Flowshop Scheduling Problem for 10-Jobs, 10-Machines By Heuristics Models Using Makespan Criterion

Ajay Kumar Agarwal
Assistant Professor, Mechanical Engineering Department
RIMT, Chidana, Sonipat, Haryana, India

Rajan Garg
Assistant Professor & H.O.D
Mechanical Engineering Department
SDITM, Israna (Panipat), India

Abstract: In modern manufacturing there is the trend of the development of the Computer Integrated Manufacturing (CIM). CIM is computerized integration of the manufacturing activities (Design, Planning, Scheduling and Control) which produces right product(s) at right time to react quickly to the global competitive market demands. The productivity of CIM is highly depending upon the scheduling of Flexible Manufacturing System (FMS). Machine idle time can be decreased by sorting the make span which results in the improvement in CIM productivity. Conventional methods of solving scheduling problems based on priority rules still result schedule, sometimes with idle times. To optimize these, this paper models the problem of a flowshop scheduling with the objective of minimizing the makespan. The work proposed here deal with the production planning problem of a flexible manufacturing system. This paper models the problem of a flowshop scheduling with the objective of minimizing the make span. The objective is to minimize the makespan of batch-processing machines in a flowshop. The processing times and the sizes of the jobs are known and non-identical. The machines can process a batch as long as its capacity is not exceeded. The processing time of a batch is the longest processing time among all the jobs in that batch. The problem under study is NP-hard for makespan objective. Consequently, comparisons based on Palmer's and Gupta's heuristics are proposed in this work. Gantt chart is generated to verify the effectiveness of the proposed approaches.

Keywords: CIM; FMS; Flowshop scheduling problem; Makespan; Heuristic models; Gantt chart

I. INTRODUCTION

A Flexible manufacturing system (FMS) consists of a collection of numerically controlled machines with multifunction ability, an automatic material handling system and an online computer network. This network is capable of controlling and directing the whole system. An FMS combines the advantages of a traditional flow line and job-shop systems to meet the changing demands. Thus, it involves many problems, which can be divided into four stages: (a) design, (b) system set-up, (c) scheduling and (d) control. FMS Scheduling system is one of the most important information-processing subsystems of CIM system. The productivity of CIM is highly depending upon the quality of FMS scheduling. The basic work of scheduler is to design an optimal FMS schedule according to a certain measure of performance, or scheduling criterion. This work focuses on productivity oriented-makespan criteria. Makespan is the time length from the starting of the first operation of the first demand to the finishing of the last operation of the last demand. The inherent efficiency of a flexible manufacturing system (FMS) combined with additional capabilities, can be harnessed by developing a suitable production plan. Machine scheduling problems arises in diverse areas such as flexible manufacturing system, production planning, computer design, logistics, communication etc. A common feature of many of these problems is that no efficient solution algorithm is known yet for solving it to optimality in polynomial time.

The classical flowshop scheduling problem is one of the most well known scheduling problems. Informally the problem can be described as follows:

There are set of jobs and a set of machines. Each job consists of chain of operation, each of which needs to be processed during an uninterrupted time period of a given length on a given machine. Each machine can process at most one operation at a time. A schedule is an allocation of operations to time intervals of the machines. The problem is to find the schedule of minimum length. This work tries to minimize the makespan of batch-processing machines in a flowshop. The processing times and the sizes of the jobs are known and non-identical. The machines can process a batch as long as its capacity is not exceeded. The processing time of a batch is the longest processing time among all the jobs in that batch. The problem under study is NP-hard for make-span objective. Consequently, comparisons based on Palmer's and Gupta's heuristics are proposed. Gantt chart is generated to verify the effectiveness of the proposed approaches.

II. SEQUENCING AND SCHEDULING

Sequencing refers to arranging items or events in a particular order. In other words Sequencing is a technique to order the jobs in a particular sequence. In industries there are different types of sequencing which are followed such as first in first out basis, priority basis, job size basis and processing time basis etc. In processing time basis sequencing for different sequence, we will achieve different processing time. The sequence is adapted which gives minimum processing time.

Scheduling is a decision making process and it concerns the allocation of the limited resources to tasks over time By Scheduling, we assign a particular time for completing a particular job. The main objective of scheduling is to arrive at a position where we will get minimum processing time.

