Indian J. pure appl. Math., 33(4) : 435-441, April 2002
© Printed in India.

BICRITERIA IN SPECIAL FLOWSHOP PROBLEMS

P. C. BAGGA* AND AMBIKA BHAMBANI**

**Moti Lal Nehru College(E), University of Delhi, India*
***Jesus and Mary College, University of Delhi, India*

(*Received* 24 *December* 1998; *Accepted* 12 *June* 2001)

This paper provides the procedure for obtaining sequence(s) in special flowshop problems which gives minimum possible makespan while minimizing total flowtime.

Key Words : Sequencing Theory; Flowshop Problems; Makespan; Total Flowtime; Bicriteria; Branch and Bound Method

1. INTRODUCTION

Bagga and Ambika[1] obtain a sequence which minimizes mean completion time of jobs for general flowshop problem when processing time of jobs on various machines follow certain conditions.

Consider the 4-jobs, 5-machines sequencing problem with processing times as in Table 1.1.

TABLE 1.1

Machines → Jobs ↓	A	B	C	D	E
1	10	1	8	6	2
2	7	3	3	8	3
3	2	8	9	7	1
4	9	4	8	5	6

Solving this problem by existing Branch and Bound technique, the optimal sequence 2-3-1-4, with total flowtime 156 units, is obtained. In this technique, if instead of terminating the procedure when the total flowtime of a complete sequence is less than/equal to value attached with all the unbranched vertices, the termination is done when the total flowtime of a complete sequence is strictly less than the value, then all the other optimal sequences (if any) can be obtained.

Using this concept, the problem gives three more sequences, viz., 2-3-4-1; 3-2-4-1; and 3-4-2-1; with same total flowtime i.e., 156 units.

Although all the four sequences give the same total flowtime value, instead of choosing a sequence at random, another criterion can be used to select one out of these sequences. One such criterion can be the least makespan time, among the sequences obtained. In this problem, the makespan of the sequences obtained viz., 2-3-1-4; 2-3-4-1; 3-2-4-1 and 3-4-2-1 are 54 units; 51 units; 47 units and 48 units respectively. The sequence 3-2-4-1 minimizes the makespan (47 units) subject to minimum total flowtime (156 units),

This paper presents the algorithm for finding such sequence(s) which satisfy this bicriteria, in special flowshop problems which the authors have discussed in paper[1]. Numerical example is given in each special case to demonstrate the algorithm.

2. MATHEMATICAL FORMULATION

Notations :

S = any sequence of n jobs

i = job to be performed i.e., $i = 1, 2, ..., n$

$Y(i, k)$ = the processing time of job i on machine k.

$Z(i, k)$ = the time when the ith job completes the processing on machine k.

$i = 1, 2, ..., n$; $k = 1, 2, ..., m$.

For an $n \times m$ flowshop problem the problem can be mathematically formulated as obtaining a sequence S^*, which satisfies the bicriteria

Minimize $Z(n, m)$

subject to $\sum_{i=1}^{n} Z(i, m)$ is minimum, for the sequence S^*.

Case (1) —

$$Y(i, k) \leq Y(j, k+1); k = 1, 2, ..., p-1$$

and

$$Y(i, k) \geq Y(j, k+1); k = p, p+1, ..., m-1;$$

$$\forall\ i\ \&\ j;\ i \neq j.$$

In this case,

$$\sum_{i=1}^{n} Z(i, m) = n \sum_{k=1}^{p-1} Y(1, k) + \sum_{i=1}^{n} (n+1-i)\ Y(i, p) + \sum_{k=p+1}^{m} \sum_{i=1}^{n} Y(i, k). ... \quad (1.1)$$

Algorithm 1

Step 1 : Arrange the jobs in the ascending order of their processing times on machine p and call the sequence(s) so obtained by S_l; $l = 1, 2, ...,$ i.e. $S_1, S_2, ...$ The first job of the sequences S_l; $l = 1, 2, ...$ i.e., $S_1, S_2, ...$ be called as J_l; $l = 1, 2, ...$ i.e., $J_1, J_2, ...$ respectively.

Step 2 : Check for each sequence S_l; $l = 1, 2, ...;$

whether $$\sum_{k=1}^{p-1} Y(J_l, k) \leq \sum_{k=1}^{p-1} Y(i, k);\ \forall\ i$$

(a) If yes, call such sequences so obtained as S_M; $l = 1, 2, \ldots$ Find total flowtime of any of S_{M_l}; $l = 1, 2, \ldots$; by formula (1.1). Go to Step 7.

(b) If not, proceed to Step 3.

Step 3 : Find all i's for which $\sum_{k=1}^{p-1} Y(i, k) < \sum_{k=1}^{p-1} Y(J_l, k)$, for each S_l.

Step 4 : Replace the job i's (one by one) in the first position of the sequence S_l, without disturbing the rest of the sequence, thus forming the new sequences.

