Journal of Global Optimization (2005) 33: 349–367
DOI 10.1007/s10898-004-1848-y


# Flowshop/No-idle Scheduling to Minimize Total Elapsed Time

LAXMI NARAIN[1] and P.C. BAGGA[2]
[1]*Department of Mathematics, University of Delhi, Delhi, India*
*(E-mail: laxmi_narain_2004@yahoo.com);*
[2]*Moti Lal Nehru College (E), University of Delhi, Delhi, India*

(Received 29 April 2003; accepted in revised form 5 August 2004)

**Abstract.** This paper studies four n-job, m-machine flowshop problems when processing times of jobs on various machines follow certain conditions. The objective is to obtain a sequence, which minimizes total elapsed time under no-idle constraint. Under no-idle constraint, the machines work continuously without idle-interval. We prove two theorems. We introduce simple algorithms without using branch and bound technique. Numerical examples are also given to demonstrate the algorithms.

**Mathematics Subject Classification (1991).** 90B35.

**Key words:** Completion time, Optimization, Sequencing theory, Total elapsed time

## 1. Introduction

Many research papers exist in literature, which deal with $n$ jobs, $m$ machines flowshop problems with minimization of total elapsed time as the criterion. Johnson (1954) made first attempt in this direction for scheduling $n$ jobs on two machines. The work of Johnson was extended to m machines by Dudek and Teuton (1964), Smith and Dudek (1967), Bagga and Chakravarti (1968), Gupta (1969) and several others. Johnson (1954), Panwalker and Khan (1975), Szwarc (1977) and Bagga and Ambika (1997) studied the problem, when the processing times of jobs on the machines satisfy certain conditions, where the minimization of total elapsed time has been taken as the criterion for optimization.

In most of the literature, it is assumed that machines are available in the starting of processing the jobs. But situation may arise, when one has got the assignment but does not have one's own machines or does not have enough money or does not want to take risk of investing money for the purchase of machines. Under these circumstances, may take machines on rent to complete the assignment. For example, in a realistic situation, if a programmer gets the work of computerizing examination results of a board/university, he may need three machines, viz., Data entry machine, Computer and Printer, to be taken on rent. In this case, objective can be when should these machines be taken on rent so that total rental cost is minimum.

We know that total rental cost $= \sum_{j=1}^{m}\sum_{i=1}^{n}\left[p_{i,j} + I_{i,j}\right] \times C_j$ where $p_{i,j}$ is the processing time of $i$th job on machine $M_j$; $I_{i,j}$ is the idle time of machine $M_j$ for job $i$, i.e., the time machine $M_j$ is waiting for job $i$ after completing the processing of job and $C_j$ is rental cost per unit time of machine $M_j$. The processing times $p_{i,j}$ and rental costs $C_j$ are constants. Therefore, total rental cost is minimum when idle times on machines are minimum.

Under no-idle constraint, machines work continuously without idle interval, i.e., each machine after starting the processing of first job will work without break till the last job completed on it. Therefore, $I_{i,j} = 0 \quad \text{for}\, i = 1, 2, \ldots, n, \quad j = 1, 2, \ldots, m.$

Therefore, total rental cost under no-idle constraint $= \sum_{j=1}^{m}\sum_{i=1}^{n} p_{i,j} \times C_j$, which is least for any sequence.

Adiri and Pohoryles (1982) studied $n$-job, $m$-machine flowshop problem under no-idle constraint with the objective being minimization of sum of completion times for increasing or decreasing series of dominating machines. Narain and Bagga (2003) studied three-machine general flowshop problem under no-idle constraint with the objective being minimization of total elapsed time.

The present paper studies four $n$-job, $m$-machine flowshop problems when processing times of jobs on various machines follow certain conditions. The objective in each case is to obtain a sequence, which gives minimum possible total elapsed time under no-idle constraint. In Section 2, we provide notations. In Section 3, we provide problem formulation and theorems. In Section 4, we provide algorithm for each of the four problems without using branch and bound technique, and examples to demonstrate each algorithm.

## 2. Notations

| | | |
|---|---|---|
| $S$ | = | sequence of jobs 1, 2,…, $n$. |
| $p_{i,j}$ | = | processing time of job $i$ on machine $M_j$. |
| $Z_{i,j}$ | = | completion time of $i$th job on machine $M_j$ when all the machines are taken on rent at the same time, i.e., in the starting of processing the jobs. |
| $I_{i,j}$ | = | idle time of machine $M_j$ for job $i$. |
| $C_j$ | = | rental cost per unit time of machine $M_j$ |
| $H_j$ | = | the time (earliest) at which machine $M_j$ should be taken on rent to process jobs continuously without idle interval. |
| $Z'_{i,j}$ | = | completion time of $i$th job on machine $M_j$ when $M_j$ starts processing jobs at time $H_j$. |

$I'_{i,j}$ = idle time of machine $M_j$ for job $i$ when $M_j$ starts processing jobs at time $H_j$.
$i$ = 1, 2,. . ., $n$
$j$ = 1, 2, . . ., $m$

## 3. Mathematical Formulation

For $n$-job, $m$-machine flowshop problem the problem can be mathematically formulated as obtaining a sequence $S^*$, which satisfies the bicriteria

Minimize $Z_{n,m}$

$$\text{Subject to} \sum_{j=1}^{m}\sum_{i=1}^{n} I_{i,j} = 0, \text{ for the sequzence } S^*.$$

Idle time of first machine $M_1$ for every sequence is zero, i.e., $M_1$ always process jobs continuously without idle interval. Therefore, the time at which machine $M_1$ should start processing jobs continuously without idle interval is zero, i.e., $H_1 = 0$.

To find the time at which machine $M_j$, $j = 2, 3, \ldots, m$ should be taken on rent to process job continuously without idle interval, we prove the following theorems.

