CHAPTER – 2

APPLICATIONS OF IDLE/WAITING TIME $O_{i,w}$ OPERATOR IN THE SCHEDULING MODELS

Sequencing essentially alludes to the determination of order of jobs over time by which the jobs are to be carried-out by different machines. The scheduling problems are regular event in day by day living e.g. sequence of jobs for processing in an assembling lodge, projects meant to be running in an arrangement at a PC focus and so forth. Fundamental work in the ground of sequencing was proposed by Johnson [45] who builds the method for the reduction of elapsed time in two/multi stage flow shop scheduling. Convey [22] so et.al. plan the programmed integer number model for sequencing, Scharge.L. and Ignall, E. [43] implemented Branch and Bound technique in scheduling problems. Additionally Mitten [64] examined the n jobs, 2 machines flowshop. Das and Maggu [61] presented the concept of equivalent job-block in field of sequencing which has numerous applications in the generation concern, healing centre administration and so on where need of one job over different jobs. It might emerge the extra cost for giving this facility. Bagga [6], Das and Maggu [59], Szwarch [87], Hitomi and Yoshida [92], Anup [4] and so on inferred the optimal methods for two/three or multistage flow shop problems considering the different limitations and criteria. Kern and Nawjin [49], Riezebos and Goalman [75] proceeds with managing distinctive scheduling problems including time lags. Singh, T.P., Gupta, D. [80]. Associates probabilities with processing time and set-up time, time of transportation and also concept of breakdown segments in their studies. Later, Singh, T.P., Gupta, D. [35], [37] considered two/multi stage flowshop problem to minimize rental cost under a pre-defined rental strategy in which the probabilities have been related with processing time on every machine. Das and Maggu [60] presented the concept equivalent job-block in the field of sequencing.

This chapter is divided into two sections:
2.1 nx2 flow shop scheduling problem including transportation time and idle/waiting time operator $o_{i,w}$ for a job-block.

2.2 Two stage flow shop scheduling problem including idle/waiting time operator $o_{i,w}$ for a job block where jobs are performed in the string of disjoint job blocks.

2.1 nX2 FLOW SHOP SCHEDULING PROBLEM INCLUDING TRANSPORTATION TIME AND IDLE/WAITING TIME OPERATOR $O_{i,w}$ FOR A JOB-BLOCK

The fundamental investigation of flowshop was been searched by Johnson, S.M. [46] added work was produced by MacNaughton, E. [57], Das, G. and Maggu, P.L. [60], [61] Schrage, L. and Ignall, E. [43], Yoshida and Hitomi [92], Smith, R.A. and Dudek, R.A. [85], Singh, T.P. [80] and Gupta, D. [83], and so on by taking into account different system parameters. The fundamental concept of equivalent job was presented by Das, G. and Maggu, P.L. [61]. Heydari [42] managed a flow shop scheduling problem in which n jobs are to be handled in twice different job blocks as the form of string comprises of first job blocking which remains unaltered whereas other is elective. Lomnicki, Z.A. [55] associated the branch and bound methodology to the flow shop arranging problems. Further number of scheduling models were delivered by Schrage, L. and Ignall, E. [43], Chandrasekharan, R. [18], Lomnicki, Z.A. and Brown, A.P.G. [13], with the branch and bound strategy to the machine shop problem by displaying unmistakable framework. In realistic circumstances times of processing are not for the most part deterministic in nature so in the present study we have related probabilities with their sequencing times to all the jobs. Gupta, D. [31], [32], [33] studied the branch and bound technique for three stage flow shop sequencing problem incorporating transportation time associated with their separate probabilities. In this part, we have built up an algorithm in which processing times are associated with probabilities and further branch and bound technique is utilized to find the ideal sequence of jobs. Subsequently the problem talked about here is more extensive and has noteworthy utilization of theoretical results in procedure commercial ventures.
This section is based on paper “Branch and Bound Approach in Two Stage Flow shop Scheduling Problem including Transportation Time and idle/waiting time Operator $O_{i,w}$ for a Job-Block”, published in International journal of computer application, Vol. 2 (2), 2012.

2.1.1 Definition
Let $R_+$ be the set of non negative numbers. Let $G = R_+ \times R_+$. Then $O_{i,w}$ is defined as a mapping from $G \times G \rightarrow G$ given by:

$$O_{i,w}[(x_1, y_1), (x_2, y_2)] = (x_1, y_1) O_{i,w} (x_2, y_2)$$

$$= \{ x_1 + \max ((x_2, y_1), 0, y_2 + \max (y_1, x_2, 0)) \}$$

where $x_1, x_2, y_1, y_2 \in R_+$

2.1.2 Statement of Theorem
Let $n$ jobs $1, 2, 3, \ldots \ldots n$ are processed through two machines A & B in order AB with processing time $a_i$ & $b_i$ ($I = 1, 2, 3, \ldots \ldots n$) on machine A and B respectively.

