12th GLOBAL CONGRESS ON MANUFACTURING AND MANAGEMENT, GCMM 2014

Optimization of Makespan in Job and Machine Priority Environment

Baskar A[1, a*] and Anthony Xavior M [2, b]

[1]Professor, Apollo Engineering College, Chennai, India
[2] Professor, SMBS, VIT University, Vellore, India

Abstract

This paper analyses a specific case of permutation flow shop scheduling problem with job and machine priority. Sometimes, a particular job has to precede or succeed another job; or a set of jobs are to be together for a specific reason. Within the set of jobs also, there may be a condition that a particular job has to precede or succeed another job. In such case, the scheduling is done in two or more phases to optimize the makespan. In many occasions, the percentage utilization of a particular machine has to be increased due to several reasons. The machine may be costlier, rented, and highly precise or needs to be operated continuously. In the context of the permutation flow shop scheduling, the percentage utilization is calculated in terms of machining time compared with the makespan. For a fixed number of jobs, the total machining time for each machine is also fixed and reduction in the makespan only can increase the percentage utilization. On the other hand, splitting the machining times and regrouping may also help for achieving the objective. The problem is demonstrated with numerical examples. For trial purpose, the machining time of all the jobs that come before and after the priority machine are split (25% and 50%) individually and also in toto, and combined with the critical machine. The number of jobs and machines is kept the same. Consequently, the makespan and percentage utilization have been computed and compared with the original case.

Keywords: Heuristics; Permutation Flow Shop Scheduling; Makespan; Priority Environment

***corresponding author: Tel:+91-9444107790
Email: a.baaskar@gmail.com

1. Introduction

A general flow shop scheduling problem (FSP) with m – machine permutation is one of the most interesting areas of research for more than six decades. In this problem, each job i, i=1,2,….,n needs to be processed on each machine j, j=1,2…..,m, in that order during an uninterrupted processing time $t_{ij} \geq 0$. In such problems, minimizing the makespan (total completion time) is considered as one of the most important performance measures that has to be optimized. Some common performance evaluation objective functions of a PFSP are:
• Make Span – total time to completely process all jobs (Most Common and most popular)
• Average Time of jobs in a flow shop
• Lateness
• Average Number of jobs in a flow shop
• Utilization of machines
• Utilization of workers

Our objective is to find a processing order of the n jobs, the same for each machine since a permutation flow shop, such that the makespan is minimized. That is, all the n jobs are finished as soon as possible. For n number of jobs, then, n! **s**equences are possible. So many assumptions have been made while finding the sequence like: all jobs are available for processing at time zero; at any time, each machine can process at most one job and each job can be processed on at most one machine; the capacity of the queue for each machine is unlimited.

The processing time for computing the exact solution grows exponentially with an increase in the problem size. But, for a problem with 2 machines and n jobs, Johnson [1] had developed a polynomial algorithm to get an optimal sequence (more than one optimal solution may be available for the same problem), that is, in a definite time, one can get an optimal solution.

The problem is NP-complete and many heuristics have been proposed over the years to obtain approximate solutions for the flow shop problem. Palmer's [2] slope Index (SI) method and Gupta's [3] functional algorithm are the ones proposed in earlier days. Both the CDS and RA (rapid access procedure) heuristics proposed by Campbell et al. [4] and Dannenbring [5] respectively are based on Johnson"s algorithm for the two machine problem. The NEH algorithm proposed by Nawaz et al. [6] has been widely accepted as one of the best polynomial heuristics in practice.

However, the case is entirely different in real situations making the problem further complex. So many constraints are to be considered in real shop floors, which are unlimited. The authors have already analyzed a few cases like, job idle time in between machines [7] and known breakdown times [8]. Priority assignments have been analyzed by researchers like Etemdi et al [9] and Chang et al [10]. Chandramouli [11] proposed a Heuristic approach for n-job, 3-machine flow-shop scheduling problem involving transportation time, breakdown time and weights of jobs. Pandian and Rajendran [12] improved and further simplified the procedure.

For computing the makespan and the corresponding sequence, codes are generated in MATLAB 2008a and run in an i5 PC with 4 GB RAM.

