Contemporary Engineering Sciences, Vol. 9, 2016, no. 7, 315 - 324
HIKARI Ltd, www.m-hikari.com
http://dx.doi.org/10.12988/ces.2016.617

A Multi Due Date Batch Scheduling Model on Dynamic Flow Shop to Minimize Total Production Cost

Tengku Nurainun*

Department of Industrial Engineering, Universitas Islam Negeri Sultan Syarif Kasim (UIN Suska) 28293, Pekanbaru, Riau Indonesia
*Corresponding author

Ahmad Fudholi

Solar Energy Research Institute, Universiti Kebangsaan Malaysia
43600 Bangi Selangor, Malaysia

Misra Hartati

Department of Industrial Engineering, Universitas Islam Negeri Sultan Syarif Kasim (UIN Suska) 28293, Pekanbaru, Riau Indonesia

Rado Yendra

Department of Mathematics, Faculty of Science and Technology
Universitas Islam Negeri Sultan Syarif Kasim (UIN Suska)
28293, Pekanbaru, Riau Indonesia

Ismu Kusumanto

Department of Industrial Engineering, Universitas Islam Negeri Sultan Syarif Kasim (UIN Suska) 28293, Pekanbaru, Riau Indonesia

Abstract

This research solved a batch scheduling problem on dynamic flow shop environment, in which new orders are considered to insert in the current work. Decision making is made by comparing the total cost of the insertion schedule with the original schedule. Another feature of this model is its ability to accommodate multi product orders that are delivered on different due dates in which revised due date is allowed for each order. The proposed model is known as Mix Integer Non Linear Programming (MINLP) with a convex objective function. An algorithm is developed to find the best solutions of batch size and sequence that minimize total cost. A numerical experience is performed to show how the algorithm works.

Keywords: Batch scheduling, dynamic flow shop, multi due dates, multi products, total cost

1 Introduction

Batch scheduling has proven as an effective way to increase the manufacturing performance. Many literatures have dealt that combination of batch size, and sequences are reliable in minimizing cost or time length production. Nowadays, uncertain environment imposes a company to the response all possibilities of changing rapidly. Disturbances might be caused by various factors such as disrupted machine, arrival of new orders, cancellation of current orders, updating due dates, and others. Scheduling research has mostly focused on inventing a model or algorithm to solve deterministic problems. In recent decades, further research has been developed to solve dynamic and uncertain environments [1-5].

In most real production environments, jobs arrive to the system randomly, and the job's arrival and release dates are not known in advance. A simple approach of this condition is inserting the new jobs into the scheduled sequence. Several works have been done to arrange the latest order to the executed schedule by designing the insertion technique [6, 7]. This situation makes a deviation from original schedule and potentially increases the total of production cost. Common resource that is used for different jobs also increases the complexity of problem and becomes more difficult to figure it.

In fact, completing all jobs on their due dates has become a complex issue in a way of increasing service level to the customer. Further, in dynamic situation, both WIP and new orders need to meet due date. Each job has its release date that is sorely important to be accomplished. Very little work is reported on the cost effect of new order's arrival [8, 9]. Earlier completion time arises consequence to the holding cost, in contrast, later completion time effects at the penalty cost. Hence, trade off among various costs are needed in making decision, whether continuing the current schedule or adjusting the original schedule with all dynamic situations. This paper is aimed to develop a more flexible model and technique to accommodate new orders with acceptable cost consequence. The model has considered setup cost, WIP cost, inventory cost, and lost sale cost. All jobs are processed on the same routing but might be different in specification (multi item). Each job has its due date (multi due date) and possible to revise it. An adaptive job-insertion based heuristic is presented to determine batch size and schedule the resulting batches with the objective to minimize total production cost.

2 Total production cost

There are four elements of production cost that are considered in this model, they are set up cost, work-in-process (WIP) cost, inventory cost, and lost sale cost. Suppose that there are n jobs of item g to be process on m stage, each stage consists of single machine. Jobs are batched into N batches with batch size Q_i (i = 1, 2, ..., N). All jobs that are processed in the same batch only need a single setup before processing. The setup time on a machine is denoted as s_m, and setup cost of a machine is denoted as $C_{1,m}$, then setup cost is formulated by multiplying setup time with setup cost. Thus, total setup cost can be written as follows:

$$TC_{1,m} = \left(N \cdot \sum_{m=1}^{M} s_m \cdot C_{1,m} \right) \quad (1)$$

Intermediate processing is defined as time length of a batch to be processed. If $F_{i,m}$ is the finishing time of a batch i on machine m, and $B_{i,1}$ is the starting time of a batch i on machine 1, then WIP is calculated as a deviation between the finishing time and the starting time. Minimizing WIP cost will also minimize the time length production simultaneously. If Intermediate cost of item g is denoted as $C_{2,g}$ then formulation of total WIP cost can be seen as follows:

$$TC_{2,g} = \sum_{g=1}^{G} \sum_{i=1}^{N} r_{i,g} \left[(F_{i,m} - B_{i,1}) Q_i \cdot C_{2,g} \right] \quad (2)$$

This model has considered about real time condition where each item has specific due date. If the shop capacity is tight, it is difficult to meet all due dates but earlier finished goods are permitted. This condition may incur an inventory cost until the products are released. If holding cost of item g is denoted as $C_{3,g}$, then we can calculate total finished goods inventory cost as follows:

$$TC_{3,g} = \sum_{g=1}^{G} \sum_{i=1}^{N} r_{i,g} \left[(d_g - F_{i,M}) Q_i \cdot C_{3,g} \right] \quad (3)$$