III. SIGNIFICANCE OF WORK

By establishing the timing of the use of equipment, facilities and human activities in an organization can:

i. Determine the order in which jobs at a work center will be processed.

ii. Results in an ordered list of jobs

Sequencing is most beneficial when we have constrained capacity (fixed machine set; cannot buy more and heavily loaded work centers

iii. Lightly loaded work centers = no big deal (excess capacity)

iv. Heavily loaded
   - a) Want to make the best use of available capacity.
   - b) Want to minimize unused time at each machine as much as possible.

IV. PARAMETERS OF THE WORK

i. Average job flow time
   - a) Length of time (from arrival to completion) a job is in the system, on average
   - b) Lateness

ii. Average length of time the job will be late (that is, exceeds the due date by)

iii. Makespan

iv. Total time to complete all jobs

v. Average number of jobs in the system

vi. Measure relating to work in process inventory

vii. Equals total flow time divided by makespan.

V. OBJECTIVE

1. *Minimizing the makespan.* To deal with the production planning problem of a flexible manufacturing system, I model the problem of a flowshop scheduling with the objective of minimizing the makespan.
2. *To provide a schedule for each job and each machine.* Schedule provides the order in which jobs are to be done and it projects start time of each job at each work center.
3. *Comparative study.* To select appropriate heuristics approach for the scheduling problem through a comparative study.
4. *To solve FMS scheduling problem in a flow-shop environment.* Considering the comparison based on Palmer's and Gupta's heuristics are proposed. Gantt chart is generated to verify the effectiveness of the proposed approaches.

My objective of scheduling can yield:

1. Efficient utilization …
   - a) Staff
   - b) Equipment
   - c) Facilities
2. Minimization of …
   - a) customer waiting time
   - b) Inventories.
   - c) Processing time.

VI. METHODOLOGY

Operations planning and scheduling (OPS) problems in flexible manufacturing systems (FMSs), are composed of a set of interrelated problems, such as part-type batching, machine grouping, part routing, tool loading, part input sequencing, and resource assignment. The performance of an FMS is highly dependent on the efficient allocation of the limited resources to the tasks, and it is strongly affected by the effective choice of scheduling rules.

In this study, a heuristic ruled based approach for dynamic scheduling of FMSs, which integrates loading, part inputting, routing, and dispatching issues of the OPS, is presented, and the implementation results are compared with several dispatching rules. Manufacturing scheduling theory is concerned with the right allocation of machines to operations over time. The basic work of scheduler is to design an optimal FMS schedule according to a certain measure of performance, or scheduling criterion. This work focuses on productivity oriented-makespan criteria. Makespan is the time length from the starting of the first operation of the first demand to the finishing of the last operation of the last demand. The approach used in this work was the comparisons based on four heuristic algorithms namely Palmer's and Gupta's algorithm are proposed. Here the main objective is to compare and find the efficient heuristics algorithm for minimizing the makespan. In this work hierarchical approach were used to determine the optimal makespan criteria.

VII. PROBLEM STATEMENT

There is a flowshop scheduling problem in which all the parameters like processing time, due date, re-fixturing time, and set-up time are given. The value of the makespan of batch-processing machines in a flowshop based on comparisons of Palmer's and Gupta's heuristics are proposed. Analytic solutions in all the heuristics are investigated. Gantt chart is generated to verify the effectiveness of the proposed approaches. Here the heuristics approaches for planning problems are proposed which provides a way to optimize the makespan which is our objective function.

VIII. FLOWSHOP SCHEDULING

It is a typical combinatorial optimization problem, where each job has to go through the processing in each and every machine on the shop floor. Each machine has same sequence of jobs. The jobs have different processing time for different machines. So in this case we arrange the jobs in a particular order and get many combinations and we choose that combination where we get the minimum makespan.

In an m-machine flowshop, there are m stages in series, where there exist one or more machines at each stage. Each job has to be processed in each of the m stages in the same order. That is, each job has to be processed first in stage 1, then in stage 2, and so on. Operation times for each job in different stages may be different. We classify flowshop problems as:

1. Flowshop (there is one machine at each stage).
2. No-wait flowshop (a succeeding operation starts immediately after the preceding operation completes).
3. Flexible (hybrid) flowshop (more than one machine exist in at least one stage) and
4. Assembly flowshop (each job consists of specific operations, each of which has to be performed on a pre-determined machine of the first stage, and an assembly operation to be performed on the second stage machine).