2. Job Priority Environment

For the purpose of analysis, one permutation flow shop scheduling problem (PFSP) with 5 machines and 5 jobs is considered as shown in Table 1. For computing the makespans and the corresponding sequences, the well known NEH algorithm has been used.

In a general PFSP, all jobs are given the same preference and weight for processing. The scheduling is not influenced by any constraint. However, the constraints are unlimited in a real shop floor. For study purpose, only one constraint is considered in our paper. That is, a group of jobs are to be together while processing. For example, let us consider the job numbers 1, 3 and 4 are to be together during processing, always. The order of processing is not constrained here. But, once the processing of any one of the jobs from this group is started, other members should follow immediately. The approaches may be many to find a sequence with optimal / near optimal makespan. Only two cases are analyzed here.

Table 1, Sample 5 M/C, 5 Jobs PFSP

Machine	Jobs Processing Times				
	1	2	3	4	5
A	7	6	4	9	5
B	5	4	6	3	7
C	4	5	3	6	6
D	8	4	2	5	8
E	6	7	5	8	9

Case 1

The procedure:
1. The sub-sequence to be computed first.
2. From the optimal sub-sequence, only the first job is taken, considered with remaining jobs for computing the other partial sequence.
3. Both the sequences are combined to obtain the optimal sequence.
4. From the sub-sequence, only the last job is taken, considered with remaining jobs for computing the other partial sequence.
5. Both the sequences are combined to obtain the optimal sequence.
6. Similarly, other jobs from the initial partial sequence are taken one by one, combined with other jobs, for computing the final partial sequence. Both the partial sequences are combined properly to obtain the optimal sequence.
7. Finally, the best sequence and the corresponding makespan are selected from the available results.

When this procedure is used for the problem:
(i)   Among the jobs 1, 3 and 4, the sub-optimal sequence is 3-1-4 (with a makespan of 42 units).
(ii)   Job number 3 is taken and the set of jobs to be considered is: 2, 3 and 5. The sequence is once again is computed as 3-2-5 (with a makespan of 45 units).
(iii)   Combining both the sequences, the optimal sequence can be 3-1-4-2-5 with a makespan of 61 time units.
(iv)   For the second option, Job number 4 is taken and the set of jobs to be considered is: 2, 4 and 5.The partial sequence is: 2-5-4 (with a makespan of 49 units).
(v)   Combining both the sequences, the optimal sequence can be 2-5-3-1-4 with a makespan of 60 time units.
(vi)   Similarly, Job number 1 is taken and the set of jobs to be considered is: 1 2 5. The partial sequence is: 2-5-1 (with a makespan of 47 units). Both the sequences are combined to obtain the optimal sequence. The optimal sequence can be 2-5-3-1-4 with a makespan of 60 time units.
(vii)   Finally, out of the available sequences, the best one is selected with the minimum makespan. That is, 2-5-3-1-4 with a makespan of 60 time units is the required sequence for the problem with the constraint.

Case 2

The Procedure:
         In this approach, initially, without considering any constraint, using the popular NEH heuristic algorithm, the solution sequence and the corresponding makespan have to be computed. Afterwards, depends on the resultant sequence, the options shall be tried to arrive the final optimal sequence considering the constraint.

For the given problem, without considering the constraint, the sequence and the corresponding makespan have been

computed as **3-2-5-1-4** and 59 time units respectively.

Now, the constraint will be considered, that is, the jobs 1, 3 and 4 are to be together.

This can be done in many ways.

    (i)   Since the jobs 1, 3 and 4 are to be together, jobs 1 and 4 are moved ahead resulting in the sequence, 3-1-4-2-5 with a makespan of 61 time units.

    (ii) Job 3 is moved behind to obtain 2-5-3-1-4. The makespan is 60 time units.

    (iii) In the third option, jobs 3, 1 and 4 are inserted in between to get 2-3-1-4-5 with a makespan of 61 time units.

    (iv) Finally, the sequence with the minimum makespan is selected as the final schedule. In this case, 2-5-3-1-4 with a makespan of 60 time units.

3. Machine Priority Environment

The same PFSP as shown in Table 2 is considered and the condition is that, the critical machine is C, whose percentage utilization has to be improved. Machine C has a total machining time of 4+5+3+6+6=24 time units.

