td>
</tr>
<tr>
<td>3D</td>
<td>10</td>
<td>31.85</td>
</tr>
</tbody>
</table>

Figure 3.11 2D Virtual map for Job3
Table 3.6 Percentage improvement over Backfill

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>AWT (%)</th>
<th>QCT (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D</td>
<td>37.86</td>
<td>44.91</td>
</tr>
<tr>
<td>3D</td>
<td>54.55</td>
<td>59.71</td>
</tr>
</tbody>
</table>

Table 3.7 Percentage improvement over 2D

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>AWT (%)</th>
<th>QCT (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKFILL</td>
<td>(60.9)*</td>
<td>(81.51)*</td>
</tr>
<tr>
<td>3D</td>
<td>26.85</td>
<td>26.87</td>
</tr>
</tbody>
</table>

* Backfill under performs 2D by 60.9% in AWT and 81.51% in QCT.

From Table 3.6 it is seen that the 3D scheduling algorithm outperforms Backfill by 54.55% in AWT and 59.71% in QCT. Comparing with 2D scheduling algorithm as shown in Table 3.7, it is seen that 3D scheduling algorithm outperforms 2D strategy by 26.85% in AWT and 26.87% in QCT.

These two metrics are normalized to the Backfill algorithm to show the performance differences. Table3.8 and Figure3.12 clearly indicate that in terms of performance measures using Average waiting time and Queue completion time, 2D algorithm outperforms Backfill algorithm, but 3D outperforms both Backfill and 2D algorithms.
Table 3.8 Normalized performance against Backfill

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>AWT</th>
<th>QCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKFILL</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2D</td>
<td>0.62</td>
<td>0.55</td>
</tr>
<tr>
<td>3D</td>
<td>0.45</td>
<td>0.40</td>
</tr>
</tbody>
</table>

Figure 3.12 Normalized Performance against Backfill algorithm
3.15 CONCLUSION

Following are the salient features of the proposed 3D scheduling algorithm.

- Each dimensional index is independent between the sites.
- Each index is calculated locally at the participating sites within the grid environment.
- The computation of the three indices performs clear stages of resource selection and scheduling.
- The ability of resource provision is computed in the sites itself. Hence, the willingness to accept a job is provided by each site.
- A job can be submitted from anywhere in the grid environment.
- The sites are allowed to get the specific requirements for each job and the indices are evaluated accordingly.
- The main scheduler in the environment does not have any complex algorithms. It sorts only the resulting Euclidean measure got from the environment.
- The Euclidean measure is computed sequentially and the resources are pre-emptively deducted when multiple jobs arrive in a site.
- The complexity of computing the indices is linear to the number of dimensions considered in the virtual map and to the number of resources. Hence, the number of resources can be increased without affecting the algorithm.

In the next chapter (chapter 4) “Economy based scheduling for utilization of space shared resources in a Bag of Tasks” is proposed.