THEOREM 3.1. *The time at which machine $M_r$ should be taken on rent (or starts processing jobs) to have zero idle time on $M_r$ is*

$$H_r = \max_{1 \leqslant k \leqslant n}\{Y_k\} \quad r = 2, 3, \ldots, m$$

*where*

$$Y_k = Z'_{k,r-1} - \sum_{i=1}^{k-1} p_{i,r} \quad \text{for } k > 1$$

$$Y_1 = Z'_{1,r-1}$$

*Proof*. Proof is based on mathematical induction. It will be shown that if machine $M_r$ starts processing jobs at time $H_r$, then idle time of $M_r$ is zero.

For $r = 2$;

$$H_2 = \max_{1 \leqslant k \leqslant n}\{Y_k\}$$

$$\text{Let } Y_q = \max_{1 \leqslant k \leqslant n}\{Y_k\}$$

$$\text{Therefore, } Y_q \leqslant Y_k \quad \text{for } k = 1, 2, \ldots, n$$

For $k = 1$;

$$Yq \geqslant Y_1$$

i.e., $H_2 \geqslant Y_1$

i.e., $H_2 \geqslant Z'_{1.1}$

i.e., $Z'_{1,1} \leqslant H_2 \quad (1)$

From Equation (1); if machine $M_2$ is taken on rent at time $H_2$, then it will start processing the first job without waiting. Therefore, idle time of machine $M_2$ for 1st job is zero when it starts processing jobs at time $H_2$.

For $k = 2, 3, \ldots, n$,

$$Y_q \geq Y_k$$

i.e., $Y_q + \sum_{i=1}^{k-1} p_{i,2} \geqslant Y_k + \sum_{i=1}^{k-1} p_{i,2}$

i.e., $H_2 + \sum_{i=1}^{k-1} p_{i,2} \geqslant Z'_{k,1} - \sum_{i=1}^{k-1} p_{i,2} + \sum_{i=1}^{k-1} p_{i,2}$

i.e., $Z'_{k-1,2} \geqslant Z'_{k,1}$

i.e., $Z'_{k,1} \leqslant Z'_{k-1,2} \quad \text{for } k = 2, 3, \ldots, n \quad (2)$

$$I'_{k,2} = \max\left[Z'_{k,1} - Z'_{k-1,2}, 0\right]$$

From Equation (2), $I'_{k,2} = 0 \quad \text{for } k = 2, 3, \ldots, n$

Therefore, the result holds for $r = 2$.

Let the result holds for $r = s$.

For $r = s + 1$;

$$H_{s+1} = \max_{1 \leqslant k \leqslant n} \{Y_k\}$$

Let $Y_t = \max_{1 \leqslant k \leqslant n} \{Y_k\}$

Therefore, $Y_t \geqslant Y_k$ for $k = 1, 2, \ldots, n$

For $k = 1$;

$$Y_t \geqslant Y_1$$

i.e., $H_{s+1} \geqslant Y_1$

i.e., $H_{s+1} \geqslant Z'_{s,1}$

$$\text{i.e.,}\quad Z'_{s,1} \leqslant H_{s+1} \tag{3}$$

From Equation (3), if machine $M_{s+1}$ is taken on rent at time $H_{s+1}$, then it will start processing the first job without waiting. Therefore, idle time of machine $M_{s+1}$ for 1st job is zero when it starts processing jobs at time $H_{s+1}$.

For $k = 2, 3, \ldots, n$;

$Y_t \geqslant Y_k$

$$\text{i.e.,}\quad Y_t + \sum_{i=1}^{k-1} p_{i,s+1} \geqslant Y_k + \sum_{i=1}^{k-1} p_{i,s+1}$$

$$\text{i.e.,}\quad H_{s+1} + \sum_{i=1}^{k-1} p_{i,s+1} \geqslant Z'_{k,s} - \sum_{i=1}^{k-1} p_{i,s+1} + \sum_{i=1}^{k-1} p_{i,s+1}$$

$$\text{i.e.,}\quad Z'_{k-1,s+1} \geqslant Z'_{k,s}$$

$$\text{i.e.,}\quad Z'_{k,s} \leqslant Z'_{k-1,s+1} \quad \text{for } k = 2, 3, \ldots, n \tag{4}$$

$$I'_{k,s+1} = \max\left[Z'_{k,s} - Z'_{k-1,s+1}, 0\right]$$

From Equation (4), $I'_{k,s+1} = 0$ for $k = 2, 3, \ldots, n$

Therefore, the result holds for $r = s + 1$ also.

Hence, by mathematical induction, this theorem holds for all $r$, where $r = 2, 3, \ldots, m$.

THEOREM 3.2. *There will be idle time on machine $M_r$ if it is taken on rent at time*

$$H_r < \max_{1 \leqslant k \leqslant n}\{Y_k\}, \quad r = 2, 3, \ldots, m$$

*where*

$$Y_k = Z'_{k,r-1} - \sum_{i=1}^{k-1} p_{i,r} \quad \text{for } k > 1$$

$$Y_1 = Z'_{1,r-1}$$

*Proof.* Proof is based on mathematical induction. It will be shown that if machine $M_r$ is taken on rent at time $H_r < \max_{1 \leqslant k \leqslant n}\{Y_k\}$, then there will be idle time on $M_r$.

For $r = 2$;

Let $Y_q = \max_{1 \leqslant k \leqslant n} \{Y_k\}$

$H_2 < \max_{1 \leqslant k \leqslant n} \{Y_k\}$

Therefore, $H_2 < Y_q$

Now, there arises two cases.

**Case 1**: $q = 1$;

$$H_2 < Y_1$$

i.e., $H_2 < Z'_{1,1}$

i.e., $Z'_{1,1} > H_2$ (5)

Machine $M_2$ will remain idle for time $\geqslant Z'_{1,1} - H_2$

From Equation (5), $Z'_{1,1} - H_2 > 0$

Therefore, idle time of machine $M_2$ for 1st job will be greater than zero when it is taken on rent at time $H_2 < Y_1$.