IX. FLOWSHOP SCHEDULING METHODS

Heuristics for general *m*-Machine Problems

1. Palmer's Heuristic Algorithm.
2. Gupta's Heuristic Algorithm.

X. GENERAL DESCRIPTION

1. There are *m* machines and *n* jobs.
2. Each job consists of *m* operations and
   a) each operation requires a different machine
3. *n* jobs have to be processed in the same sequence on *m* machines.
4. Processing time of job *i* on machine *j* is given by t_{ij}
   a) (i =1… n ; j =1,…,m)
5. Makespan: find the sequence of jobs minimizing the maximum flow time.

XI. MAIN ASSUMPTIONS

1. Every job has to be processed on all machines in the order (j = 1, 2, …., m).
2. Every machine processes only one job at a time.

3. Every job is processed on one machine at a time.
4. Operations are not preemptive.
5. Set-up times for the operations are sequence-independent and are included in the processing times.

Operating sequences of the jobs are the same on every machine, and the common sequence has to be determined.

XII. THREE CATEGORIES OF FSP

1. Deterministic flow-shop scheduling problem:
   - ¾ Assume that fixed processing times of jobs are known.
2. Stochastic flow-shop scheduling problem:
   - ¾ Assume that processing times vary according to chosen probability distribution.
3. Fuzzy flow-shop scheduling problem:
   - ¾ Assume that a fuzzy due date is assigned to each job to represent the grade of satisfaction of decision makers for the completion time of the job.

XIII. HEURISTICS FOR GENERAL 10-MACHINES AND 10-JOBS PROBLEMS

1. Palmer's Heuristic Algorithm.
2. Gupta's Heuristic Algorithm.

1. Palmer's Heuristic Rule:

Algorithm: Palmer's Heuristic
Procedure: Palmer's Heuristic
Input: job list *i, machine m*;
Output: schedule *S*;
begin
for $i = 1$ to n
for $j = 1$ to m
Calculates Slope = $(m-1)t_{j,m} + (m-3)t_{j,(m-1)} + (m-5)t_{j,(m-2)} \ldots\ldots\ldots$ // step 1:
Permutation schedule is constructed by sequencing the jobs in Non-increasing order of s_i such as: $s_{i1} \geq s_{i2} \geq \ldots\ldots \geq s_{in}$; // step 2:
end
Output optimal sequence is obtained as schedule S; // step 3:
end.

Consider a 10-job problem:

Table1: General 10-Jobs, 10-Machines Problem

Job ⇨ M/c ⇩	1	2	3	4	5	6	7	8	9	10
1	5	2	1	7	6	3	7	5	7	4
2	2	6	2	5	6	7	2	1	8	3
3	3	4	2	6	1	5	4	7	6	5
4	5	2	1	3	8	2	6	1	9	8
5	7	6	3	2	6	2	5	7	1	3
6	9	2	7	3	4	1	5	3	8	1
7	7	5	2	2	3	5	1	6	2	3
8	8	2	5	4	9	3	2	6	1	8
9	2	6	4	2	6	2	5	2	6	3
10	7	1	4	2	4	6	2	2	6	7

The solution constructed as follows:

Step 1:
For 10-jobs and 10- machines:
$s_1 = (m-1)t_{1,10} + (m-3)t_{1,9} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots + (m-19)t_{1,1}$

For 10 machines (m=10)

= (10-1)*7+(10-3)*2+(10-5)*8+(10-7)*7+(10-9)*9+(10-11)*7+(10-13)*5+(10-15)*3+(10-17)*2+(10-19)*5

= 51

Similarly

$s_2 = -14$

$s_3 = 63$

$s_4 = -78$

$s_5 = 5$

$s_6 = -10$

$s_7 = -49$

$s_8 = -14$

$s_9 = -62$

$s_{10} = 25$

Step 2: Optimal sequence is constructed on the basic of decreasing order of slope values

Optimal sequence: $s_3 \geq s_1 \geq s_{10} \geq s_5 \geq s_6 \geq s_2 \geq s_8 \geq s_7 \geq s_9 \geq s_4$

Step 3: Output optimal sequence is {3, 1, 10, 5, 6, 2, 8, 7, 9, 4}

Thus total processing time can be calculated as:

Table 2. Total Processing Time for 10-Jobs, 10-Machines by Palmer's Heuristic Model

Job i	M/c 1 Time		M/c 2 Time		M/c 3 Time		M/c 4 Time		M/c 5 Time		M/c 6 Time		M/c 7 Time		M/c 8 Time		M/c 9 Time		M/c 10 Time	
	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out
3	0	1	1	3	3	5	5	6	6	9	9	16	16	18	18	23	23	27	27	31
1	1	6	6	8	8	11	11	16	16	23	23	32	32	39	39	47	47	49	49	56
10	6	10	10	13	13	18	18	26	26	29	32	33	39	42	47	55	55	58	58	65
5	10	16	16	22	22	23	26	34	34	40	40	44	44	47	55	64	64	70	70	78
6	16	19	22	29	29	34	34	36	40	42	44	45	47	52	64	67	70	72	78	84
2	19	21	29	35	35	39	39	41	42	48	48	50	52	57	67	69	72	78	84	85
8	21	26	35	36	39	46	46	47	48	55	55	58	58	64	69	75	78	80	85	87
7	26	33	36	38	46	50	50	56	56	61	61	66	66	67	75	77	80	85	87	89
9	33	40	40	48	50	56	56	65	65	66	66	74	74	76	77	78	85	91	91	97
4	40	47	48	53	56	62	65	68	68	70	74	77	77	79	79	83	91	93	97	99

Therefore, total processing time = 99

Total Idle Time for M/c 1 = 99-47 = 52 (Units)

Total Idle Time for M/c 2 = 1+3+2+3+2+(99-53) = 57 (Units)

Total Idle Time for M/c 3 = 3+3+2+4+6+1+(99-62) = 56 (Units)

Total Idle Time for M/c 4 = 5+5+2+3+5+3+(99-68) = 54 (Units)

Total Idle Time for M/c 5 = 6+6+3+5+1+4+2+(99-70) = 56 (Units)

Total Idle Time for M/c 6 = 12+4+7+3+5+3+(99-77) = 56 (Units)

Total Idle Time for M/c 7 = 19+11+2+1+2+7+1+(99-79) = 63 (Units)

Total Idle Time for M/c 8 = 21+13+1+(99-83) = 51 (Units)

Total Idle Time for M/c 9 = 26+17+6+8+(99-93) = 63 (Units)

Total Idle Time for M/c 10 = 30+15+2+5+2 = 54 (Units)

The Gantt chart according to Table 2. is shown in Figure 1.

2. Gupta's Heuristic Rule:

Algorithm: Gupta's Heuristic

Procedure: Gupta's Heuristic
Input: job list i, machine m;
Output: schedule S;
begin
    for $i = 1$ to n
    for $k = 1$ to $m-1$
if $t_{i1} < t_{im}$ then
    $e_i = 1$;
else
    $e_i = -1$;
    calculate $s_i = e_i / \min\{t_{ik} + t_{i,k+1}\}$; // step 1:
end
    Permutation schedule is constructed by sequencing the jobs in Non-increasing order of s_i such as:; // step 2:
end
    Output optimal sequence is obtained as schedule S. // step 3:
end .

Consider the above 10-job and 8-machine problem:

The solution constructed as follows:

Step 1: Set the slope index s_i for job i as:

$s_1 = 1/\min(48, 50) = 0.0208$
$s_2 = -1/\min(35, 34) = -0.0294$
$s_3 = 1/\min(27, 30) = 0.0370$
$s_4 = -1/\min(34, 29) = -0.0345$
$s_5 = -1/\min(49, 47) = -0.0213$
$s_6 = 1/\min(30, 33) = 0.0333$
$s_7 = -1/\min(37, 32) = -0.0313$
$s_8 = -1/\min(38, 35) = -0.0286$
$s_9 = -1/\min(48, 47) = -0.0213$
$s_{10} = 1/\min(38, 41) = 0.0263$

Step 2: Jobs are sequenced according:

Optimal sequence:

$0.037 \geq 0.033 \geq 0.026 \geq 0.021 \geq -0.021 \geq -0.021 \geq -0.029 \geq -0.029 \geq -0.031 \geq -0.034$