If $(a_p, b_p) O_{i,w} (a_q, b_q) = (a_\beta, b_\beta)$

then $a_\beta = a_p + \max (a_q - b_p, 0)$ and

$$b_\beta = b_q + \max (b_q - a_q, 0)$$

where $\beta$ is the equivalent job for job block $(p, q)$ and $p, q \in \{1, 2, 3, \ldots \ldots n\}$.

Proof:
Starting by the equivalent job block criteria theorem for $\beta = (p, q)$ given by Maggu & Das (5), we have:

$$a_\beta = a_p + a_q - \min (b_p, a_q) \quad \ldots (1)$$

$$b_\beta = b_p + b_q - \min (b_p, a_q) \quad \ldots (2)$$

Now, we prove the above said theorem by a simple logic:

CASE I: When $a_q > b_p$

$$a_q > b_p > 0$$
\[ \max \{ a_q > b_p, 0 \} = a_q > b_p \quad \ldots(3) \]

and \( b_p > a_q < 0 \)

\[ \max \{ b_p > a_q, 0 \} = 0 \quad \ldots(4) \]

\begin{align*}
(1) \quad a_\beta &= a_p + a_q - \min (b_p, a_q) \\
&= a_p + a_q - b_p \quad \text{as } a_q > b_p \\
&= a_p + \max \{ a_q - b_p, 0 \} \quad \text{using (3)}
\end{align*}

\begin{align*}
(2) \quad b_\beta &= b_p + b_q - \min (b_p, a_q) \\
&= b_p + b_q - b_p \quad \text{as } a_q > b_p \\
&= b_q + (b_p - b_p) \\
&= b_q + 0 \\
&= b_q + \max (b_p - a_q, 0) \quad \text{using (4)}
\end{align*}

\textbf{CASE II: When } \quad a_q < b_p 

\begin{align*}
a_q - b_p < 0 \\
\max (a_q - a_q, 0) &= b_p - a_q \\
\text{and} \quad b_p - a_q > 0 \\
\max (b_p - a_q, 0) &= b_p - a_q
\end{align*}

\begin{align*}
(1) \quad a_\beta &= a_p + a_q - \min (b_p, a_q) \\
&= a_p + a_q - a_q \quad \text{as } a_q > b_p \\
&= a_p + a_q - a_q \quad \text{as } a_q > b_p \\
&= a_p + 0 \\
&= a_p + \max (a_q - b_p, 0) \quad \text{using (7)} \quad \ldots(9)
\end{align*}

\begin{align*}
(2) \quad b_\beta &= b_p + b_q - \min (b_p, a_q)
\end{align*}
\[ b_p + b_q - a_q \quad \text{as} \quad a_q < b_p \]
\[ = b_p + (b_p - a_q) \]
\[ = b_p + \max(b_p - a_q, 0) \quad \text{using (8)} \quad \text{(10)} \]

**CASE III:** When \( a_q = b_p \)

\[ a_q - b_p = 0 \]
\[ \max(a_q - b_p, 0) = 0 \quad \text{(11)} \]

Also

\[ b_p - a_q = 0 \]
\[ \max(b_p - a_q, 0) = 0 \]

\( (1) \)
\[ a_\beta = a_p + a_q - \min(b_p, a_q) \quad \text{(12)} \]
\[ = b_p + a_q - a_p \quad \text{as} \quad b_q = a_p \]
\[ = a_p + 0 \]
\[ = a_p + \max(a_q - b_p, 0) \quad \text{(13)} \]

\( (2) \)
\[ b_\beta = b_p + b_q - \min(b_p, a_q) \]
\[ = b_p + b_q - b_p \]
\[ = b_q + (b_p - b_p) \]
\[ = b_q + 0 \]
\[ = b_q + \max(b_p - a_q, 0) \quad \text{using (12)} \quad \text{(14)} \]

By (5), (6), (9), (10), (13) and (14) we conclude:

\[ a_\beta = a_p + a_q - \max(a_q, b_p, 0) \]
\[ b_\beta = b_p + \max(b_p, a_q, 0) \quad \text{for all possible three cases} \]
The theorem can be generalized for more number of job blocks as stated:

Let n jobs 1, 2, 3, ...........n are processed through two machines A & B in order AB with processing time aᵢ & bᵢ (i = 1, 2, 3, ........n) on machine A & B respectively.