Table 2, Sample 5 M/C, 5 Jobs PFSP

Machine	Jobs Processing Times				
	1	2	3	4	5
A	7	6	4	9	5
B	5	4	6	3	7
C	*4*	*5*	*3*	*6*	*6*
D	8	4	2	5	8
E	6	7	5	8	9

Using the popular NEH heuristic algorithm, the solution sequence and the corresponding makespan have been computed as 3-2-5-1-4 and 59 time units respectively.

The percentage utilization of machine C= (24/59)*100=40.68.

Analysis (i)

Without altering other processing times, 25% on machining times of machine B are added to the machine C. The new problem arrived is presented in Table 3.

Table 3, New 5 M/C, 5 Jobs PFSP

Machine	Jobs Processing Times				
	1	2	3	4	5
A	7	6	4	9	5
B	3.75	3	4.5	2.25	5.25
C	*5.25*	*6*	*4.5*	*6.75*	*7.75*
D	8	4	2	5	8
E	6	7	5	8	9

The solution sequence and the corresponding makespan have been computed as 3-2-5-1-4 and 59 time units respectively.

Machine C has a machining time of 30.25 time units.

The percentage utilization of machine C= (30.25/59)*100=51.27.

Now, 50% on machining times of machine B are added to machine C as shown in Table 4.

Table 4, Modified 5 M/C, 5 Jobs PFSP

Machine	Jobs Processing Times				
	1	2	3	4	5
A	7	6	4	9	5
B	2.5	2	3	1.5	3.5
C	*6.5*	*7*	*6*	*7.5*	*9.5*
D	8	4	2	5	8
E	6	7	5	8	9

The solution sequence and the corresponding makespan have been computed as 3-2-5-1-4 and 60.5 time units respectively.
Machine C has a machining time of 36.5 time units.
The percentage utilization of machine C= (36.5/60.5)*100=60.33.
Finally, all the machining times of machine B are combined with the machine C (Table 5).

Table 5, Machine B completely eliminated

Machine	Jobs Processing Times				
	1	2	3	4	5
A	7	6	4	9	5
B	0	0	0	0	0
C	*9*	*9*	*9*	*9*	*13*
D	8	4	2	5	8
E	6	7	5	8	9

Repeating the computation, the solution sequence and the corresponding makespan have been computed as 5-1-4-2-3 and 61 time units respectively.
Machine C has a machining time of 49 time units.
The percentage utilization of machine C= (49/61)*100=80.33.

Analysis (ii)

In this phase, the processing times pertaining to the machine D are split and appended to machine C, initially 25% and then 50%. The resultant problems are shown in Tables 6 and 7.

Table 6, Processing Times of Machine D Split

Machine	Jobs Processing Times				
	1	2	3	4	5
A	7	6	4	9	5
B	5	4	6	3	7
C	*6*	*6*	*3.5*	*7.25*	*8*
D	6	3	1.5	3.75	6
E	6	7	5	8	9

Using the NEH heuristic algorithm, as earlier, the solution sequence and the corresponding makespan have been computed as 3-2-5-1-4 and 59 time units respectively.
Machine C has a machining time of 30.75 time units.
The percentage utilization of machine C= (30.75/59)*100=52.12.

Table 7, 50% of Processing Times of Machine D added to C

Machine	Jobs Processing Times				
	1	2	3	4	5
A	7	6	4	9	5
B	5	4	6	3	7
C	*8*	*7*	*4*	*8.5*	*10*
D	4	2	1	2.5	4
E	6	7	5	8	9

For the 50% modified problem, the solution sequence and the corresponding makespan have been computed as 3-2-5-1-4 and 59 time units respectively.
Machine C has a machining time 37.5 time units.
The percentage utilization of machine C= (37.5/59)*100=.63.60.

If all the machining times of D are combined to the machine C as shown in Table 8, then, the solution sequence and the corresponding makespan have been computed as 3-2-5-4-1 and 67 time units respectively.
Machine C has a machining time 51 time units.