$s_3 \geq s_6 \geq s_{10} \geq s_1 \geq s_5 \geq s_9 \geq s_8 \geq s_2 \geq s_7 \geq s_4$

Step 3: Output optimal sequence is {3, 6, 10, 1, 5, 9, 8, 2, 7, 4}

Thus total processing time can be calculated as:

Table 3. Total Processing Time for 10-Jobs, 10-Machines by Gupta's Heuristic Model

Job i	M/c 1 Time		M/c 2 Time		M/c 3 Time		M/c 4 Time		M/c 5 Time		M/c 6 Time		M/c 7 Time		M/c 8 Time		M/c 9 Time		M/c 10 Time	
	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out	In	Out
3	0	1	1	3	3	5	5	6	6	9	9	16	16	18	18	23	23	27	27	31
6	1	4	4	11	11	16	16	18	18	20	20	21	21	26	26	29	29	31	31	37
10	4	8	11	14	16	21	21	29	29	32	32	33	33	36	36	44	44	47	47	54
1	8	13	14	16	21	24	29	34	34	41	41	50	50	57	57	65	65	67	67	74
5	13	19	19	25	25	26	34	42	42	48	50	54	57	60	65	74	74	80	80	88
9	19	26	26	34	34	40	42	51	51	52	54	62	62	64	74	75	80	86	88	94
8	26	31	34	35	40	47	51	52	52	59	62	65	65	71	75	81	86	88	94	96
2	31	33	35	41	47	51	52	54	59	65	65	67	71	76	81	83	88	94	96	97
7	33	40	41	43	51	55	55	61	65	70	70	75	76	77	83	85	94	99	99	101
4	40	47	47	52	55	61	61	64	70	72	75	78	78	80	85	89	99	101	101	103

Therefore, total processing time = 103 (Units)

