International Conference on Mechanical, Industrial and Materials Engineering 2015 (ICMIME2015)
11-13 December, 2015, RUET, Rajshahi, Bangladesh.

Paper ID: IE-46

A Comparative Analysis of Heuristics for Optimizing the Makespan in Flow Shop Scheduling

Dewan Sabbir Ahammed RAYHAN[1] and Samiran CHAKMA[2]
[1, 2] Student, Department of Industrial and Production Engineering, Rajshahi University of Engineering and Technology, Rajshahi-6204, BANGLADESH
E-mail: [1] *sabbirrayhan88@gmail.com (Corresponding Author)* & [2] *samiranchakma92@gmail.com*

Abstract

*Scheduling is the allocation of launch and finish time to each particular order. Therefore scheduling can bring about a revolutionary change in productivity in shop floor by providing a calendar for processing a set of jobs. The single machine scheduling problem consists of **n** jobs with the same single operation on each of the jobs, while the flow shop scheduling problem consists of **n** jobs with **m** operations on each of the jobs. The main objective of this investigation is to find out the optimal makespan for a 5 jobs and 5 machines flow shop scheduling problem. Then the makespan that have been determined by Palmer's heuristic, CDS heuristic (Campbell Dudek Smith), Insertion algorithm (Nawaz Enscore Ham) and Gupta's algorithm are compared and which is better technique among these four techniques also have been shown in this paper. From the comparative analysis, it has been found that Insertion algorithm (NEH) up to 4 machines problem provides better results as compared to Palmer's heuristic, CDS heuristic, Insertion algorithm and Gupta's algorithm. Gant chart is provided for confirming the effectiveness of heuristics. The practice of these four techniques makes it conceivable to generate a schedule that minimizes the makespan.*

Keywords: Flow Shop Scheduling, Palmer's Heuristic, CDS heuristic, Insertion algorithm, Gupta's Heuristic, Makespan, Comparison of Makespan.

1. Introduction and Literature Review

Flow Shop Scheduling is one kind of decision making procedure and its intention is to optimize the allocation of resources over time to perform a collection of tasks. In a production process system a job is operated by a series of machines. In the case of multiple job operation, jobs must go through each machine in a sequence or a series so that a machine can operate every job effectively. But problems arise when scheduling and sequencing of these jobs, if the amount of Job and Machine is quit high. The order of processing of the required jobs with different processing times over different machines is included by a typical permutation flow shop scheduling problem [8].

A variety of objectives to be minimized for flow shop scheduling are total job completion time, total Flow time, Makespan, Tardiness based objectives etc. The flow shop scheduling was first introduced by Johnson in which a given set of machines are used to process a set of n jobs with an identical order [1]. Different heuristics and Meta heuristics are developed to solve the complex type of flow shop scheduling problems. The most well-known methods are Palmer Heuristics [4], Campbell Dudek Heuristics [5], Insertion algorithm (NEH) [6] and Gupta's Heuristic algorithm [7] for solving flow shop problems having machine greater than 3.

Literature shows that most of the heuristics developed over the last half century for minimizing makespan in flow shop scheduling. A K Sahu (2009) compared RA, Gupta's, CDS and Palmer's Heuristics in Flow Shop Scheduling on 8 jobs & 3 machines, 10 jobs & 8 machines & 10 jobs & 10 machines. He determined that RA heuristic performs well when compared to other heuristics. Malik A. & A. K. Dhingra (2013) had made a Comparative Analysis of Heuristics for Makespan Minimizing in Flow Shop Scheduling [13]. They concluded that NEH heuristic shows the minimum value of makespan when compared to other heuristics.