Step 5 : Obtain the total flowtime for all the sequences, obtained in Step 1 and Step 4, by formula (1.1)

Step 6 : Comparing the total flowtime of these sequences; call the sequences which minimizes the total flowtime by S_{M_l}; $l = 1, 2, \ldots$; with its total flowtime value.

Step 7 : Comparing the makespan of the sequences S_{M_l}; $l = 1, 2, \ldots$; obtain the sequence(s) which gives the minimum makespan.

Step 8 : The sequence(s) so obtained in Step 7, is (are) the optimal sequence(s) satisfying the bicriteria with the makespan, as obtained in Step 7, and the total flowtime, as either obtained from Step 2 or from Step 6.

Example 1.1 — Consider 4-jobs, 5-machines sequencing problem with processing times as in Table I.2

TABLE I.2

Machines → Jobs ↓	A	B	C	D	E
1	5	7	8	6	2
2	3	7	9	7	1
3	6	5	7	4	5
4	5	7	8	5	4

Using the procedure of Algorithm 1 the sequences 3-1-4-2; 3-4-1-2; 2-3-4-1; 2-3-1-4; are generated. The optimal sequence is 2-3-4-1 with makespan as 50 units and total flowtime as 155 units.

Case 1 (*a*) — $p = 1$

$$Y(i, k) \geq Y(j, k+1); k = 1, 2, \ldots . m-1; \ \forall\ i \text{ and } j,\ i \neq j$$

In this case,

$$\sum_{i=1}^{n} Z(i, m) = \sum_{i=1}^{n} (n+1-i)\, Y(i, 1) + \sum_{k=2}^{m} \sum_{i=1}^{n} Y(i, k) \qquad \ldots (1.1\ (a))$$

Algorithm 1 (*a*)

Step 1 : Arrange the jobs in the ascending order of their processing times on machine 1. Call the sequences, so obtained as $S_l : l = 1, 2, \ldots$ i.e., $S_1, S_2, \ldots$

Obtain the total flowtime of any of S_l; $l = 1, 2, ...$; by formula (1.1 (a)).

Step 2 : Enumerate the makespan for all these sequences S_l; $l = 1, 2, ...$; and comparing, obtain the sequence(s) which gives the minimum makespan.

Step 3 : The sequences, so obtained in Step 2, is (are) the optimal sequence(s) satisfying the bicriteria, with the makespan, as obtained in Step 2, and the total flowtime, as obtained in Step 1.

Example 1.1 (*a*) — Consider 4-jobs, 5-machines sequencing problem with processing times as in Table 1.2 (*a*)

TABLE I.2 (*a*)

Machines → Jobs ↓	A	B	C	D	E
1	12	10	8	5	1
2	15	9	8	5	2
3	15	10	6	4	3
4	11	9	7	6	4

Using Algorithm 1(*a*), the sequences 4-1-2-3 and 4-1-3-2 are generated. The optimal sequence is 4-1-2-3 with makespan as 76 units and total flowtime as 222 units.

Case (2) :

$$Y(i,k) \leq Y(j,k+1);\ \forall\ i\ \&\ j;\ i \neq j;\ k = 1, 2 ..., p-1$$

In this case,

$$Z(n,p) = \sum_{k=1}^{p-1} Y(1,k) + \sum_{i=1}^{n} Y(i,p) \qquad \ldots (2.1)$$

Algorithm 2

Step 1 : Apply the usual Branch and Bound technique to minimize the total flowtime, for the reduced $n \times (m - \overline{p-1})$ problem, by using the following formula, for obtaining the lower bound for the partial schedule J_r f r jobs :-

$$LB[J_r] = \sum_{i \in Jr} Z(i,m) + \max_{p \leq t \leq m}$$

$$\left[(n-r)\, Z(J_r, t) + \sum_{j=r}^{n-1} (n-j)\ Y(i_{s_{j+1}}, t) + \sum_{i \in J_r'} \sum_{k=t+1}^{m} Y(i,k) \right] \qquad \ldots (2.2)$$

$$= \text{Max } [G_1, G_2, ..., G_l] \text{ say; where } l = m - \overline{p-1}$$

where J_r' = complement of J_r in S.

$Z(J_r, t)$ = completion time of the last job of partial schedule J_r on machine t.

$\langle i_{s_{r+1}}, i_{s_{r+2}}, ..., i_{s_n} \rangle$ = partial sequence of jobs in J_r' in the ascending order of their processing times on machine t.

The procedure terminates only when total flowtime of sequence(s) is strictly less than the value attached to all the unbranched vertices.

Step 2 : The sequence(s) obtained in Step 1, also minimizes the total flowtime for the original $n \times m$ flowhop problem with the same total flowtime.

Step 3 : Enumerate the makespan, for $n \times m$ problem, of the sequences(s) obtained in Step 2, and comparing, obtain the sequence(s) which minimizes the makespan.