**Case 2**: $q > 1$;

$$H_2 < Yq$$

i.e., $H_2 + \sum_{i=1}^{q-1} p_{i,2} < Y_q + \sum_{i=1}^{q-1} p_{i,2}$

i.e., $Z'_{q-1,2} < Z'_{q,1} - \sum_{i=1}^{q-1} p_{i,2} + \sum_{i=1}^{q-1} p_{i,2}$

i.e., $Z'_{q-1,2} < Z'_{q,1}$

i.e., $Z'_{q,1} < Z'_{q-1,2}$ (6)

Therefore, $I'_{q,2} \geqslant \max\lfloor Z'_{q,1} - Z'_{q-1,2}, 0\rfloor$

From equation (6), $I'_{q,2} \geqslant Z'_{q,1} - Z'_{q-1,2} > 0$

Therefore, idle time of machine $M_2$ for $q$th job will be greater than zero when it is taken on rent at time $H_2 < \max_{1 \leqslant k \leqslant n}\{Y_k\}$.

Therefore, the result holds for $r = 2$.

Let the result holds for $r = s$.

For $r = s + 1$;

$H_{s+1} < \max_{1 \leqslant k \leqslant n} \{Y_k\}$

Let $Y_t = \max_{1 \leqslant k \leqslant n} \{Y_k\}$

Therefore, $H_{s+1} < Y_t$

Now, there arises two cases.

**Case 1**: $t = 1;$

$$H_{s+1} < Y_1$$
$$\text{i.e.,}\quad H_{s+1} < Z'_{1,s} \qquad (7)$$
$$\text{i.e.,}\quad Z'_{1,s} > H_{s+1}$$

Machine $M_{s+1}$ will remain idle for time $\geqslant Z'_{1,s} - H_{s+1}$.

From Equation (7), $Z'_{1,s} - H_{s+1} > 0$

Therefore, idle time of machine $M_{s+1}$ for 1st job will be greater than zero when it is taken on rent at time $H_{s+1} < Y_1$.

**Case 2**: $t > 1;$

$$H_{s+1} < Y_t$$
$$\text{i.e.,}\quad H_{s+1} + \sum_{i=1}^{t-1} p_{i,s+1} < Y_t + \sum_{i=1}^{t-1} p_{i,s+1}$$
$$\text{i.e.,}\quad Z'_{t-1,s+1} < Z'_{t,s} - \sum_{i=1}^{t-1} p_{i,s+1} + \sum_{i=1}^{t-1} p_{i,s+1}$$
$$\text{i.e.,}\quad Z'_{t-1,s+1} < Z'_{t,s}$$
$$\text{i.e.,}\quad Z'_{t,s} > Z'_{t-1,s+1} \qquad (8)$$

Therefore, $I'_{t,s+1} \geqslant \max\lfloor Z'_{t,s} - Z'_{t-1,s+1}, 0 \rfloor$

From Equation (8), $I'_{t,s+1} \geqslant Z'_{t,s} - Z'_{t-1,s+1} > 0$

Therefore, idle time of machine $M_{s+1}$ for $t$th job will be greater than zero when it is taken on rent at time $H_{s+1} < \max_{1 \leqslant k \leqslant n}\{Y_k\}$.

Therefore, the result holds for $r = s + 1$ also.

Hence, by mathematical induction the result holds for all $r$, where $r = 2, 3, \ldots, m$.

From Theorems 3.1 and 3.2, the earliest time at which machine $M_r$ should start processing jobs continuously without idle interval is

$$H_r = \max_{1 \leqslant k \leqslant n} \{Y_k\}$$
$$= \max_{1 \leqslant k \leqslant n} \left\{ Z'_{k,r-1} - \sum_{i=1}^{k-1} p_{i,r} \right\}$$
$$= \max_{1 \leqslant k \leqslant n} \left\{ H_{r-1} + \sum_{i=1}^{k} p_{i,r-1} - \sum_{i=1}^{k-1} p_{i,r} \right\}$$

$$= H_{r-1} + \max_{1 \leqslant k \leqslant n} \left\{ \sum_{i=1}^{k} p_{i,r-1} - \sum_{i=1}^{k-1} p_{i,r} \right\}$$

$$= H_{r-1} + I_r \tag{9}$$

where $I_r = \max_{1 \leqslant k \leqslant n} \left\{ \sum_{i=1}^{k} p_{i,r-1} - \sum_{i=1}^{k-1} p_{i,r} \right\}$ is the idle time of machine $M_r$ for machine pair ($M_{r-1}$, $M_r$) as a two machines flowshop sequencing problem.

Total elapsed time when all the machines process jobs continuously without idle interval is

$$Z'_{n,m} = H_m + \sum_{i=1}^{n} p_{i,m} \tag{10}$$

From Equation (10), total elapsed time will be minimum when $H_m$ will be minimum.

## 4. Special Flowshop Problems

**Case 1**: $p_{i,k} \leqslant p_{j,k+1}, \quad \forall i, j, \quad i \neq j, \quad k = 1, 2, \ldots, m-2.$

Here $$Z_{n,1} = \sum_{i=1}^{n} p_{i,1} \tag{11}$$

The expression for $Z_{n,2}$ is obtained as follows:

$$Z_{1,2} = p_{1,1} + p_{1,2}$$

$$Z_{2,2} = \max(Z_{1,2}, Z_{2,1}) + p_{2,2} \tag{12}$$

From Equations (11) and (12);

$$Z_{2,2} = \max(p_{1,1} + p_{1,2}, p_{1,1} + p_{2,1}) + p_{2,2}$$

$$= p_{1,1} + p_{1,2} + p_{2,2}$$

$$= p_{1,1} + \sum_{i=1}^{2} p_{i,2}$$

$$Z_{3,2} = \max(Z_{3,1},\ Z_{2,2}) + p_{3,2} \tag{13}$$

From Equations (11) and (13);