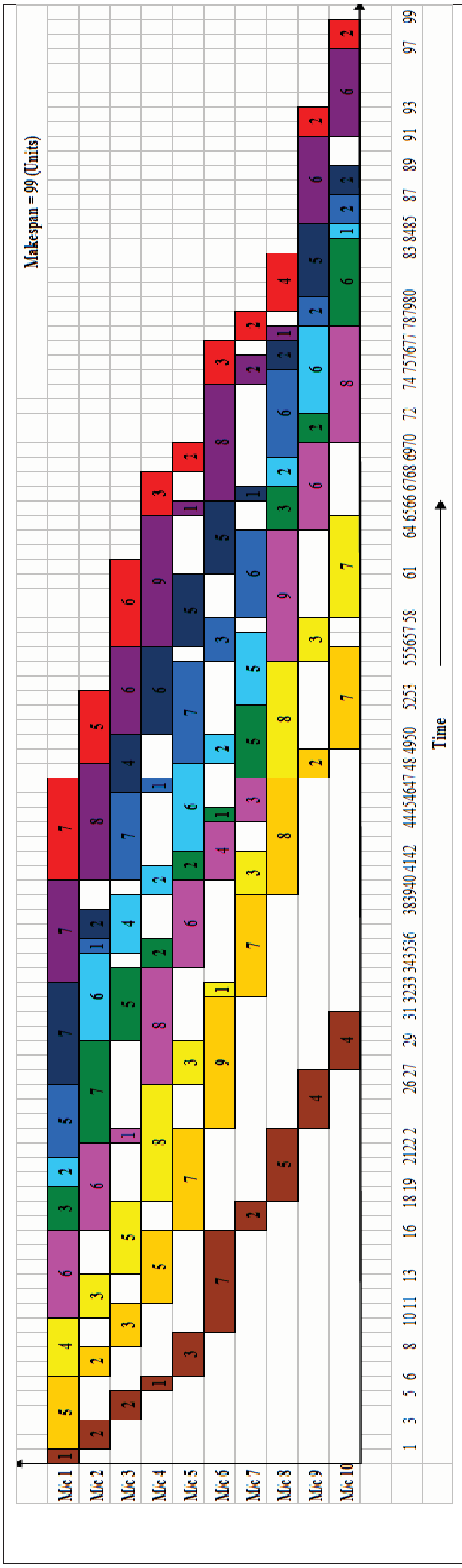

Fig.1: Gantt Chart for 10-Jobs, 10-Machines Problem by Palmer's Heuristic Model

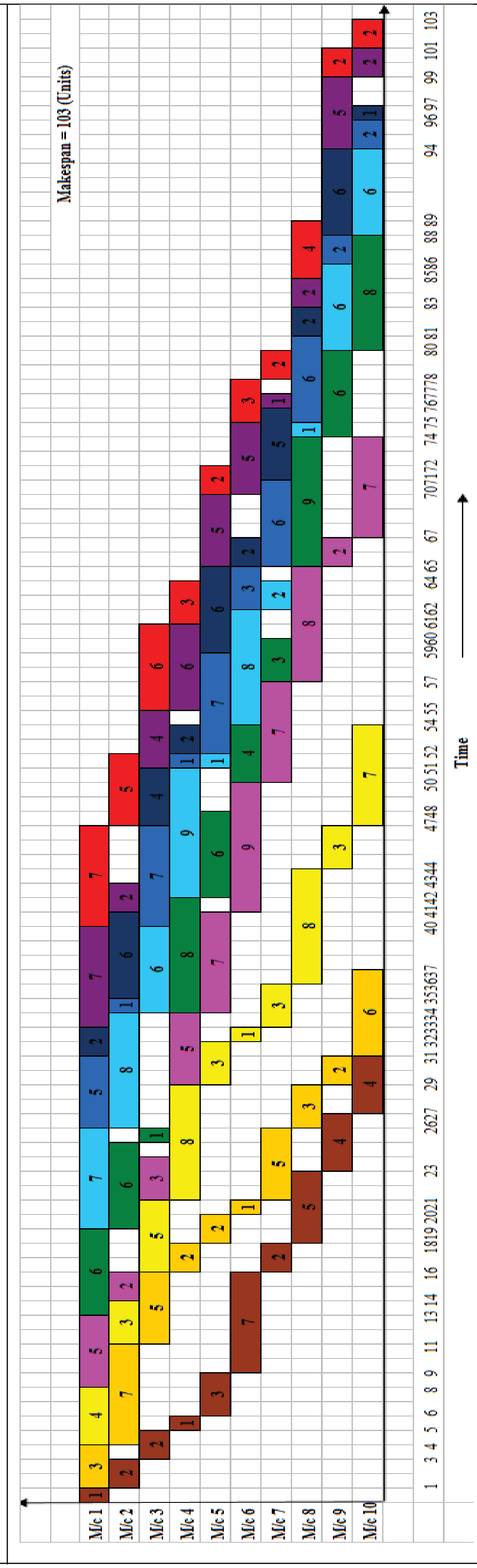

Fig.2: Gantt Chart for 10-Jobs, 10-Machines Problem by Gupta's Heuristic Model

Total Idle Time for M/c 1 = 103-47 = 56 (Units)
Total Idle Time for M/c 2 = 1+1+3+1+4+(103-52) = 61 (Units)
Total Idle Time for M/c 3 = 3+6+1+8+(103-61) = 60 (Units)
Total Idle Time for M/c 4 = 5+10+3+1+(103-64) = 58 (Units)
Total Idle Time for M/c 5 = 6+9+9+2+1+3+(103-72) = 61 (Units)
Total Idle Time for M/c 6 = 9+4+11+8+3+(103-78) = 60 (Units)
Total Idle Time for M/c 7 = 16+3+7+14+2+1+2+(103-80) = 68 (Units)
Total Idle Time for M/c 8 = 18+3+7+13+(103-89) = 55 (Units)
Total Idle Time for M/c 9 = 23+2+13+18+7+(103-101) = 65 (Units)
Total Idle Time for M/c 10 = 27+10+13+6+2 = 58 (Units)

The Gantt chart according to Table 3. is shown in Fig. 2

XIV. RESULTS

Makespan for the applied heuristics rules are:

Rule	Palmer’s	Gupta’s
Makespan	99 Units	103 Units

“Makespan is the time length from the starting of the first operation of the first demand to the finishing of the last operation of the last demand.”