If \((a_{i_0}, b_{i_0}) O_{i,w} (a_{i_1}, b_{i_1}) O_{i,w} (a_{i_2}, b_{i_2}) O_{i,w} ................. O_{i,w} (a_{i_p}, b_{i_p}) = (a_\beta, b_\beta)\)

Then \((a_\beta = a_{i_0} + \sum_{j=1}^{p} \max \{a_{ij} - b_{i(j-1)} \} \) and \((b_\beta = b_{i_p} + \sum_{j=1}^{p} \max \{b_{i(j-1)} - a_{ij}, 0\} )\)

where \(i_0, i_1, i_2, i_3, .................i_p \in \{1, 2, 3 ..............n\} \) and \(\beta\) is the equivalent job for job block \((i_0, i_1, i_2, i_3, .................i_p)\). The proof can be made using Mathematical induction technique on the lines of Maggu & Das [60]

2.1.3 Notations

\(a_i\) : Processing time for job i on machine
\(b_i\) : Processing time for job i on machine B
\(t_i\) : transportation time from machine A to B
\(S\) : Obtained best/optimal sequence
\(J_r\) : Scheduled Partial r planned jobs.
\(J_r'\) : The set of rest (n-r) left jobs.

2.1.4 Mathematical Model

Assumed n jobs are to be carried-out on two machines P and Q in the manner PQ has processing time \(a_i\) and \(b_i\) on every individual machine. Let jobs r and s are carried out together as job-block \((r, s)\) on the machines P and Q. \(t_i\) is a transportation time from machine P to Q. The numerical procedure of the model in structure can be expressed as in Tableau - 2.1.4.1.
Our aim is to acquire the ideal schedule of all assigned jobs for minimization of the make span utilizing branch and bound technique.

2.1.5 Algorithm

Step 1:
Developed two fictitious machines M & N with their processing time $M_i$ & $N_i$ as follows:

$$M_i = p_i + t_i \quad \text{and} \quad N_i = q_i + t_i$$

Step 2:
Establish job block $\beta = (m, n)$ using idle/waiting time operator:

$$O_{i,w}[(x_1, y_1), (x_2, y_2)] = (x_1, y_1) \ O_{i,w} (x_2, y_2) = \{ x_1 + \max ((x_2, y_1, 0), y_2 + \max (y_1, x_2, 0)) \}$$

Step 3:
Compute the least bounds using the formula given below:

(i) $$l_1 = t(j_r,1) + \sum_{i \in j_r} G_i + \min_{i \in j_r} (H_i)$$

(ii) $$l_2 = t(j_r,2) + \sum_{i \in j_r} H_i$$

Step 4:
Compute $l = \max (l_1, l_2)$, evaluate $l$ first for the $n$ classes of permutations, i.e. for these starting with 1, 2, 3……..n respectively, having labelled the appropriate vertices of the scheduling tree by these values.

Step 5:
Now explore the vertex with lowest label. Evaluate $l$ for the $(n-1)$ subclasses starting
with this vertex and again concentrate on the lowest label vertex. Continuing this way, until we reach at the end of the tree represented by two single permutations, for which we evaluate the total work time. Thus we get the optimal schedule of the jobs.

**Step 6:**
Find the In-Out table for the obtained optimal sequence

### 2.1.6 Numerical Illustration
Let us assume 5 jobs be processed on two machines P and Q. Jobs 2 and 4 are processed as a group job (2, 4). Our objective is to obtain the optimal schedule of all jobs which minimize the total elapsed time as described in Tableau 2.1.6.1.

<table>
<thead>
<tr>
<th>Jobs</th>
<th>Machine P</th>
<th>Transportation time p → q</th>
<th>Machine Q</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>p_i</td>
<td>t_i</td>
<td>q_i</td>
</tr>
<tr>
<td>1</td>
<td>9</td>
<td>4</td>
<td>17</td>
</tr>
<tr>
<td>2</td>
<td>13</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>5</td>
<td>17</td>
<td>8</td>
<td>5</td>
</tr>
</tbody>
</table>

**Tableau: 2.1.6.1**

**Solution:**

**As per step 1:** Prepare table for M_i & N_i as in Tableau:2.1.6.2

<table>
<thead>
<tr>
<th>Jobs</th>
<th>M_i</th>
<th>N_i</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13</td>
<td>21</td>
</tr>
<tr>
<td>2</td>
<td>16</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>14</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>25</td>
<td>13</td>
</tr>
</tbody>
</table>