$$Z_{3,2} = \max(p_{1,1} + p_{2,1} + p_{3,1}, p_{1,1} + p_{1,2} + p_{2,2}) + p_{3,2}$$

$$= p_{1,1} + p_{1,2} + p_{2,2} + p_{3,2}$$

$$= p_{1,1} + \sum_{i=1}^{3} p_{i,2} \tag{14}$$

Continuing in this way;

$$Z_{n,2} = p_{1,1} + \sum_{i=1}^{n} p_{i,2} \tag{15}$$

Now, the times at which machines should be taken on rent to process jobs continuously without idle interval are :

$$H_1 = 0$$

From Theorem 3.1;

$$H_2 = \max_{1 \le u \le n} \left\{ Z'_{u,1} - \sum_{i=1}^{u-1} p_{i,2} \right\}$$

$$= \max_{1 \le u \le n} \left\{ H_1 + \sum_{i=1}^{u} p_{i,1} - \sum_{i=1}^{u-1} p_{i,2} \right\}$$

$$\text{Since} \quad H_1 = 0; \text{therefore, } H_2 = \max_{1 \le u \le n} \left[ \sum_{i=1}^{u} p_{i,1} - \sum_{i=1}^{u-1} p_{i,2} \right] = p_{1,1} \tag{16}$$

$$H_3 = \max_{1 \leqslant u \leqslant n} \left\{ Z'_{u,2} - \sum_{i=1}^{u-1} p_{i,3} \right\}$$

$$= \max_{1 \leqslant u \leqslant n} \left\{ H_2 + \sum_{i=1}^{u} p_{i,2} - \sum_{i=1}^{u-1} p_{i,3} \right\}$$

$$= H_2 + \max_{1 \leqslant u \leqslant n} \left\{ \sum_{i=1}^{u} p_{i,2} - \sum_{i=1}^{u-1} p_{i,3} \right\}$$

$$= H_2 + p_{1,2}$$

From Equation (16);

$$H_3 = p_{1,1} + p_{1,2} = \sum_{k=1}^{m-2} p_{1,k} \tag{17}$$

Continuing in this way;

$$H_{m-1} = \sum_{k=1}^{m-2} p_{1,k} \tag{18}$$

From Equations (16), (17) and (18),

$$H_r = \sum_{k=1}^{r-1} p_{1,k}, \qquad r = 2, 3, \ldots, m-1 \tag{19}$$

Taking $r = m$ in Equation (9),

$$H_m = H_{m-1} + I_m$$

From Equation (19),

$$H_m = \sum_{k=1}^{m-2} p_{1,k} + I_m \tag{20}$$

$I_m$ is idle time of machine $M_m$ for machine pair ($M_{m-1}$, $M_m$) as a two-machine flowshop sequencing problem. $H_m$ is minimum when $\sum_{k=1}^{m-2} p_{1,k}$ and $I_m$ are minimum. $I_m$ is minimum when we apply Johnson's two-machine algorithm on $M_{m-1}$ and $M_m$. The following algorithm gives the sequence, which minimizes total elapsed time under no-idle constraint.

**ALGORITHM 4.1.**

**Step 1:** Obtain sequence $S$ by applying Johnson's two-machine algorithm on machines $M_{m-1}$ and $M_m$. *In case of a tie for the first position of the sequence* $S$, prefer that job for the first position for which sum of the processing times on $M_1$, $M_2$,. . .,$M_{m-2}$ is minimum.

**Step 2:** If the sum of the processing times of the first job of sequence $S$ on $M_1$, $M_2$,. . .,$M_{m-2}$ is minimum, then $S$ is an optimal sequence. Otherwise, go to Step 3.

**Step 3:** Obtain other sequence from $S$ by substituting that job in the first position whose sum of the processing times on $M_1$, $M_2$,. . .,$M_{m-2}$ is less than the sum of the processing times of first job of $S$ and without disturbing the remaining order of $S$.

**Step 4:** Repeat Step 3 till all such jobs have been placed in the first position. Let these sequences be $S_1$, $S_2$,. . .,$S_{r_1}$.

**Step 5:** Find out $H_m$ for the sequences $S$, $S_1$, $S_2$,. . .,$S_{r_1}$ using Equation (14). The sequence(s) having minimum $H_m$ is (are) the required sequence(s).

**Step 6:** For the sequence(s) obtained in step 5, find out total elapsed time by Equation (10) and $H_2$, $H_3$,. . .,$H_{m-1}$ by Equation (19).

EXAMPLE 4.1. Consider the 5-job, 4-machine sequencing problem whose processing times are given as in Table 1.

*Table 1.* Processing times of jobs

| Jobs | Machines | | | |
|---|---|---|---|---|
| | $M_1$ | $M_2$ | $M_3$ | $M_4$ |
| 1 | 6 | 8 | 13 | 5 |
| 2 | 5 | 10 | 10 | 14 |
| 3 | 7 | 9 | 11 | 12 |
| 4 | 4 | 10 | 12 | 10 |
| 5 | 5 | 9 | 11 | 9 |

Applying Algorithm 4.1 Johnson's algorithm on $M_3$ and $M_4$ provides the sequence $S = $ 2-3-4-5-1. The sum of the processing times of job 2 on $M_1$ and $M_2$ is 15 units. Job 1, 4 and 5 have the sum of processing times on $M_1$ and $M_2$ less than 15 units. Therefore, the other sequences are $S_1 = $ 1-2-3-4-5; $S_2 = $ 4-2-3-5-1 and $S_3 = $ 5-2-3-4-1. $H_m$ for sequences $S$, $S_1$, $S_2$, $S_3$ are 27 units, 32 units, 26 units, 26 units respectively. $H_m$ is minimum for the sequences $S_2$ and $S_3$. Therefore, 4-2-3-5-1 and 5-2-3-4-1 are optimal sequences. The total elapsed time under no-idle constraint is 76 units. For sequence 4-2-3-5-1; $H_1$, $H_2$, $H_3$ are 0 unit, 4 units, 14 units respectively. For sequence 5-2-3-4-1; $H_1$, $H_2$, $H_3$ are 0 unit, 5 units, 14 units respectively.

