MINIMIZING WAITING TIME OF JOBS FOR SCHEDULING n JOBS THROUGH m MACHINES

P. C. BAGGA

University of Delhi, Delhi

AND

N. K. CHAKRAVARTI*

Cabinet Sectt., New Delhi

(*Received* 28 *September* 1970 ; *revised* 13 *October* 1971)

Studies are done in this paper to find an optimal sequence of n jobs, which when processed through m machines minimise the total waiting time of jobs.

Situations may arise in practice where jobs as soon as they come out from the machines are required for immediate use and the waiting of jobs result in significant costs. For example, consider that certain refuellers are under repairs. If these are the only ones, flying schedule of the aircraft will be disturbed until at least one can be got ready quickly. Thus it is clearly seen that we are not interested in the minimum elapsed time, but in getting the repaired refuellers as early as possible.

This paper deals with the n job- m machine flow shop sequencing problem in which the jobs are assumed to be equally costly. The method is illustrated by means of a numerical example.

MATHEMATICAL FORMULATION

Consider a set of jobs 1, 2, 3,........ n which require processing over each of the m machines A, B, C, ... L, M in this order of machines. Let

S = a schedule of the n jobs in the order, 1, 2, ..n

X_k = the processing time of job k on machine X, $X = A, B, C, \ldots\ldots, M$ and $k = 1, 2, 3, \ldots n$.

$z(k, X)$ = the completion time of job k on machine X.

S' = a schedule derived from S by interchanging the position of jobs j and $j+1$ in the schedule S.

$z'(k, X)$ = the completion time of job k on machine X when the jobs are scheduled according to the schedule S.

We have,

$$z(k, X) = \max \left\{ \begin{array}{l}\text{completion time of preceding}\\ \text{job on machine } X\end{array}, \begin{array}{l}\text{completion time of job } k \text{ on}\\ \text{machine } X-1\end{array} \right\} + X_k \text{ for all } k. \quad (1)$$

If the jobs are scheduled according to schedule S, the total waiting time of the jobs is

$$T = \sum_{k=1}^{n} z(k, M) \quad (2)$$

On the other hand the total waiting time of the jobs according to S' is

$$T' = \sum_{k=1}^{n} z'(k, M) \quad (3)$$

Thus sequence S will be preferable to S' if

$$T < T' \quad (4)$$

*Now at National Sample Survey, R.K. Puram.

Since S and S' do not differ before the jth position is processed, the above inequality reduces to

$$z(j, M) + z(j+1, M) \leqslant z'(j+1, M) + z'(j, M) \tag{5}$$

and

$$\sum_{k=j+2}^{n} z(k, M) \leqslant \sum_{k=j+2}^{n} z'(k, M) \tag{6}$$

The inequality $z(n, M) \leqslant z'(n, M)$ holds if

$$z(n, L) \leqslant z'(n, L) \tag{7}$$

and

$$z(n-1, M) \leqslant z'(n-1, M) \tag{8}$$

Now

$$z(n-1, M) \leqslant z'(n-1, M)$$

if

$$z'(n-2, M) \leqslant z'(n-2, M) \tag{9}$$

$$z(n-1, L) \leqslant z'(n-1, L) \tag{10}$$

Continuing in the similar way, we have

$$z(k, M) \leqslant z'(k, M) \qquad (k = j+2,\ j+3, \ldots\ldots n)$$

if

$$z(j+1, M) \leqslant z'(j, M) \tag{11}$$

and

$$z(k, L) \leqslant z'(k, L) \qquad (k = j+2, j+3, \ldots, n) \tag{12}$$

On the same lines

$$z(k, L) \leqslant z'(k, L)$$

if

$$z(j+1, L) \leqslant z'(j, L) \tag{13}$$

and

$$z(k, K) \leqslant z'(k, K) \qquad (k = j+2, \ldots\ldots n) \tag{14}$$

Proceeding in the same manner, we have

$$\Sigma\, z(k, M) \leqslant \Sigma\, z'(k, M) \qquad (k = j+2, j+3, \ldots\ldots n)$$

if

$$z(j+1, X) \leqslant z'(j, X) \qquad (X = B, C, \ldots K\, L\, M) \tag{15}$$

Hence the condition for sequence S to be preferable to S' is

$$\left.\begin{aligned} & z(j, M) + z(j+1, M) \leqslant z'(j+1, M) + z'(j, M) \\ \text{and}\quad & z(j+1, X) \leqslant z'(j, X) \qquad (X = B, C, \ldots\ldots L, M) \end{aligned}\right\} \tag{16}$$