**Tableau:-2.1.6.2**

**As per step 2:** For job block (2,4) the equivalent processing time as given below:
\{(p_2 + \max(p_4 - q_2, 0), (q_4 + \max(q_2 - p_4)\}\}

The new reduced problem is:

<table>
<thead>
<tr>
<th>Jobs</th>
<th>M_i</th>
<th>N_i</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13</td>
<td>21</td>
</tr>
<tr>
<td>β</td>
<td>16</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>14</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>25</td>
<td>13</td>
</tr>
</tbody>
</table>

**Tableau: 2.1.6.3**

<table>
<thead>
<tr>
<th>Node (J_r)</th>
<th>LB (J_r)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>77</td>
</tr>
<tr>
<td>β</td>
<td>79</td>
</tr>
<tr>
<td>3</td>
<td>81</td>
</tr>
<tr>
<td>5</td>
<td>88</td>
</tr>
<tr>
<td>1β</td>
<td>77</td>
</tr>
<tr>
<td>13</td>
<td>81</td>
</tr>
<tr>
<td>15</td>
<td>80</td>
</tr>
<tr>
<td>1β3</td>
<td>81</td>
</tr>
<tr>
<td>1β5</td>
<td>77</td>
</tr>
</tbody>
</table>

**Tableau: 2.1.6.4**

We have, LB (1) = 77,
LB (β) = 79,
LB (3) = 81,
LB (5) = 88

As per step 5:

LB(1β) = 77,
LB(13) = 81,
LB(15) = 80,
LB(1β3) = 81,
LB(1β5) = 77,
Therefore, sequence is $S_1 (1-2-4-5-3)$
Therefore, sequence is $S_1 (1-2-4-5-3)$

FIG: 2.1.6.1

The In – Out table for the sequence 1-2-4-5-3 is as shown in Tableau:2.1.6.5

<table>
<thead>
<tr>
<th>Jobs</th>
<th>Machines P</th>
<th>$t_i$</th>
<th>Machine Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0-9</td>
<td>4</td>
<td>13-30</td>
</tr>
<tr>
<td>2</td>
<td>9-22</td>
<td>3</td>
<td>30-42</td>
</tr>
<tr>
<td>4</td>
<td>22-29</td>
<td>3</td>
<td>42-54</td>
</tr>
<tr>
<td>5</td>
<td>29-46</td>
<td>8</td>
<td>54-59</td>
</tr>
<tr>
<td>3</td>
<td>46-59</td>
<td>6</td>
<td>65-68</td>
</tr>
</tbody>
</table>

Tableau: 2.1.6.5

Total Elapsed Time
= 68 units.
2.2 TWO STAGE FLOW SHOP SCHEDULING PROBLEM INCLUDING IDLE/WAITING TIME OPERATOR O_{i,w} FOR A JOB BLOCK WHERE JOBS ARE PERFORMED IN THE STRING OF DISJOINT JOB BLOCKS

The scheduling problems are regular event in day by day living e.g. sequence of jobs for processing in an assembling lodge, projects meant to be running in an arrangement at a PC focus and so forth. Fundamental work in the ground of sequencing was proposed by Johnson, S.M. [45] gave technique for decision the optimal sequence for n jobs by m machines flow shop scheduling problem with the core objective of minimization of the make span. Ignall, E. and Schrage, L. [43] implemented Branch and Bound strategy for obtaining a sequence for the minimization of flow time. Chandrasekharan, R. [18] has specified cases to acquire a schedule which reduce all elapsed time subject to least make span. Anup [4], Singh, T.P. [80], [81], Bagga, P.C. [6], Maggu, P.L. and Das, G. [60], Yoshida and Hitomi [92], Szwarch, S. [86], and so forth inferred the best method for two or multistage flow shop problems considering the different restraints and criterion. In this section we have performed all the jobs in a string with ease of job block system.

This section is based on the paper “Branch and Bound Technique in Two Stage Flow Shop Scheduling Problem Including idle/waiting Time Operator O_{i,w} For a Job Block Where Jobs Are Performed in The Form of String”, published in International Journal of Advanced Research in Computer Engineering & Technology, Vol. 1 (7), 2012.