2. Problem Statement

The minimization of the makespan by four different heuristics and selection of the best heuristics that gives the optimal makespan of a flow shop scheduling problem is our main objective in this study. As the primary data of a flow shop is not available, a secondary data of a flow shop industry is used in this paper [9]. By using this secondary data, the minimum makespan and optimum job sequence have computed for four Heuristics. The makespan of these four methods are compared and the minimum (optimal) makespan is selected for this flow shop scheduling problem. This flow shop problem has the processing times for 5 jobs and 5 machines which are tabulated below;

Table 1: Processing times for 5 jobs and 5 machines in a flow shop [9]

⇩ ⇨	Processing time				
Job	Machine-1 (M1)	Machine-2 (M2)	Machine-3 (M3)	Machine-4 (M4)	Machine-5 (M5)
Job-1(J1)	5	9	8	10	1
Job-2(J2)	9	3	10	1	8
Job-3(J3)	9	4	5	8	6
Job-4(J4)	4	8	8	7	2
Job-5(J5)	3	5	6	3	7

3. Methodology

In order to minimize the makespan and to obtain the optimum job sequence palmers heuristics, CDS algorithm, Insertion algorithm and Gupta's heuristic algorithm are used in this paper. These methods consist of a series of steps which are given below;

3.1 Palmer Heuristics

To minimize the makespan in static flow shop, palmer has developed a heuristic which is known as the Palmer Heuristics. This algorithm sometimes gives optimal solution (Makespan) but not guarantee to give optimal solution for all the time. Palmer has developed a concept of slope to compute the optimum makespan and job sequence in flow shop. Based on this slope (A_j), this heuristic can evaluate only one optimal sequence of job. Palmer heuristics comprises two following steps [4, 3];

***Step1*:** If the number of job is n and machine is m, then the slope A_j for j^{th} job is computed as follows;

$$\mathbf{Aj} = \sum_{i=1}^{m} \{\mathbf{m} - (\mathbf{2i} - \mathbf{1})\}\mathbf{Pij} \qquad (1)$$

Where, P_{ij} denotes the processing time of the job.

Step 2: Jobs are arranged in the sequence according to their descending (decreasing) order of slope $\mathbf{A_j}$ value.

Table 2: Determination of job's slope

Job ⇩	M1	M2	M3	M4	M5	**Slope A_j**
J1	5	9	8	10	1	*-14*
J2	9	3	10	1	8	*-8*
J3	9	4	5	8	6	*-4*
J4	4	8	8	7	2	*-10*
J5	3	5	6	3	7	*+10*
m-(2i-1)	-4	-2	0	+2	+4	

Arranging slope values in decreasing order; the job sequence **J5-J3 -J2-J4-J1** is obtained.

Table 3: Makespan calculation of Palmer's heuristic for job sequence **J5-J3 -J2-J4-J1**

Job ⇩	M1	M2	M3	M4	M5	*Makespan*
J5	3	8	14	17	24	
J3	12	16	21	29	35	
J2	21	24	34	35	43	*61*
J4	25	33	42	49	51	
J1	30	42	50	60	61	

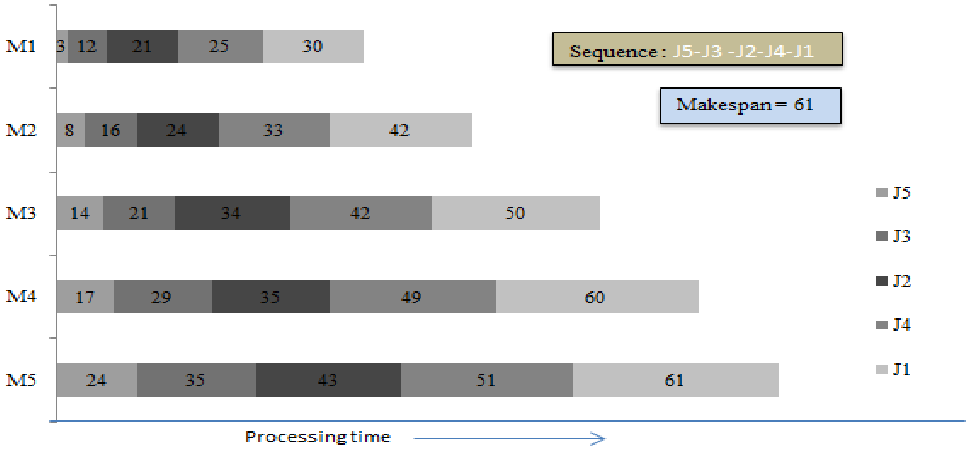

Figure 1: Gantt chart for sequence **J5-J3 -J2-J4-J1** (Palmer's Heuristic)

