OPEN ACCESS

International
Journal
Of Modern Engineering Research (IJMER)

Job Shop Scheduling Using Mixed Integer Programming

M. S. Al-Ashhab[1,2*],Shadi Munshi[1],Mowffaq Oreijah[1],Hamza A. Ghulman[1]
[1]*Dept. Of Mechanical Engineering, CollegeOf Engineering AndIslamic Architecture, Umm Al-Qura University, Makkah, Saudi Arabia.*
[2]*Design &Production Engineering Dept. Faculty OfEngineering, Ain-Shams University– Egypt*

ABSTRACT: *In this study, four different models in terms of mixed integer programming (MIP) are formulated for fourdifferent objectives. The first model objective is to minimizethemaximum finishing time (Makespan) without considering the products' due dates, while the second model is formulated to minimize the makespan considering the due dates for all the products, the third model is to minimize the total earliness time, and the fourth one is to minimize the total lateness time. The proposed models are solved, and their computational performance levels are compared based on parameters such as makespan, machine utilization, and time efficiency. The results are discussed to determine the best suitable formulation.*
Keywords:*Constraint programming; Job shop; Mixed integer programming.*

I. INTRODUCTION

The job shop scheduling has many practical applications in production and manufacturing industriesin addition to service-based industries such as transportation systems and hospitals where some tasks need to be scheduledin a certain order.Early studies addressed single machine scheduling as well as multi-machine scheduling. The first attempt addressed the flow shop problem to minimize the makespan with three machines, was reported by Wagner [1].Another study Manne[2] focused on the job shop problem with the aim of minimizing the makespan.The huge number of variables and solutions are the features of mathematical programming models which may lead to the failure to obtain the optimum as an exponential function of time. In the last three decades, tosolvethis problem, the combinatorial framework have been developed. López-Ortiz[3] called this frameworkconstraint programming (CP), and it is now extensively used for solving big scheduling problems. In general, themain features of CP are: (1) the ability to handle heterogeneous constraints, multiple disjunctions, and non-convex solutionspaces; (2) it is independent of the problem domain size; and (3) it allows the use of an optimization programming language (OPL).

In contrast to job shops with the makespan objective, the solution procedures for the job scheduling - total weighted tardiness (JS-TWT) are very limited. It is consideredthat it is convenient to divide previous JS-TWT research into three categories: exact methods, heuristics, and theoretical methods. For example, the branch and bound algorithm developed by Singer, M., &Pinedo, M. [4] belongs to the first category.Heuristics are also divided into the local search approach which are used by Essafi et al. [5] and Mati, Y. et al. andthe shifting bottleneck procedure that developedby Pinedo, M., & Singer, M. [7]. Gromicho, J. A. et al. [8] used the theoretical approaches such as dynamic programming algorithms, as well as the first category, may be more appropriate for obtaining deep insights into the structural properties and complexity of the JS-TWT.In just-in-time (JIT) scheduling, the common objective is to minimize a cost function which includes a penalty for both the early and tardy completion of jobs. A frequently used cost function sums the penalties due to earliness and tardiness, and the resulting problem is usually referred to as the earliness–tardiness scheduling problem.

In a Just-In-Time (JIT) system, a job should not be completed until just before its committed shipping date to avoid additional inventory and handling costs. The objective of minimization of the sum of the earliness'sof all jobs helps in fulfill JIT requirements.In this research, four models are developed; the first model aims to minimize the maximum completion time or make span without considering due dates for the jobs, the second model objective is the minimize the make span considering due dates for all jobs, while the third model is developed to minimize the total earliness of all jobs to accomplish the Just-In-Time (JIT) manufacturing, and the fourth one objective is tominimize the total latenesstimeof all jobs to reduce lateness penalties.

The following assumptions are considered in the study:

1) All jobs and machines are available at time zero.
2) The processing times of all operations are known.

3) Set-up time for any operation is included in the processing time.
4) The transportation time required for the movement for the job between the machines is assumed to be negligible.
5) Preemption of job is not allowed.

II. ModelsFormulation

In this paper, the following notations are used to develop the MIP formulations.

Parameters:
N: number of jobs (N: jobs set);
M: number of machines (M: machines set);
Phj: processing time for job j on machine h;
Dj: due date of job j;
NUMT: no. of machines (tasks) for each job
SEQ: processing sequince i.e, SEQ: [1, 3, 0] means that job 1 seq 1 - 3
NUMJ: no. of jobs per each machine.
DISJ: disjunction i.e, DISJ: [2, 3, 0] means that Machine 1 process onlyjobs 2 and 3.