2.2.1 Algorithm

**Step 1:** Determine equivalent jobs for each job blocks using idle/waiting Time Operator as following:

\[(a_p, b_p) O_{i,w} (a_q, b_q) = (a_\beta, b_\beta)\]

then

\[a_\beta = a_p + \max (a_q - b_p, 0)\]

and

\[b_\beta = b_q + \max (b_q - a_q, 0)\]

Where \(\beta\) is the equivalent job for job block \((p, q)\) and \(p, q \in \{1, 2, 3, \ldots, n\}.\)
Step 2:
Calculate the lower bounds using the following formula:

\[(i)\quad l_1 = t(j_r, 1) + \sum_{i \in j_r} G_i + \min_{i \in j_r}(H_i)\]

\[(ii)\quad l_2 = t(j_r, 2) + \sum_{i \in j_r} H_i\]

Step 3:
Calculate \(l = \max(l_1, l_2)\)

Step 4:
We evaluate \(l\) first for the \(n\) classes of permutations, i.e. for these starting with 1, 2, 3, \ldots, \(n\) respectively, having labelled the appropriate vertices of the scheduling tree by these values.

Step 5:
Now explore the vertex with lowest label. Evaluate \(l\) for the \((n-1)\) subclasses starting with this vertex and again concentrate on the lowest label vertex. Continuing this way, until we reach at the end of the tree represented by two single permutations, for which we evaluate the total work time. Thus we get the optimal schedule of the jobs.

Step 6:
Prepare in-out table for the optimal sequence obtained in step 3 and get the minimum total elapsed time.

2.2.2 Numerical Illustration
Consider 6 jobs 2 machine flow shop problem whose processing time of the jobs on each machine is given and jobs are to be processed in job block (2,4) with transportation time as shown in tableau 2.2.2.1

<table>
<thead>
<tr>
<th>Jobs</th>
<th>Machine P</th>
<th>Machine Q</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(P_i)</td>
<td>(Q_i)</td>
</tr>
<tr>
<td>1</td>
<td>17</td>
<td>28</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>35</td>
</tr>
<tr>
<td>3</td>
<td>23</td>
<td>30</td>
</tr>
<tr>
<td>4</td>
<td>37</td>
<td>34</td>
</tr>
<tr>
<td>5</td>
<td>19</td>
<td>38</td>
</tr>
<tr>
<td>6</td>
<td>22</td>
<td>16</td>
</tr>
</tbody>
</table>

Tableau:2.2.2.1
**Step 1:** Determine equivalent jobs for each job blocks using idle/waiting Time Operator as following we get (Tableau - 2.2.2.2)

<table>
<thead>
<tr>
<th>Jobs</th>
<th>Machine $P_i$</th>
<th>Machine $Q_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>17</td>
<td>43</td>
</tr>
<tr>
<td>$\beta$</td>
<td>30</td>
<td>34</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>19</td>
<td>22</td>
</tr>
</tbody>
</table>

Tableau: 2.2.2.2

Again applying equivalent job block criteria on jobs ($\beta, \gamma$) we get

<table>
<thead>
<tr>
<th>Jobs</th>
<th>Machine $P_i$</th>
<th>Machine $Q_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>17</td>
<td>43</td>
</tr>
<tr>
<td>$\eta$</td>
<td>30</td>
<td>37</td>
</tr>
</tbody>
</table>

Tableau: 2.2.3

**Step 2:**
$J_1 = (\alpha)$,
$J_{1'} = (\eta)$

**Step 3:**
Whereas, $t_1 = 94$ and
$t_2 = 84$

$LB (\alpha) = \text{maximum (84, 97)} = 97$

Also, $LB (\eta) = 110$

**Step 4:**
So, optimised schedule is $S_i = \alpha, \eta$
i.e. $S_i = 1-2-3-4-5-6$
Step 5:
Compute In–out table for $S_i$ and the reduction of make span as in Tableau: 2.2.2.4.

<table>
<thead>
<tr>
<th>Jobs</th>
<th>Machine P</th>
<th>Machine Q</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>In –Out</td>
<td>In –Out</td>
</tr>
<tr>
<td>1</td>
<td>0-17</td>
<td>17-45</td>
</tr>
<tr>
<td>2</td>
<td>17-37</td>
<td>45-80</td>
</tr>
<tr>
<td>3</td>
<td>37-60</td>
<td>80-110</td>
</tr>
<tr>
<td>4</td>
<td>60-97</td>
<td>110-144</td>
</tr>
<tr>
<td>5</td>
<td>97-116</td>
<td>144-182</td>
</tr>
<tr>
<td>6</td>
<td>116-138</td>
<td>182-198</td>
</tr>
</tbody>
</table>

Tableau: 2.2.2.4

So, the total elapsed time is 198 units.
Conclusion

This chapter emphasis on the idle/waiting time criteria in which we have to give priority of one job over another in the case of emergency etc. like emergency cases in hospitals, landing of aircrafts etc. The study may further be reached out by presenting diverse parameters, for example, setup time separated from their processing time, interval of break-down by linking probabilities with their handling times or additionally.