3.2 CDS Algorithm

An unostentatious heuristic algorithm was proposed by Campbell et al based on extension of Johnson's rule which is known as the Campbell, Dudek, and Smith (CDS) heuristic [5]. This method constructs (m-1) different sequences of jobs in which each sequence assembles to the engagement of Johnson's rule on a new 2-machines problem [10]. After computing the sequences, the best sequence is picked. This heuristic is better than the Palmer heuristics because it can evaluate (m-1) sequences. CDS heuristic comprises following steps;

***Step 1*:** Taking machine M1 & M5 and using Johnson's rule, sequence **J5-J2-J3-J4-J1** is obtained;

Table 4: Makespan calculation of M1 & M2 for job sequence **J5-J2-J3-J4-J1**

Job⇩	M1	M2	M3	M4	M5	*Makespan*
J5	3	8	14	17	24	
J2	12	15	25	26	34	
J3	21	25	30	38	44	*61*
J4	25	33	41	48	50	
J1	30	42	50	60	61	

***Step 2*:** Taking M1+M2 and M4+M5, the following sequence is obtained that is shown in Table 5;

Table 5: Makespan calculation for M1+M2 & M4+M5

Sequence	Makespan
J5-J3-J1-J4-J2	61
J5-J3-J1-J2-J4	61

***Step 3*:** Taking M1+M2+M3 and M3+M4+M5, the following sequence is obtained that is shown in Table 6;

Table 6: Makespan calculation for M1+M2+M3 & M3+M4+M5

Sequence	Makespan
J5-J3-J2-J1-J4	62
J5-J3-J1-J2-J4	61

Among the two sequences, J5-J3-J1-J2-J4 sequence is selected because of minimum makespan 61.

***Step 4*:** Taking M1+M2+M3+M4 and M2+M3+M4+M5, the sequence **J5-J1-J4-J3-J2** is obtained and the makespan is calculated which is shown in Table 7 below;

Table 7: Makespan calculation for M1+M2+M3+M4 and M2+M3+M4+M5

Job⇩	M1	M2	M3	M4	M5	*Makespan*
J5	3	8	14	17	24	
J1	8	17	25	35	36	
J4	12	25	33	42	44	*64*
J3	21	29	38	50	56	
J2	30	33	48	51	64	

***Step 5*:** From the step 1, step 2, step 3and step 4; job sequences **J5-J2-J3-J4-J1, J5-J3-J1-J2-J4, J5-J3-J1-J4-J2** are selected for optimum makespan 61.

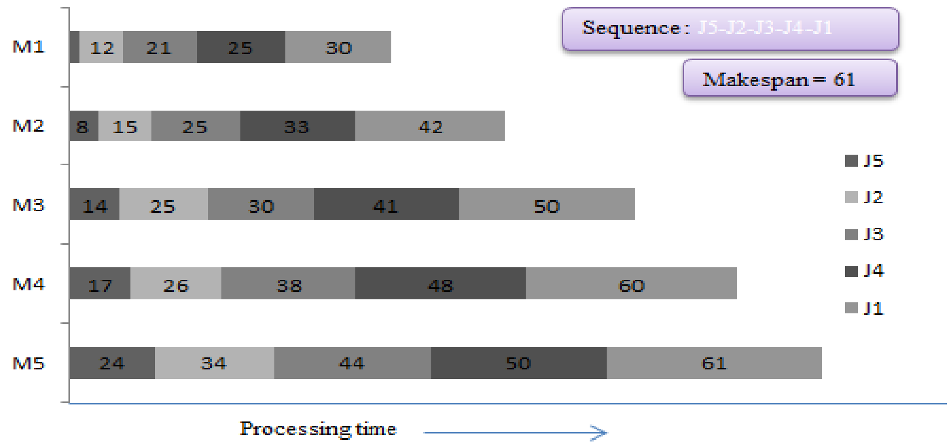

Figure 2: Gantt chart for sequence **J5-J2-J3-J4-J1** (CDS Heuristics)