Decision variables:
Cj: completion time of job j;
Shj: starting time of job j on machine h (a continuous non-negative variable);
Ej:earliness of job j = (Dj - Cj) if Dj>Cj and 0 otherwise;
Lj:lateness of job j = (Cj - Dj) if Cj > Dj and 0 otherwise;

2.1. Model Formulation

The formulations of the four models; objective functions and constraints are presented in the following sections.

2.1.1 Objective Function

2.1.1.1 Model 1 Objective

The objective of this model is to minimize the maximum completion time or the make span as shown in Equation (1) without considering due dates of all jobsand it is formulated as follows:

Minimize Cmax (1)

2.1.1.2 Model 2 Objective

The objective of this model is the same of model 1 but it is considering due dates of all jobs and it is formulated as follows:

2.1.1.3Model 3 Objective

The objective of this model is to minimize the total earliness time and it is formulated in Equation 2.

$$\text{Minimize} \quad \sum_{j \in N} Ej \tag{2}$$

2.1.1.4 Model 4 Objective

The objective of the fourth model is to minimize the total lateness time and it is formulated in Equation 3.

$$\text{Minimize} \quad \sum_{j \in N} Lj \tag{3}$$

2.1.2 Model Constraints

The constraints are of two types: the so-called conjunctive constraints represent the precedence between the operations for a single job type, and the disjunctive constraints express the fact that a machine can only execute a single operation at a time.

a) Disjunction Constraints

The disjunctive constraints can be written as follows:

$$\left(S_{hi} - S_{hj}\right) \geq P_{hj} - MY_{hij} \ , \forall i, j \in N, \forall h \in M \tag{4}$$
$$\left(S_{hi} - S_{hj}\right) \geq P_{hj} - M(1 - Y_{hij}) \ , \forall i, j \in N, \forall h \in M \tag{5}$$

This formulation technique employs disjunctive constraints, which utilize big-M and binary variables to decide the best ordering of tasks at each disjunction.

One of these two mutually exclusive constraints (4) or (5) must be relaxed when job *I* precedes job *j* and *j* precedes *i* on machine *k*.Where*M* is a large positive number and Yhij is a binary variable that takes the value 1if job*i* comes before job *j* on machine*h*, and 0 otherwise.

b) Conjunctive (precedence) Constraints

The processing sequence or operational precedence between the tasks should also be satisfied. In particular, it is necessary to indicate that the processing of operation *l*+1 for job *j* on machine *h* must be started after the completion of operation*l*. This feature can be characterized as follows:

$$\sum_{h \in M} \left(S_{SEQ(j,l),j} + P_{SEQ(j,l),j} \right) \geq \sum_{h \in M} S_{SEQ(j,l+1),j} \ , \forall j \in N, \forall l \in M-1 \qquad (6)$$

The model is solved using Xpress-MP 7.9 software on an Intel® Core™ i3-2310M CPU @2.10 GHz (3 GB of RAM) [9].

III. COMPUTATIONAL RESULTS AND ANALYSIS

In this section the results of the fourmodels are presented and analysed. The processing sequences of the five jobs for the four models are shown in Table 1.The durationsof the five jobs processes for the four models are shown in Table 2. And, the due dates of all jobs for the second and the third model are given in Table 3.

Table 1. Processing sequience of the five jobs in the four machines

	Job 1	Job2	Job3	Job 4	Job 5
M/C 1	1	1	0	4	1
M/C 2	0	2	3	1	0
M/C 3	2	3	2	2	2
M/C 4	3	4	1	3	0

Table 2. Duration matrix of the five jobs in the four machines

	Job 1	Job2	Job3	Job 4	Job 5
M/C 1	19	10	0	14	15
M/C 2	0	30	15	10	0
M/C 3	10	18	18	20	16
M/C 4	19	11	31	19	0

Table 3. Due Date Matrixof the five jobs

	Job 1	Job 2	Job 3	Job 4	Job 5
Due date	100	115	90	85	31

3.1 Case 1 Results (Make Span Minimization)

The resulted schedule of the first model are as shown in Table 4. The schedule is drawn as a Gantt chart in Figure 1.

Table 4. The resulted production schedule of the first model.