**Case 2**: $p_{i,k} \geqslant p_{j,k+1}, \quad \forall i,j \quad i \neq j, \quad k = 1, 2, \ldots, m-2.$

$$\text{Here } Z_{n,1} = \sum_{i=1}^{n} p_{i,1} \tag{21}$$

The expression for $Z_{n,2}$ is obtained as follows:

$$\begin{aligned} Z_{2,1} &= p_{1,1} + p_{1,2} \\ Z_{2,2} &= \max(Z_{2,1}, Z_{1,2}) + p_{2,2} \end{aligned} \tag{22}$$

From Equations (21) and (22);

$$\begin{aligned} Z_{2,2} &= \max(p_{1,1} + p_{2,1}, p_{1,1} + p_{1,2}) + p_{2,2} \\ &= p_{1,1} + p_{2,1} + p_{2,2} \\ &= \sum_{i=1}^{2} p_{i,1} + p_{2,2} \end{aligned} \tag{23}$$

$$Z_{3,2} = \max(Z_{3,1} + Z_{2,2}) + p_{3,2}$$

From Equations (21) and (23);

$$\begin{aligned} Z_{3,2} &= \max\,(p_{1,1} + p_{2,1} + p_{3,1}, p_{1,1} + p_{2,1} + p_{2,2}) + p_{3,2} \\ &= p_{1,1} + p_{2,1} + p_{3,1} + p_{3,2} \\ &= \sum_{i=1}^{3} p_{i,1} + p_{3,2} \end{aligned} \tag{24}$$

Continuing in this way;

$$Z_{n,2} = \sum_{i=1}^{n} p_{i,1} + p_{n,2} \tag{25}$$

Now, the times at which machines should be taken on rent to process jobs continuously without idle interval are:

$H_1 = 0$

From Theorem 3.1,

$$H_2 = \max_{1 \leqslant u \leqslant n} \left\{ Z'_{u,1} - \sum_{i=1}^{u-1} p_{i,2} \right\}$$
$$= \max_{1 \leqslant u \leqslant n} \left\{ H_1 + \sum_{i=1}^{u} p_{i,1} - \sum_{i=1}^{u-1} p_{i,2} \right\}$$

Since $H_1 = 0$; therefore,

$$\begin{aligned} H_2 &= \max_{1 \leqslant u \leqslant n} \left\{ \sum_{i=1}^{u} p_{i,1} - \sum_{i=1}^{u-1} p_{i,2} \right\} \\ &= \sum_{i=1}^{n} p_{i,1} - \sum_{i=1}^{n-1} p_{i,2} \\ &= \sum_{i=1}^{n} p_{i,1} - \sum_{i=1}^{n} p_{i,2} + p_{n,2} \end{aligned} \tag{26}$$

$$\begin{aligned} H_3 &= \max_{1 \leqslant u \leqslant n} \left\{ Z'_{u,2} - \sum_{i=1}^{u-1} p_{i,3} \right\} \\ &= \max_{1 \leqslant u \leqslant n} \left( H_2 + \sum_{i=1}^{u} p_{i,2} - \sum_{i=1}^{u-1} p_{i,3} \right) \\ &= H_2 + \max_{1 \leqslant u \leqslant n} \left( \sum_{i=1}^{u} p_{i,2} - \sum_{i=1}^{u-1} p_{i,3} \right) \\ &= H_2 + \sum_{i=1}^{n} p_{i,2} - \sum_{i=1}^{n-1} p_{i,3} \end{aligned}$$

From Equation (26);

$$\begin{aligned} H_3 &= \sum_{i=1}^{n} p_{i,1} - \sum_{i=1}^{n} p_{i,2} + p_{n,2} + \sum_{i=1}^{n} p_{i,2} - \sum_{i=1}^{n} p_{i,3} + p_{n,3} \\ &= \sum_{i=1}^{n} p_{i,1} - \sum_{i=1}^{n} p_{i,3} + \sum_{k=2}^{3} p_{n,k} \end{aligned} \tag{27}$$

Continuing in this way;

$$H_{m-1} = \sum_{i=1}^{n} p_{i,1} - \sum_{i=1}^{n} p_{i,m-1} + \sum_{k=2}^{m-1} p_{n,k} \tag{28}$$

From Equations (26), (27) and (28);

$$H_r = \sum_{i=1}^{n} p_{i,1} - \sum_{i=1}^{n} p_{i,r} + \sum_{k=2}^{r} p_{n,k}, \quad r = 2, 3, \ldots, m-1 \tag{29}$$

Taking $r = m$ in Equation (9);

$$H_m = H_{m-1} + I_m$$

From Equation (28);

$$H_m = \sum_{i=1}^{\mathrm{n}} p_{i,1} - \sum_{i=1}^{n} p_{i,m-1} + \sum_{k=2}^{m-1} p_{n,k} + I_m \tag{30}$$

$I_{\mathrm{m}}$ is idle time of machine $M_m$ for machine pair ($M_{m-1}$, $M_m$) as a two-machine flowshop sequencing problem. Here $H_m$ is minimum when $\sum_{k=2}^{m-1} p_{n,k}$ and $I_{\mathrm{m}}$ are minimum. $I_m$ is minimum when we apply Johnson's algorithm on machines $M_{m-1}$ and $M_m$. The following algorithm gives the sequence which minimizes total elapsed time under no-idle constraint.