Table 8, Machine D completely eliminated

Machine	Jobs Processing Times				
	1	2	3	4	5
A	7	6	4	9	5
B	5	4	6	3	7
C	*12*	*9*	*5*	*11*	*14*
D	0	0	0	0	0
E	6	7	5	8	9

The percentage utilization of machine C= (51/67)*100=.76.12.
The results are summarized in Table 9.

Table 9, Summary of Results

Condition	Percentage Split	Makespan	Percentage Utilization
B => C	25	59	51.27
	50	60.5	60.33
	100	61	80.33
D => C	25	59	52.12
	50	59	63.60
	100	67	76.12

4. Discussion and Conclusion

This paper analyzed two specific constraints with numeric examples, one with job priority and the other one with machine priority. In the case of job priority environment, the constraint of togetherness has been considered and analyzed. Three jobs have been assumed randomly for this purpose. Two different procedures are used to obtain the minimum possible makespan. When the first procedure is used, the final solution is obtained from three sequences with makespans 61, 60 and 60 time units. Similarly, the other procedure also yields three sequences

with makespans 61, 60 and 61 time units. However, in both the procedures, the final solution is the same. That is, the sequence 2-5-3-1-4 with a makespan of 60 time units. It has been observed that, specifying one heuristic procedure is extremely difficult to arrive a possible solution.

The other machine priority constraint problem deals with increasing the percentage utilization of a particular machine, termed as a critical machine. This has been tried by increasing the processing times of different jobs in that machine. Different situations have been analyzed with 25%, 50% and finally 100% reallocation of processing times of adjacent machines to the critical machine. The percentage utilization naturally increases with increased machining times at the cost of makespan. However, the percentage increase of makespan is very less when compared to the increase in percentage utilization (Table 9). Hence, it is recommended that, wherever possible, the processing times of the neighbouring machines are to be split and added with that of critical machine. The reallocation of processing times need not be same for all the jobs.

References

[1] Johnson SM. Optimal two and three machine production scheduling with set up times included. *Naval Research* 1954; Log.1, No.1.

[2] Palmer DS. Sequencing Jobs Through A Multi-Stage Process in the Minimum Total Time-A Quick Method of Obtaining A Near Optimum. *Operations Research 1965*; 16:101-107.

[3] Gupta JND. A Functional Heuristic Algorithm for the Flow shop Scheduling Problem. *Operations Research* 1971; 22:39 – 47.

[4] Campbell HG, Dudek RA and Smith ML. A Heuristic Algorithm for the *n* Job m Machine Sequencing Problem. *Management Science* 1970; 16:630-637.

[5] Dannenbring DG. An Evaluation of Flow-shop Sequencing Heuristics. *Management Science* 1977; 23:1174-1182.

[6] Nawaz M, Enscore Jr E and Ham I. A Heuristic Algorithm for the m-Machine, n-Job Flow-shop Sequencing Problem. *OMEGA, The International Journal Of Management Science* 1983; 11, No. 1:91-953.

[7] Baskar A and Anthony Xavior M. Solving Permutation Flow Shop Scheduling Problems with Job Idle Time in between Two Machines. *Key Engineering Materials* 2013, 531-532:699-702.

[8] Baskar A and Anthony Xavior M. A Simple Model to Optimize General Flow Shop Scheduling Problems with Known Break down Time and Weights of Jobs. *Procedia Engineering* 2012; 38:191-196.

[9] Etemadi R, Karam G and Majumdar S. Heuristic algorithms for priority assignment in flow shops. *Performance, Computing and Communications IPCCC, IEEE International* 1998; 15-22.

[10] Pei-Chann Chang, Jih-Chang Hsieh and Shui-Geng Lin. The development of gradual-priority weighting approach for the multi-objective flowshop scheduling problem. *International Journal of Production Economics*, 2002; 79, No. 3:171–183.

[11] Chandramouli AB. Heuristic approach for n-job, 3-machine flow-shop scheduling problem involving transportation time, breakdown time and weights of jobs. *Mathematical and Computational Applications* 2005; 10:301-305.

[12] Pandian P and Rajendran P. Solving Constrained Flow-Shop Scheduling Problems with Three Machines. *International Journal Of Contemporary Mathematical Sciences* 2010; 10 No.19:921-929.