	Job1	Job2	Job3	Job 4	Job 5
M/C 1	10-29	0-10		78-92	29-44
M/C 2		10-40	58-73	0-10	
M/C 3	30-40	58-76	40-58	10-30	76-92
M/C 4	50-69	78-89	0-31	59-78	
Makespan					*92*

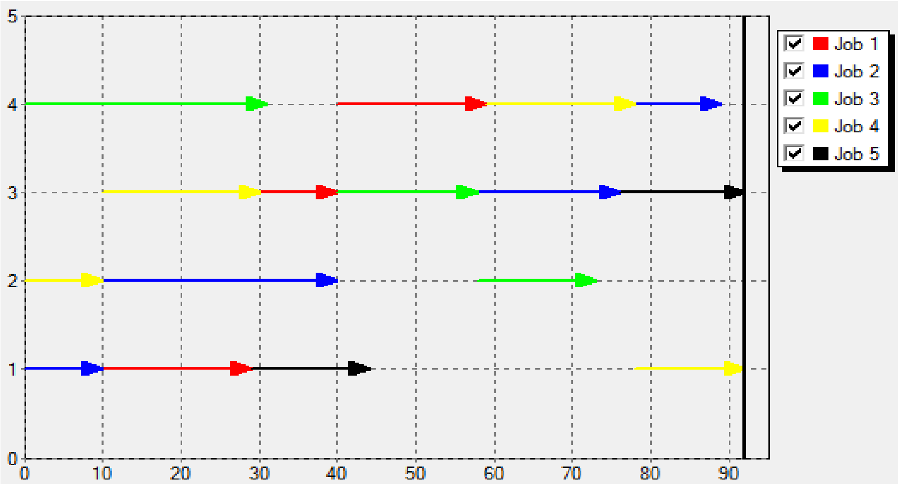

Figure 1. The resulted Gantt chart of the first model.

The resulted starting, idle time, and machines utilizations of the first model are shown in Table 5. Also, the summation of the starting and idle times and overall utilization are calculated and presented in the same table.

Table 5. The resulted starting, idle time, and machines utilizations of the first model.

	Starting Time	Idle Time	Utilization %
M/C 1	0	34	63.04
M/C 2	0	37	59.78
M/C 3	10	10	89.13
M/C 4	0	12	86.96
Overall	10	93	74.72

3.2 Case 2 Results (Make Span Minimization with Due Date Constraints)

The resulted schedule of the second model are as shown in Table 6. The schedule is drawn as a Gantt chart in Figure 2.

Table 6. The resulted production schedule of the second model.

	Job1	Job2	Job3	Job 4	Job 5
M/C 1	25 - 44	15 - 25		70 - 84	0–15
M/C 2		41 -71	75-90	0 - 10	
M/C 3	69-79	80 - 98	51-69	31-51	15–31
M/C 4	79-98	98-109	20- 51	51-70	
Makespan					109

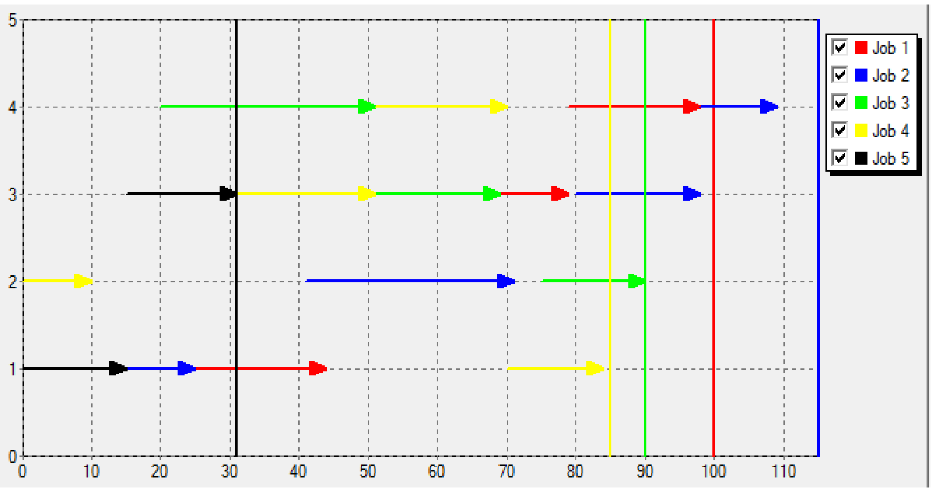

Figure 2. The resulted Gantt chart of the second model.

The resulted starting, idle time, and machines utilizations of the second model are shown in Table 7. Also, the summation of the starting and idle times and overall utilization are calculated and presented in the same table.