**ALGORITHM 4.2.**

**Step 1:** Obtain sequence $S$ by applying Johnson's two-machine algorithm on machines $M_{m-1}$ and $M_m$. In case of a tie for the last position of the sequence $S$, prefer that job for the last position for which the sum of the processing times on $M_2$, $M_3$,..., $M_{m-1}$ is minimum.

**Step 2:** If the sum of the processing times of the last job of S on $M_2$, $M_3$,..., $M_{m-1}$ is minimum, then $S$ is an optimal sequence. Otherwise, go to Step 3.

**Step 3:** Obtain other sequence from $S$ by substituting that job in the last position whose sum of the processing times on $M_2$, $M_3$,...,$M_{m-1}$ is less than the sum of the processing times of last job of $S$ and without disturbing the remaining order of $S$.

**Step 4:** Repeat Step 3 till all such jobs have been placed in the last position. Let these sequences be $S_1$, $S_2$,..., $S_{r_2}$.

**Step 5:** Find out $H_m$ for the sequences $S$, $S_1$, $S_2$,...,$S_{r_2}$ using Equation (30). The sequence(s) having minimum $H_m$ is (are) the required sequence(s).

**Step 6:** For the sequence(s) obtained in Step 5; find out total elapsed time using equation (10) and $H_2$, $H_3$,..., $H_{m-1}$ using Equation (29).

*Table 2.* Processing times of jobs

| Jobs | Machines | | | | |
|---|---|---|---|---|---|
| | $M_1$ | $M_2$ | $M_3$ | $M_4$ | $M_5$ |
| 1 | 16 | 12 | 9 | 5 | 10 |
| 2 | 14 | 11 | 10 | 7 | 5 |
| 3 | 13 | 10 | 8 | 6 | 6 |
| 4 | 12 | 12 | 7 | 7 | 8 |

EXAMPLE 4.2. Consider the 4-job, 5-machine sequencing problem whose processing times are given as in Table 2.

Applying Algorithm 4.2, Johnson's algorithm on $M_4$ and $M_5$ provides the sequence $S$ = 1-4-3-2. The sum of the processing times of job 2 on $M_2$, $M_3$ and $M_4$ is 28 units. Job 1, 3 and 4 have the sum of the processing times on $M_2$, $M_3$ and $M_4$ less than 28 units. Therefore, the other sequences are $S_1$ = 4-3-2-1; $S_2$ = 1-4-2-3 and $S_3$ = 1-3-2-4. $H_{\mathrm{m}}$ for sequences $S$, $S_1$, $S_2$, $S_3$ are 63 units, 63 units, 59 units, 61 units respectively. $H_{\mathrm{m}}$ is minimum for sequence 1-4-2-3. Therefore, 1-4-2-3 is the optimal sequence. The total elapsed time under no-idle constraint is 88 units. For sequence 1-4-2-3; $H_1$, $H_2$, $H_3$ and $H_4$ are 0 unit, 20 units, 39 units and 54 units respectively.

**Case 3**: $p_{i,k} \leqslant p_{j,k+1}$; $\forall i,j, \quad i \neq j, \quad k = 2, 3, \ldots, m-1.$

The times at which machines should be taken on rent to process jobs continuously without idle interval are:

$H_1 = 0$

From Theorem 3.1;

$$H_2 = \max_{1 \leqslant u \leqslant n} \left\{ Z'_{u,1} - \sum_{i=1}^{u-1} p_{i,2} \right\}$$
$$= \max_{1 \leqslant u \leqslant n} \left\{ H_1 + \sum_{i=1}^{u} p_{i,1} - \sum_{i=1}^{u-1} p_{i,2} \right\}$$

Since $H_1 = 0$; therefore,

$$\begin{aligned} H_2 &= \max_{1 \leqslant u \leqslant n} \left\{ \sum_{i=1}^{u} p_{i,1} - \sum_{i=1}^{u-1} p_{i,2} \right\} \\ &= \sum_{i=1}^{n} I_{i,2} = Z_{n,2} - \sum_{i=1}^{n} p_{i,2} \end{aligned} \tag{31}$$

$$\begin{aligned} H_3 &= \max_{1 \leqslant u \leqslant n} \left\{ Z'_{u,2} - \sum_{i=1}^{u-1} p_{i,3} \right\} \\ &= \max_{1 \leqslant u \leqslant n} \left\{ H_2 + \sum_{i=1}^{u} p_{i,2} - \sum_{i=1}^{u-1} p_{i,3} \right\} \\ &= H_2 + \max_{1 \leqslant u \leqslant n} \left\{ \sum_{i=1}^{u} p_{i,2} - \sum_{i=1}^{u-1} p_{i,3} \right\} \\ &= H_2 + p_{1,2} \end{aligned} \tag{32}$$

$$
\begin{aligned}
H_4 &= \max_{1 \leqslant u \leqslant n} \left\{ Z'_{u,3} - \sum_{i=1}^{u-1} p_{i,4} \right\} \\
&= \max_{1 \leqslant u \leqslant n} \left\{ H_3 + \sum_{i=1}^{u} p_{i,3} - \sum_{i=1}^{u-1} p_{i,4} \right\} \\
&= H_3 + \max_{1 \leqslant u \leqslant n} \left\{ \sum_{i=1}^{u} p_{i,3} - \sum_{i=1}^{u-1} p_{i,4} \right\} \\
&= H_3 + p_{1,3}
\end{aligned}
$$

From Equation (32);

$$
\begin{aligned}
H_4 &= H_2 + p_{1,2} + p_{1,3} \\
&= H_2 + \sum_{k=2}^{3} p_{1,k}
\end{aligned} \tag{33}
$$

Continuing in this way;

$$
H_m = H_2 + \sum_{k=2}^{m-1} p_{1,k} \tag{34}
$$

From Equations (32), (33) and (34);

$$
H_r = H_2 + \sum_{k=2}^{r-1} p_{1,k}, \; r = 3, \; 4, \ldots, m \tag{35}
$$

From Equation (34), $H_{\mathrm{m}}$ is minimum when idle time of $M_2$ and $\sum_{k=2}^{m-1} p_{1,k}$ are minimum. The following algorithm gives the sequence, which minimizes total elapsed time under no-idle constraint.