The sequence S can be written as $\{J_{j-1}\, j\, (j+1)\, \pi\}$ where J_{j-1} denote the presequence consisting of $j-1$ jobs. Condition (16) ensures that sequence $\{J_{j-1}\, j\, (j+1)\, \pi\}$ is preferable to $J_{j-1}\, (j+1)\, j\pi\}$ where π denote any permutation of jobs from $j+2$ to n. These conditions say nothing about the preferability or otherwise of $J_{j-1}\, j\, (j+1)\, \pi$ over $J_{j-1}\, (j+1)\, \pi' j \pi''$ where π' and π'' are subsets of π such that $\pi' \cup \pi'' = \pi$ and $\pi' \cap \pi'' = 0$. Taking now the sequences $J_{j-1}\, j\, (j+1)\, \pi$ and $J_{j-1}\, (j+1)\, \pi' j\, \pi''$ and applying the above results and using equation (19) of Smith-Dudek[1] it can be seen the corresponding conditions for the sequence $J_{j-1}\, j (j+1)\, \pi$ to be preferable to $J_{j-1}\, (j+1)\, \pi' j \pi''$ are

$$\left.\begin{aligned} & z(j, M) + z(j+1, M) \leqslant 2z'(j+1, M) + \min M_k, \\ & z(j+1, X) \leqslant z'(j+1, X) + \min X_k \\ & \qquad\qquad \begin{pmatrix} k = j, j+2, \ldots n \\ X = B, C, \ldots KLM \end{pmatrix} \end{aligned}\right\} \tag{17}$$

As (16) is always satisfied when (17) is satisfied. This is due to the fact that the right hand side of (16) is not less than (17). The decision rule for determining the optimal sequence can be stated as follows:

Job j dominates $j+1$ if

$$\left.\begin{array}{ll} & z(j, M) + z(j+1, M) \lesssim zz'(j+1, M) + \min M_k \\ \text{and} & z(j+1, X) \lesssim z'(j+1, X) + \min X_k \\ & \left(\begin{array}{l} k = j, j+2, \ldots n \\ X = B, C, \ldots KLM \end{array}\right) \end{array}\right\} \quad (18)$$

SEQUENCE DOMINANCE CHECK

If ρ and ρ' be two different permutations of the same set of jobs, then the partial sequence ρ dominates ρ' if

$$\sum_{k \epsilon \rho} z(k, M) \lesssim \sum_{k \epsilon \rho'} z'(k, M)$$

and

$$t(\rho, X) \lesssim t'(\rho', X)$$

where $t(\rho, X)$ and $t'(\rho', X)$ denote the completion time of last job of sequence ρ and ρ' respectively on machine X. ($X = B, C, \ldots K, L, M$). The proof for this can be shown on the above lines.

Example

Let us consider the following 5 job - 3 machine sequencing problem. The processing times of the jobs on the machines are given in Table 1.

TABLE 1

PROCESSING TIMES OF 5 JOBS ON 3 MACHINES

Jobs	Machines		
(k)	A	B	C
1	123	300	100
2	57	156	200
3	198	201	211
4	154	162	170
5	142	179	211

The problem is to find a sequence of the jobs which minimise the total waiting time.

Solution

The corresponding conditions for the job j to dominate job $j+1$ become

$$z(j, C) + z(j+1, C) \lesssim 2z'(j+1, C) + \min C_k$$

$$z(j+1, C) \lesssim z'(j+1, C) + \min C_k$$

$$z(j+1, B) \lesssim z'(j+1, B) + \min B_k$$

$$(k = j, j+2, j+3, \ldots n)$$

Considering job 2 at position no. 1 and job 1 at position no. 2, we have the following sequences for comparison

$$S = 2 \quad 1$$

$$S' = 1$$

Calculating the values of $z(2, C)$, $z(1, C)$, $z'(1, C)$ etc. and substituting these values the corresponding condition for job 2 to precede job 1 becomes

$$413 + 613 \leqslant 2 \times 523 + 170$$

$$613 \leqslant 523 + 170$$

$$513 \leqslant 423 + 156$$

As all the conditions are satisfied, job 2 dominates job 1. Similarly replacing job 1 by 3, 4, 5, we see that job 2 dominates all of them. Thus we schedule job 2 in the first position.