3.3 Insertion (NEH) Algorithm

Newaz, Enscor and Ham (NEH) algorithm concept creates job sequence in iterative manner [6]. It evaluates n-sequences and results better than Palmer and CDS heuristics. It comprises following steps [6];

***Step 1*:** The total work content (**T_j**) for each job is calculated using expression;

$$\mathbf{Tj} = \sum_{i=1}^{m} \mathbf{Pij} \qquad (2)$$

***Step 2*:** The jobs are arranged in a work content list according to decreasing values of **T_j.**
***Step 3*:** First two jobs are selected from the list and from two partial sequences by inter changing the place of two jobs. The value of partial sequences is computed. Of the two sequences, the sequence having larger value of Makespan is discarded. The lower value of Makespan is called as incumbent sequence.
***Step 4*:** The next job is picked and put in incumbent sequence. The value of Makespan of all sequences is calculated.
***Step 5*:** If there is no job left in work content list to be added to Incumbent sequence then STOP otherwise go to step 4.

Table 8: Makespan and job sequence for Insertion algorithm

Sequence	Makespan	Optimal Makespan & Sequence
J5-J4-J3-J1-J2	58	Optimal
J4-J5-J3-J1-J2	58	Optimal
J4-J3-J5-J1-J2	58	Optimal
J4-J3-J1-J5-J2	63	Not optimal
J4-J3-J1-J2-J5	61	Not optimal

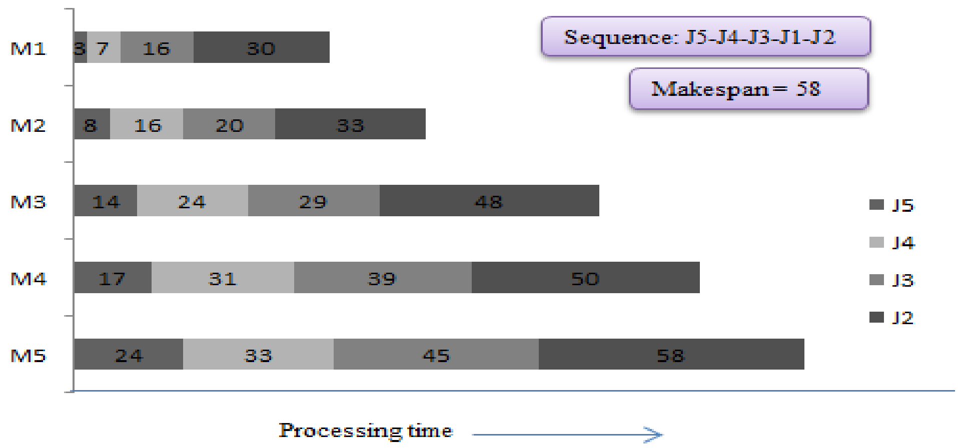

Figure 3: Gantt chart for sequence **J5-J4 –J3-J1-J2** (Insertion Algorithm)