**ALGORITHM 4.3.**

**Step 1:** Obtain sequence $S$ by applying Johnson's two-machine algorithm on $M_1$ and $M_2$. In case of a tie for the first position of the sequence $S$, prefer that job for the first position for which sum of the processing times on $M_2$, $M_3, \ldots, M_{m-1}$ is minimum.

**Step 2:** If the sum of the processing times of the first job of sequence $S$ on $M_2$, $M_3, \ldots, M_{m-1}$ is minimum, then $S$ is an optimal sequence. Otherwise, go to Step 3.

**Step 3:** Obtain other sequence from $S$ by substituting that job in the first position whose sum of the processing times on $M_2$, $M_3, \ldots, M_{m-1}$ is less than the sum of the processing times of first job of $S$ and without disturbing the remaining order of $S$.

*Table 3.* Processing times of jobs

| Jobs | Machines | | | |
|---|---|---|---|---|
| | $M_1$ | $M_2$ | $M_3$ | $M_4$ |
| 1 | 8 | 2 | 8 | 11 |
| 2 | 10 | 3 | 7 | 10 |
| 3 | 5 | 4 | 5 | 12 |
| 4 | 4 | 3 | 6 | 10 |
| 5 | 7 | 5 | 7 | 9 |

**Step 4:** Repeat Step 3 till all such jobs have been placed in the first position. Let these sequences be $S_1$, $S_2$,…, $S_{r_3}$.

**Step 5:** Find out $H_m$ for the sequences $S$, $S_1$, $S_2$,…, $S_{r_3}$ using Equation (34). The sequence(s) having minimum $H_m$ is (are) the required sequence(s).

**Step 6:** For the sequence(s) obtained in Step 5; find out total elapsed time using Equation (10) and $H_2$, $H_3$,…, $H_{m-1}$ using Equations (31) and (35).

EXAMPLE 4.3. Consider the 5-job, 4-machine problem whose processing times are given as in Table 3. Applying Algorithm 4.3, Johnson's algorithm on $M_1$ and $M_2$ provide the sequence $S$ = 5-3-2-4-1. The sum of the processing times of job 5 on $M_2$ and $M_3$ is 12 units. Jobs 1, 2, 3 and 4 have sum of the processing times on $M_2$ and $M_3$ less than 12 units. Therefore, the other sequences are $S_1$ = 1-5-3-2-4; $S_2$ = 2-5-3-4-1; $S_3$ = 3-5-2-4-1 and $S_4$ = 4-5-3-2-1. $H_m$ for sequences $S$, $S_1$, $S_2$, $S_3$ and $S_4$ are 31 units, 30 units, 29 units, 28 units and 28 units respectively. $H_m$ is minimum for 3-5-2-4-1 and 4-5-3-2-1. Therefore, 3-5-2-4-1 and 4-5-3-2-1 are optimal sequences. The total elapsed time under no-idle constraint is 80 units. For sequence 3-5-2-4-1;$H_1$, $H_2$ and $H_3$ are 0 unit, 19 units and 23 units respectively. For sequence 4-5-3-2-1; $H_1$, $H_2$ and $H_3$ are 0 unit, 21 units and 24 units respectively.

**Case 4**: $p_{i,k} \geqslant p_{j,k+1}; \forall i, j, i \neq j, \quad k = 2, 3, \ldots, m-1.$

The times at which machine should be taken on rent to process jobs continuously without idle interval are :

$$H_1 = 0$$

$$H_2 = Z_{n,2} - \sum_{i=1}^{n} p_{i,2} \tag{36}$$

From Theorem 3.1,

$$
\begin{aligned}
H_3 &= \max_{1 \leqslant u \leqslant n} \left\{ Z'_{u,2} - \sum_{i=1}^{u-1} p_{i,3} \right\} \\
&= \max_{1 \leqslant u \leqslant n} \left\{ H_2 + \sum_{i=1}^{u} p_{i,2} - \sum_{i=1}^{u-1} p_{i,3} \right\} \\
&= H_2 + \max_{1 \leqslant u \leqslant n} \left\{ \sum_{i=1}^{u} p_{i,2} - \sum_{i=1}^{u-1} p_{i,3} \right\} \\
&= H_2 + \sum_{i=1}^{n} p_{i,2} - \sum_{i=1}^{n-1} p_{i,3} \\
&= H_2 + \left\{ \sum_{i=1}^{n} p_{i,2} - \sum_{i=1}^{n} p_{i,3} \right\} + p_{n,3} \qquad (37) \\
H_4 &= \max_{1 \leqslant u \leqslant n} \left\{ Z'_{u,3} - \sum_{i=1}^{u-1} p_{i,4} \right\} \\
&= H_3 + \max_{1 \leqslant u \leqslant n} \left\{ \sum_{i=1}^{u} p_{i,3} - \sum_{i=1}^{u-1} p_{i,4} \right\} \\
&= H_3 + \sum_{i=1}^{n} p_{i,3} - \sum_{i=1}^{n-1} p_{i,4}
\end{aligned}
$$

From Equation (37),

$$
\begin{aligned}
H_4 &= H_2 + \left\{ \sum_{i=1}^{n} p_{i,2} - \sum_{i=1}^{n} p_{i,3} \right\} + p_{n,3} + \sum_{i=1}^{n} p_{i,3} - \sum_{i=1}^{n} p_{i,4} + p_{n,4} \\
&= H_2 + \left\{ \sum_{i=1}^{n} p_{i,2} - \sum_{i=1}^{n} p_{i,4} \right\} + \sum_{k=3}^{4} p_{n,k}
\end{aligned} \qquad (38)
$$

Continuing in this way;

$$
H_m = H_2 + \left\{ \sum_{i=1}^{n} p_{i,2} - \sum_{i=1}^{n} p_{i,m} \right\} + \sum_{k=3}^{m} p_{n,k} \qquad (39)
$$

From Equations (37), (38) and (39);

$$
H_r = H_2 + \left\{ \sum_{i=1}^{n} p_{i,2} - \sum_{i=1}^{n} p_{i,r} \right\} + \sum_{k=3}^{r} p_{n,k} \qquad (40)
$$

From Equation (39), $H_m$ is minimum when idle time of $M_2$ and $\sum_{k=3}^{m} p_{n,k}$ are minimum.