XV. CONCLUSION AND FUTURE SCOPE

By Scheduling, we assign a particular time for completing a particular job. The main objective of scheduling is to arrive at a position where we will get minimum processing time. The problem examined here is the n-job, m-machine problem in a flow shop .This work arrange the jobs in a particular order and get many combinations and choose that combination where we get the minimum make span. This study tries to solve the problem of a flow shop scheduling with the objective of minimizing the makes pan. Here the objective is to minimize the make span of batch-processing machines in a flowshop. Comparisons based on Palmer’s and Gupta’s heuristics, are proposed here. Analytic solutions in these heuristics are investigated. Gantt chart is generated to verify the effectiveness of the proposed approaches. As a result of the work proposed here the researcher found that out of the Palmer’s Heuristic Model and Gupta’s Heuristic Model, the earlier one is the best Heuristic Model because of make span is minimum than that of later, so the Palmer’s Heuristic Model is best than Gupta’s Heuristic Model based upon the comparative study in this paper .

Further research may be conducted to investigate the applications of other meta-heuristics to the lot-streaming flow shop problem. Future research should address problems with different shop environments, including parallel machines flow shop, job shop, and open shop. Problems with other performance measures, such as minimum due dates, maximum lateness, and multi-criteria measures should also be studied. Future research should be directed to generalize the method to multipart, multi machine group cases.

REFERENCES

[1] Ajay Kumar Agarwal, Rajan Garg, “Advanced Modelistic Approach of Flowshop Scheduling Problem for 10-Jobs, 8-Machines by Heuristics Models Using Makespan Criterion”, International Journal of Engg., Business and Enterprise Applications (IJEBEA) December 2012 – Feb., 2013 Issue (Under Publication Process)

[2] Ajay Kumar Agarwal, Rajan Garg, “Flowshop Scheduling Problem for 10-Jobs, 8-Machines by Heuristics Models Using Makespan Criterion”, International Journal of Mechanical Science and Civil Engineering (IJMSCE) Volume 2 Issue 1 February 2013 (Under Publication Process)

[3] Ajay Kumar Agarwal, Rajan Garg, “Advanced Modelistic Approach of Flowshop Scheduling Problem for 8-Jobs, 3-Machines by Heuristics Models Using Makespan Criterion”, International Journal of Mechanical Science and Civil Engineering (IJMSCE) Volume 1 Issue 1 December 2012

[4] Ajay Kumar Agarwal, “Flowshop Scheduling Problem for 8-Jobs, 3-Machines with Makespan Criterion”, International Journal of Applied Engineering Research (IJAER) ISSN 0973-4562 Vol. 7 No. 11(2012) 1757–1762 (Article in a Conference Proceedings)

[5] Adam Janiak, Tomasz Krysiak, Costas P. Pappis, Theodore G. Voutsinas, “A scheduling problem with job values given as a power function of their completion times”, European Journal of Operational Research 193 (2009) 836-848

[6] Tamer Eren, Ertan Gu¨ner, “A bicriteria flowshop scheduling with a learning effect” , Applied Mathematical Modelling 32 (2008) 1719-1733

[7] Xianyi Wu, Xian Zhou, “Stochastic scheduling to minimize expected maximum lateness”, European Journal of Operational Research 190 (2008) 103–115

[8] Seong-Jong Joo, “Scheduling preventive maintenance for modular designed components: A dynamic approach”, European Journal of Operational Research 192 (2009) 512-520

[9] K.H. Ecker, J.N.D. RA, “Scheduling tasks on a flexible manufacturing machine to minimize tool change delays”, European Journal of Operational Research 164 (2005) 627–638
[10] Lixin Tang, Hua Gong, “A hybrid two-stage transportation and batch scheduling problem”, Applied Mathematical Modelling 32 (2008) 2467–2479
[11] Jiaqiong Chen, Ronald G. Askin, “Project selection, scheduling and resource allocation with time dependent returns”, European Journal of Operational Research 193 (2009) 23–34
[12] Lixin Tang, Yufang Zhao, “Scheduling a single semi-continuous batching machine”, Omega 36 (2008) 992 – 1004
[13] Gur Mosheiov, Assaf Sarig, “Due-date assignment on uniform machines”, European Journal of Operational Research 193 (2009) 49–58
[14] Dirk Biskup, Jan Herrmann, “Single-machine scheduling against due dates with past-sequence-dependent setup times”, European Journal of Operational Research 191 (2008) 587–592
[15] Ruslan Sadykov, “A branch-and-check algorithm for minimizing the weighted number of late jobs on a single machine with release dates”, European Journal of Operational Research 189 (2008) 1284-1304