Table 7. The resulted starting, idle time, and machines utilizations of the second model.

	Starting Time	Idle Time	Utilization %
M/C 1	0	51	53.21
M/C 2	0	54	50.45
M/C 3	15	27	75.22
M/C 4	20	29	73.39
Total	35	161	63.06

The job finishing order of the first model as shown in Figure 1 is 1-3-2-4-5 while the job finishing order of the second model as shown in Figure 2 is 5-4-3-1-2. Figure 2 shows that the make span increased to 109 instead of 92 in Figure 1 because of considering the due date of the job that effected the machines loading schedule to satisfy the due date constraints. Since job number 5 has the least due date of 31, it took it as first priority and has loaded it before any other job while in the first model which concentrated only on the minimization of the make span, gave it the last priority. Furthermore, tables 5 and 7 show an increase in machine idle time due to the due date constraint.

3.3 Case 3 Results (Earliness Minimization)

The resulted schedule of the third model are as shown in Table 8. The schedule is drawn as a Gantt chart in Figure 3.

Table 8. The resulted production schedule of the third model.

	Job1	Job2	Job3	Job 4	Job 5
M/C 1	15-34	32 -45		71-85	0-15
M/C 2		45-75	75-90	0-10	
M/C 3	71-81	81-91	51-69	31-51	15-31
M/C 4	81-100	104-115	0-31	51-70	
Makespan					115

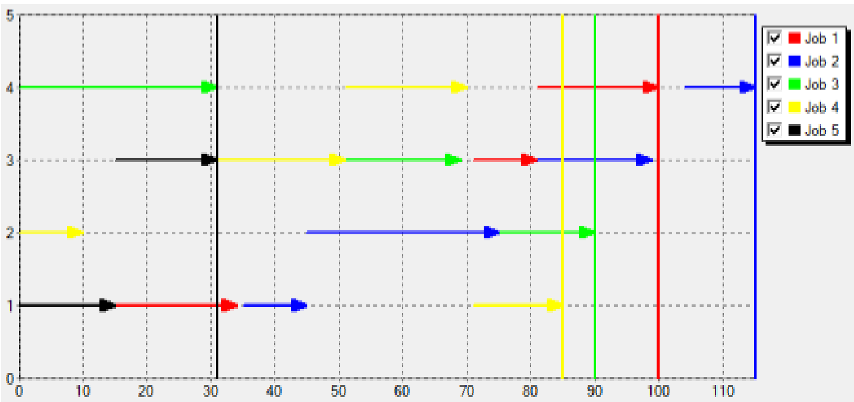

Figure 3. The resulted Gantt chart of the third model.

The resulted starting, idle time, and machines utilizations of the third model are shown in Table 9. Also, the summation of the starting and idle times and overall utilization are calculated and presented in the same table.

Table 9. The resulted starting, idle time, and machines utilizations of the third model.

	Starting Time	Idle Time	Utilization %
M/C 1	0	57	50.43
M/C 2	0	60	47.82
M/C 3	15	33	71.30
M/C 4	0	35	69.56
Overall	15	185	59.7

Adding a second constraint of finishing just in time (earliness minimization)at a due date incsreased the make span from 92 in case 1 to 109 in case 2 to 115 in case 3with a possibility of having all machines idle at some periods of time due to earliness minimization constraint. Furthermore, the idle time has increased from 93 without consraints to 185 lowering the utilization percentage from 74.72 % to 59.7 %.

3.4 Case 4Results (Total Lateness Minimization)

The resulted schedule of the fourth model are as shown in Table 10. The schedule is drawn as a Gantt chart in Figure 4.

Table 10. The resulted production schedule of the fourth model.

	Job1	Job2	Job3	Job 4	Job 5
M/C 1	**50 – 69**	**15 - 25**		**71 – 85**	**0 - 15**
M/C 2		**25 – 55**	**75 – 90**	**0 – 10**	
M/C 3	**69 – 79**	**97 – 97**	**51 – 69**	**31 - 51**	**15 - 31**
M/C 4	**81 - 100**	**104 - 115**	**104 – 115**	**51 - 70**	
	Makespan				***115***