*Table 4.* Processing times of jobs

| Jobs | Machines | | | |
|---|---|---|---|---|
| | $M_1$ | $M_2$ | $M_3$ | $M_4$ |
| 1 | 6 | 10 | 8 | 3 |
| 2 | 5 | 9 | 7 | 6 |
| 3 | 9 | 12 | 9 | 3 |
| 4 | 10 | 9 | 7 | 5 |
| 5 | 12 | 11 | 8 | 7 |

The following algorithm gives the sequence, which minimizes the total elapsed time under no-idle constraint.

**ALGORITHM 4.4.**

**Step 1:** Obtain sequence $S$ by applying Johnson's two-machine algorithm on machine $M_1$ and $M_2$. In case of a tie for the last position of the sequence $S$, prefer that job for the last position for which the sum of the processing times on $M_3$, $M_4, \ldots, M_m$ is minimum.

**Step 2:** If the sum of the processing times of the last job of $S$ on $M_3$, $M_4, \ldots, M_m$ is minimum, then $S$ is an optimal sequence. Otherwise, go to Step 3.

**Step 3:** Obtain other sequence from $S$ by substituting that job in the last position whose sum of processing times on $M_3$, $M_4, \ldots, M_m$ is less than the sum of the processing times of last job of $S$ and without disturbing the remaining order of $S$.

**Step 4:** Repeat Step 3 till all such jobs have been placed in the last position. Let these sequences by $S_1, S_2, \ldots, S_{r_4}$.

**Step 5:** Find out $H_m$ for the sequences $S$, $S_1, S_2, \ldots, S_{r_4}$ using Equation (39). The sequence(s) having minimum $H_m$ is (are) the required sequence(s).

**Step 6:** For the sequence(s) obtained in Step 5; find out total elapsed time using Equation (10) and $H_2, H_3, \ldots, H_{m-1}$ using Equations (36) and (40).

EXAMPLE 4.4. Consider a 5-job, 4-machine sequencing problem whose processing times are given as in Table IV. Applying Algorithm 4.4, Johnson's algorithm on $M_1$ and $M_2$ provides the sequence $S$ = 2-1-3-5-4. The sum of the processing times of job 4 on $M_3$ and $M_4$ is 12 units. Job 1 has the sum of the processing times on $M_3$ and $M_4$ less than 12 units. Therefore, the other sequence is $S_1$ = 2-3-5-4-1. $H_m$ for $S$ and $S_1$ are 44 units and 43 units respectively. $H_m$ is minimum for sequence 2-3-5-4-1. Therefore, 2-3-5-4-1 is the optimal sequence. The total elapsed time under no-idle constraint‘ is 67 units. For sequence 2-3-5-4-1; $H_1$, $H_2$ and $H_3$ are 0 unit, 5 units and 25 units respectively.

## 5. Discussion

We studied four n-job, m-machine flowshop problems under no-idle constraint when processing times of jobs on various machines follow certain conditions. No-idle constraint is important when machines are taken on rent. Under no-idle constraint, machines work continuously without idle interval, i.e., each machine is taken on rent for time equal to the sum of processing times of all jobs on it. Therefore, under no-idle constraint, total rental cost of machines is minimum. Here, we considered the problem with objective being minimization of total elapsed time under no-idle constraint. We introduced simple algorithm for each of these problems without using Branch and Bound technique. Many significant open problems exist, e.g., minimize mean flowtime under no-idle constraint, minimize tardiness under no-idle constraint etc.

## Acknowledgements

The authors are extremely grateful to the referees for their valuable comments and suggestions.

## References

1. Adiri, I. and Pohoryles, D. (1982), Flowshop/no idle or no wait scheduling to minimize the sum of completion times, *Naval Research Logistic Quarterly* 29, 495–504.
2. Bagga, P.C. and Ambika, B. (1997), Sequencing with restrictions in processing times, *Opsearch* 34(2), 116–127.
3. Bagga, P.C. and Chakravarti, N.K. (1968), Optimal m-stage production schedule, *CORS Journal* 6, 61–78.
4. Dudek, R.A. and Teuton, O.F. (1964), Development of m-stage decision rule for scheduling n jobs through m machines, *Operations Research* 12, 471–497.
5. Gupta, J.‘N.D. (1969), A general algorithm for the n × m flowshop scheduling problem, *The International Journal of Production Research* 7, 241–247.
6. Johnson, S.M. (1954), Optimal two and three stage production schedule with set up times included, *Naval Research Logistic Quarterly* 1, 61–68.
7. Narain, L. and Bagga, P.C. (2003), Minimize total elapsed time subject to zero total idle time of machines in n × 3 flowshop problem, *Indian Journal of Pure* & *Applied Mathematics* 34(2), 219–228.
8. Panwalker, S.S. and Khan, A.W. (1975), An improved branch and bound procedure for the n × m flowshop problems, *Naval Research Logistic Quarterly* 22, 787–790.
9. Smith, R.D. and Dudek, R.A. (1967), A general algorithm for solution of n jobs m machines sequencing problems of the flowshop, *Operations Research* 15, 71–82.
10. Szwarc, W. (1977), Special cases of the flowshop problem, *Naval Research Logistic Quarterly* 24, 483–492.