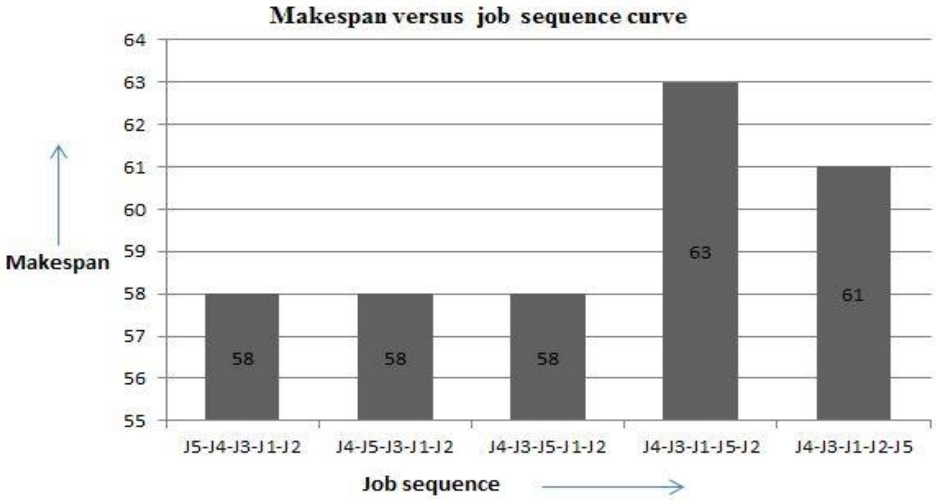

Figure 4: Makespan versus job sequence curve for Insertion (NEH) Algorithm

3.4 Gupta's Heuristic Algorithm

Gupta developed a heuristic to achieve a nearly minimum makespan in which all the jobs are allocated into two groups by comparing the dispensation times of the first machine and the last machine in each job [7]. For each group, the sum of processing times of any two adjoining job are calculated and the minimum processing time is quested, and then the jobs are arranged according to the ascending order of summed processing times [2, 7].

Step 1: Form the group of jobs **U** less time is taken on the first machine than on the last machine.
Step 2: Form the group of jobs **V** less time is taken on the last machine than on the first machine.
Step 3: For each job Ji in **U**, the minimum of (tkj + t(k+1)j) is calculated for k = 1 to m-1.
Step 4: For each job Jj in **V**, the minimum of (tkj + t(k+1)j) is calculated for k = 1 to m-1.
Step 5: The jobs in **U** in ascending order of πi's are sorted; if two or more jobs have the same value of πi then the jobs are sorted in an arbitrary order.
Step 6: The jobs in **V** in descending order of πj's are sorted; if two or more jobs have the same value of πj then the jobs are sorted in an arbitrary order.
Step 7: The jobs on the machines in the sorted order of **U** are scheduled, then in the sorted order of **V** [2].

Table 9: Makespan and job sequence for Gupta's Heuristic Algorithm

Sequence	Makespan	Optimal Makespan & Sequence
J2-J3-J4-J5-J1	63	Not Optimal
J3-J2-J4-J5-J1	64	Not Optimal
J4-J2-J3-J5-J1	60	Optimal
J5-J2-J3-J4-J1	61	Not optimal

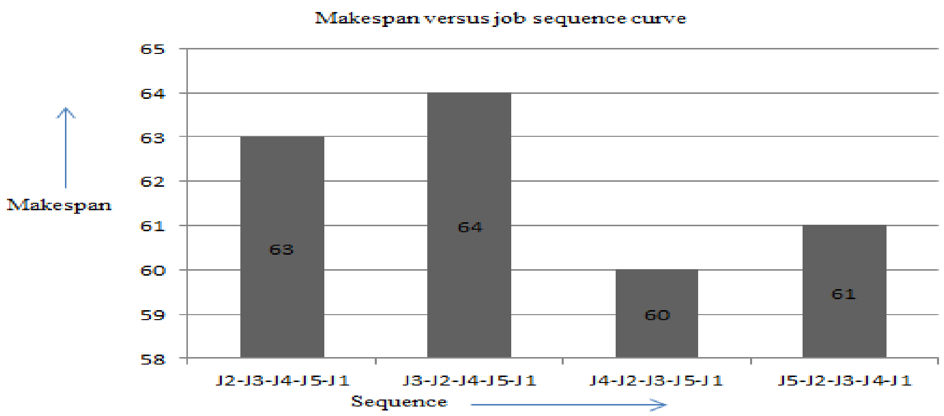

Figure 5: Makespan versus job sequence curve for Gupta's Heuristic

4. Results

Table 10: Makespan Comparison of Palmer Heuristics, CDS Heuristics, Insertion Algorithm and Gupta's Heuristics

Method No.	Method Name	Optimal sequence	Optimal Makespan
1	Palmer's Heuristic	J5-J3 -J2-J4-J1	61
2	CDS Heuristics	J5-J2-J3-J4-J1; J5-J3-J1-J2-J4; J5-J3-J1-J4-J2	61
3	Insertion (NEH) Algorithm	**J5-J4-J3-J1-J2; J4-J5-J3-J1-J2; J4-J3-J5-J1-J2**	**58**
4	Gupta's Heuristic	J4-J2-J3-J5-J1	60

The Makespan of the Palmer Heuristics, CDS Heuristics, Insertion Algorithm and Gupta's Heuristic are compared. From the comparison table 10, it is clear that Insertion Algorithm (NEH) results the best optimal Makespan (58) and gives the best optimal job sequence. So to achieve best optimal makespan, job sequences **J5-J4-J3-J1-J2; J4-J5-J3-J1-J2; J4-J3-J5-J1-J2** can be employed in the flow shop.