Step 4 : The sequences(s) obtained in Step 3, is (are) the optimal sequence(s) satisfying the bicriteria, with makespan, as obtained in Step 3, and total flowtime, as obtained in Step 2, for $n \times m$ flowshop problem.

Example 2.1 — Consider 4-jobs, 5-machines sequencing problem with processing times as in Table II.1.

TABLE II.1

Machines → Jobs ↓	A	B	C	D	E
1	2	4	8	1	11
2	3	4	9	10	3
3	4	5	7	2	4
4	1	7	8	5	6

Applying the Algorithm 2, the sequences generated are 1-3-4-2 and 1-4-3-2 with total flowtime as 147 units, for 4 × 3 flowshop problem (on machines C, D and E) and for 4 × 5 flowshop problem. The optimal sequences are 1-3-4-2 and 1-4-3-2 with makespan as 51 units and total flowtime as 147 units.

Case 3 :

$$Y(i,k) \geq Y(j,k+1);\ \forall\ i\ \&\ j\ i \neq j; p \leq k \leq m-1$$

In this case,

$$\sum_{i=1}^{n} Z(i,m) = \sum_{i=1}^{n} Z(i,p) + \sum_{k=p+1}^{m} \sum_{i=1}^{n} Y(i,k) \qquad \ldots (3.1)$$

Algorithm 3 :

Step 1 : Apply the usual Branch and Bound technique to minimize the total flowtime, for the reduced $n \times p$ problem (on machines 1, 2, ..., p), by using the following formula for obtaining the lower bound for the partial schedule J_r of r jobs:

$$LB[J_r] = \sum_{i \in J_r} Z(i,p) + \underset{1 \leq t \leq p}{Max}$$

$$\left[(n-r)\, Z(J_r, t) + \sum_{j=r}^{n-1} (n-j)\, Y(i_{S_{j+1}}, t) + \sum_{i \in J_r'} \sum_{k=t+1}^{p} Y(i, k) \right] \quad \ldots (3.2)$$

where J_r' = complement of J_r in S.

$Z(J_r, t)$ = completion time of the last job of the partial schedule J_r on machine t.

$\left\langle i_{S_{r+2}}, i_{S_{r+1}}, \ldots, i_{S_n} \right\rangle$ = partial sequence of jobs in J_r' in the ascending order of their processing times on machine t.

The procedure terminates only when the total flowtime of sequence(s) is strictly less than the value attached to all the unbranched vertices.

Step 2 : The sequence(s) obtained in Step 1, also minimizes the total flowtime for the original $n \times m$ flowshop problem. Obtain the total flowtime of any of the sequence(s), for the original $n \times m$ problem, by formula (3.1).

Step 3 : Enumerate the makespan for $n \times m$ problem, of the sequence(s) obtained in Step 2, and comparing, obtain the sequence(s) which minimizes the makespan.

Step 4 : The sequence(s) obtained in Step 3, is (are) the optimal sequence(s) satisfying the bicriteria, with makespan, as obtained in Step 3, and total flowtime, as obtained in Step 2, for $n \times m$ flowshop problem.

Example 3.1 — Consider 4-jobs, 5-machines sequencing problem with processing times as in Table III.1

TABLE III.1

Machines → Jobs ↓	A	B	C	D	E
1	10	1	8	6	2
2	2	8	9	7	1
3	7	3	7	8	3
4	9	4	8	5	6

Applying Algorithm 3, the sequences generated are 2-3-1-4 and 2-3-4-1 with total flowtime as 121 units, for 4 × 3 flowshop problem (on machines *A*, *B* and *C*) and with total flowtime as 159 units for the original 4 × 5 flowshop problem. The optimal sequence is 2-3-4-1 with makespan as 50 units and total flowtime as 159 units.

Case 4 :

$$Y(i, k) \geq Y(j, k+1);\ \forall\ i\ \&\ j;\ i \neq j;\ k = 1, 2, \ldots, p-1$$

In this case,

$$Z(n, p) = \sum_{i=1}^{n} Y(i, 1) + \sum_{k=2}^{p} Y(n, k). \quad \ldots (4.1)$$

The Algorithm in this case, for obtaining the optimal sequence(s) satisfying the bicriteria is same as Algorithm 2.

Example 4.1 — Consider 4-jobs, 5-machines sequencing problem with processing times as in Table IV.1.

TABLE IV.1

Machines → Jobs ↓	A	B	C	D	E
1	8	6	2	10	1
2	9	7	1	2	8
3	7	4	3	7	3
4	8	5	4	9	4

The sequence generated is 3-1-2-4 for the reduced 4 × 3 problem (on machines *C*, *D* and *E*) and for the original 4 × 5 flowshop problem, with total flowtime as 155 units. The optimal sequence is 3-1-2-4 with makespan as 54 units and total flowtime as 155 units.

REFERENCE

1. P. C. Bagga and Ambika Bhambani, *Optimization*, **40** (1997), 1, 85-100.