Having determined the job for the first position, we now examine job 4 for the second position. Placing job 1 in the third position and comparing the sequences $S = 2 \quad 4 \quad 1$ and $S' = 2 \quad 1$, the conditions for precedence become

$$583 + 775 \leqslant 2 \times 613 + 170$$

$$775 \leqslant 613 + 170$$

$$675 \leqslant 513 + 162$$

As all the conditions are satisfied, job 4 dominates job 1. Replacing job 1 by job 3, we get the corresponding conditions as

$$583 + 821 \leqslant 2 \times 667 + 100$$

$$821 \leqslant 667 + 100$$

$$610 \leqslant 456 + 162$$

Since two of these conditions are not satisfied, job 4 cannot dominate job 3. Similarly, it can be shown that job 4 does not dominate job 5. This shows that the second position can be filled by any of the remaining jobs 3, 4 and 5. Thus we have three feasible sequences :

$$S_1 = 2 \quad 3$$

$$S_2 = 2 \quad 4$$

$$S_3 = 2 \quad 5$$

Taking all these sequences one by one, it can be seen that none of the remaining jobs in all these sequences are dominated. As any of the remaining jobs can assume the third position, we get the following nine feasible sequences :

$$S_1 = 2 \quad 3 \quad 1$$

$$S_2 = 2 \quad 3 \quad 4$$

$$S_3 = 2 \quad 3 \quad 5$$

$$S_4 = 2 \quad 4 \quad 1$$

$$S_5 = 2 \quad 4 \quad 3$$

$$S_6 = 2 \quad 4 \quad 5$$

$$S_7 = 2 \quad 5 \quad 1$$

$$S_8 = 2 \quad 5 \quad 3$$

$$S_9 = 2 \quad 5 \quad 4$$

Applying sequence dominance check to the sequence pairs { 2 4 3, 2 3 4} {2 4 5, 2 5 4}, {2 3 5, 2 5 3} it can be shown that the sequences 2 3 4, 2 5 4 and 2 3 5 are dominated. Thus we have the following feasible sequences

$$S_1 = 2 \quad 4 \quad 1$$

$$S_2 = 2 \quad 4 \quad 3$$

$$S_3 = 2 \quad 4 \quad 5$$

$$S_4 = 2 \quad 3 \quad 1$$

$$S_5 = 2 \quad 5 \quad 1$$

$$S_6 = 2 \quad 5 \quad 3$$

Determining now the last two positions, we have finally the following feasible sequences

$$S_1 = 2 \quad 4 \quad 1 \quad 5 \quad 3$$

$$S_2 = 2 \quad 4 \quad 3 \quad 5 \quad 1$$

$$S_3 = 2 \quad 4 \quad 5 \quad 3 \quad 1$$

$$S_4 = 2 \quad 3 \quad 1 \quad 4 \quad 5$$

$$S_5 = 2 \quad 5 \quad 1 \quad 4 \quad 3$$

$$S_6 = 2 \quad 5 \quad 3 \quad 1 \quad 4$$

All these sequences are enumerated and the total waiting time of the jobs is determined. The sequence 2 4 5 3 1 has the minimum waiting time viz., 3 9 5 0 hrs.

VERIFICATION AND DISCUSSION

Thus we have the solution for the *m* machines flowshop scheduling problem. It can be easily seen that conditions for total waiting time of the jobs as criterion contain conditions for the criterion of minimum total elapsed time.

TABLE 2

NUMBER OF FEASIBLE SEQUENCES AND THE MEAN COMPUTING TIME IN GETTING OPTIMAL SEQUENCES

No. of machines	No of jobs	No. of problems	Average feasible sequences	Average computing time (min)
3	3	15	3	0·21
	4	20	5	0·51
	5	20	10	0·91
	6	15	20	2·272
	7	10	25	5·275
5	3	15	4	0·125
	4	15	7	0·418
	5	10	15	2·345
	6	10	25	14·672

The efficiency of the algorithm was investigated by working out several problems for three machines and five machines. The job range varies from three to seven. The processing times of the jobs on various machines were taken at random. A honeywell computor was used for the running of the programme. The preceding table shows the number of feasible sequences obtained and the mean computing time in getting the optimal sequence.

REFERENCE

SMITH, R. A. & DUDEK, R. A., *Opns. Res.*, 15 (1967), 71.