5. Conclusion and Summary

Scheduling plays an important role in several flow shop industries. Without proper scheduling productivity decreases that may reduce the profit of the organization and cause an increased price of the product. This paper mainly shows how the makespan are minimized by using the Palmer's Heuristic, CDS Heuristic, Insertion Algorithm and Gupta's heuristic of a 5 job and 5 machine flow shop problem and their differences. Finally, this analysis proves that the Insertion (NEH) algorithm is the best among these four heuristics because it results most optimum makespan 58 for the given problem. In future, this work can help the flow shop industries and researches to identify the effective and efficient heuristic techniques for resolving the flow shop scheduling problems. By elaborating the size of the problems and adding any assumptions this analysis can be extended. If the hybrid methods of these heuristics are developed in future, larger type of flow shop scheduling problem can be mitigated.

6. References

[1] S. M. Johnson, "Optimal two-and three-stage production schedules with setup times included," *Naval Research Logistics Quarterly*, vol. 1, no. 1, pp. 61–68, 1954.

[2] Sandeep Kumar, Pooja Jadon, "A Novel Hybrid Algorithm for Permutation Flow Shop Scheduling", *International Journal of Computer Science and Information Technologies*, vol. 5, no. 4, pp. 5057-5061, 2014.

[3] R. Paneerselvam, Production and Operations management, Asoke K. Ghosh publisher, Third Edition, pp. 276-291, 2013.

[4] D. Palmer, "Sequencing jobs through a multi-stage process in the minimum total time – a quick method of obtaining a near optimum", *Operation Research*, vol. 16, pp. 101-107, 1965.

[5] H. G. CAMPBELL, R. A. DUDEK and M. L. SMITH, "A heuristic algorithm for the n-job, m-machine sequencing problem", *Management Science*, vol. 16, pp. 630 – 637, 1970.

[6] M. Nawaz, Jr. E. Enscore and I. Ham, A Heuristic Algorithm for the m-Machine, n-job flow-shop sequencing problem, *OMEGA - Elsevier*, vol. 11, pp. 91 – 95, 1983.

[7] J. N. D. Gupta, "Heuristic algorithm for multistage flow shop problem", *AIIE Transactions*, vol. 4, pp. 11-18, 1972.

[8] Baskar A , Anthony Xavior M , "Analyzing a few Heuristic Algorithms Considering Machine Idle Time and Processing Time in Permutation Flow Shop Scheduling Problems", *International Journal of Applied Information Systems Foundation of Computer Science FCS, New York, USA*, vol. 6 , no. 4, 2013.

[9] Ekta Singhal, Shalu Singh, Aneesh Dayma, "An Improved Heuristic for Permutation Flow Shop Scheduling (NEH ALGORITHM)", *International Journal of Computational Engineering Research*, vol. 2, 2012.

[10] Vladimír Modrák, R. Sudhakara Pandian, "Flow Shop Scheduling Algorithm To Minimize Completion Time For *N*-Jobs *M*-Machines Problem", *Technical Gazette 17*, vol. 3, pp. 273-278, 2010.

[11] Md. Sanowar Hossain, Md. Asadujjaman, Md. Ashraful Alam Nayon, Priyanka Bhattacharya, "Minimization of Makespan in Flow Shop Scheduling Using Heuristics", *International Conference on Mechanical, Industrial and Energy Engineering*, no. PI-140163, 2014.

[12] Chia, Lee, "Minimizing the total completion Time in permutation flow shop", *Computers and operations research,* vol.6, pp. 2111-2121, 2009.

[13] Malik A. & A. K. Dhingra, " Comparative Analysis of Heuristics for Makespan Minimizing in Flow Shop Scheduling", *International Journal of Innovations in Engineering and Technology*, vol. 2, issue 4, pp. 263-269, 2013.