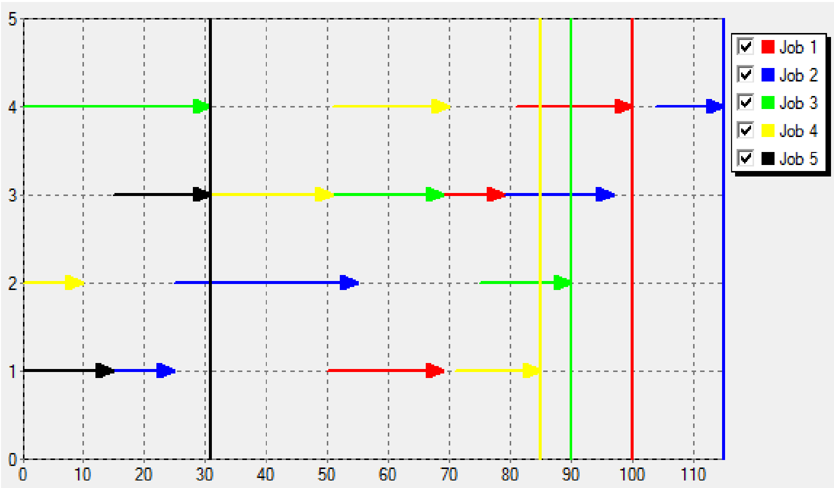

Figure 4. The resulted Gantt chart of the fourth model.

The resulted starting, idle time, and machines utilizations of the fourth model are shown in Table 11. Also, the summation of the starting and idle times and overall utilization are calculated and presented in the same table.

Table 11. The resulted starting, idle time, and machines utilizations of the fourth model.

	Starting Time	Idle Time	Utilization %
M/C 1	**0**	**57**	**50.43**
M/C 2	**0**	**60**	**47.82**
M/C 3	**15**	**33**	**71.30**
M/C 4	**0**	**35**	**69.7**
Total	**15**	**185**	**59.7**

Case 4 displays the minimization of the total lateness time to reduce any lateness penalties. Figure 4 shows no lateness while maintaining the due date of all jobs. In spite of some changes in the scheduling between the third and the forth cases, make span, machine idle time and utilization percentage remain the same as case3.

IV. CONCLUSION

Scheduling tasks, especially when there are many, is a tedious job and requires programming with optimization techniques to achieve the required results with minimal costs.

In this study,four modules have been successfully developed and applied as examples to different requirements for different applications. The first model ofminimizing the maximum completion time (make span) can be observed in applications using make to stock strategies. Other applications require constraints as in the second model of minimizing the make span considering due dates for all jobs. These can be observed in make to order strategies. The third model introduced a different constraint of minimizing the total earliness of all jobs to accomplish the Just-In-Time (JIT) manufacturing. The fourth model displayed the minimization of the total lateness time of all jobs to reduce lateness penalties. Catering is a good example of such applications for the last twomodels.

Observing the results, the application of job shop scheduling is clearly affected by the required objective. Having the least constraints gives the highest utilization of the machines and gives the lowest make span. This study has successfully illustrated the ability of optimizing job shop scheduling with different constraints and objectives.

REFERENCES

[1]. Wagner, H. M. (1959). An integer linear-programming model for machine scheduling. Naval Research Logistics (NRL), 6(2), 131-140S.A.
[2]. Manne, On the job shop scheduling problem, Oper. Res. 8 (2) (1960) 219–223.
[3]. López-Ortiz, A., Quimper, C. G., Tromp, J., & Van Beek, P. (2003, August). A fast and simple algorithm for bounds consistency of the alldifferent constraint. In IJCAI (Vol. 3, pp. 245-250).
[4]. Singer, M., &Pinedo, M. (1998). A computational study of branch and bound techniques for minimizing the total weighted tardiness in job shops. IIE transactions, 30(2), 109-118.
[5]. Essafi, I., Mati, Y., &Dauzère-Pérès, S. (2008). A genetic local search algorithm for minimizing total weighted tardiness in the job-shop scheduling problem. Computers & Operations Research, 35(8), 2599-2616.
[6]. Mati, Y., Dauzère-Pérès, S., &Lahlou, C. (2011). A general approach for optimizing regular criteria in the job-shop scheduling problem. European Journal of Operational Research, 212(1), 33-42.
[7]. Pinedo, M., & Singer, M. (1999). A shifting bottleneck heuristic for minimizing the total weighted tardiness in a job shop. Naval Research Logistics, 46(1), 1-17.
[8]. Gromicho, J. A., Van Hoorn, J. J., Saldanha-da-Gama, F., &Timmer, G. T. (2012). Solving the job-shop scheduling problem optimally by dynamic programming. Computers & Operations Research, 39(12), 2968-2977.
[9]. www.Fico.com