In this formulation, a job is not allowed to deliver after its due date. If jobs are impossible to be finished on their due date, then they will not be processed on the shop floor. It means the company will lose their opportunity to deliver the products and caused the lost sale cost as penalty. In case the arrival of a new job, model will compare inventory cost if the products are finished earlier, and lost sale cost if the products are refused to be process. Due date can be rearranged with the customer based on the feasibility of the model. It clearly assists the decision maker in dealing with the customer immediately. The result of this consideration is yes or no decision. It will involve a Binner variable, X_g, in which the variable value is 1 if the new order is accepted or the variable value is 0 if the new order is rejected. If lost sale cost of item g is denoted as $C_{4,g}$, then total lost sale cost is formulated as follows:

$$TC_{4,g} = \sum_{g=1}^{G} X_g \cdot C_{4,g} \tag{4}$$

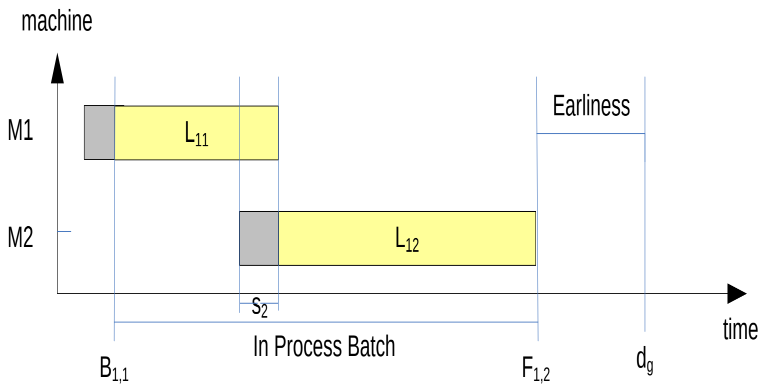

Fig. 1 Production cost

3 Problem formulations

The variables that are involved in this model formulation can be denoted as below:
$B_{i,m}$: Starting time for batch, which is sequenced in position *i* on machine *m*
$F_{i,m}$: Finishing time for batch, which is sequenced in position *i* on machine *m*
Q_i : Batch size which is sequenced in position *i*
N : The number of batches
$r_{i,g}$: Binner variable for item *g* within batch which is sequenced in position *i*
X_g : Binner variable for arrival of new order
TC : Total Production Cost
The parameters that are used in this proposed model can be defined as follows:
M : The number of machines
G : The number of item
n_g : The number of item *g* to be requested on d_g
d_g : Due date of item *g*
t_{gm} : The processing time for item *g* processed on machine *m*
s_m : The setup time required before any batch is processed on machine *m*
A_m : Availability of machine *m*
$C_{1,m}$: Setup cost for machine *m*
$C_{2,g}$: Intermediate cost for item *g*
$C_{3,g}$: Holding Cost of item *g*

There are *n* jobs of item *g* (*g* = 1, 2, ... *G*) that are requested to be processed on a *m* stages flow shop, each stage consist of a single machine, which is each item will be delivered on its due date, d_1, d_2, ..., d_G. All items have the same routing but different in specification, e.g. colour, size, etc., All variant items will be processed in batches,

and setup time is required before any batch enters a machine. The problem is determining batch sizes and sequences that minimize total production cost simultaneously. This model has accommodated arrival of new orders, so that a variable to check availability of machine is needed. The function of this variable will be clearly explained in proposed algorithm.
The model is developed under several assumptions:

- Holding cost of raw material is negligible.
- Setup time and setup cost for all batches that are processed on the same machine is fixed.
- The arrival of a new job is not permitted to interrupt the current work.
- None of the batches that is processed on more than one machine, and none of machines that run the process for more than one batch.

The objective of the developed model is to minimize total production cost. It consists of several cost elements, and we need the trade-off between the cost variables as the result. Thus, the formulation of total production cost is:

$$TC = \left(N \cdot \sum_{m=1}^{M} s_m \cdot C_{1,m} \right) + \sum_{g=1}^{G} \sum_{i=1}^{N} r_{i,g} \left[\left(F_{i,m} - B_{i,1} \right) Q_i \cdot C_{2,g} \right] + \sum_{g=1}^{G} \sum_{i=1}^{N} r_{i,g} \left[\left(d_g - F_{i,M} \right) Q_i \cdot C_{3,g} \right] + \sum_{g=1}^{G} X_g \cdot C_{4,g} \quad (5)$$

Where,

$$B_{1,1} = A_1 + s_1 \quad (6)$$

$$F_{1,1} = B_{1,1} + \sum_{g=1}^{G} r_{1,g} \left(t_{g,1} \cdot Q_1 \right) \quad (7)$$

$$B_{i,1} = F_{i-1,1} + s_1 \;;\; i = 2,\ldots,N \quad (8)$$

$$F_{i,1} = B_{i,1} + \sum_{g=1}^{G} r_{i,g} \left(t_{g,1} \cdot Q_i \right) \;;\; i = 2,\ldots,N \quad (9)$$

$$B_{1,m} = \max\left(F_{1,m-1}, A_m \right) \;;\; m = 2,\ldots,M \quad (10)$$

$$F_{1,m} = B_{1,m} + \sum_{g=1}^{G} r_{1,g} \left(t_{g,m} \cdot Q_1 \right) \;;\; m = 2,\ldots,M \quad (11)$$

$$B_{i,m} = \max\left(F_{i,m-1}, F_{i-1,m} + s_m \right) \;;\; i = 2,\ldots,N \;;\; m = 2,\ldots,M \quad (12)$$

$$F_{i,m} = B_{i,m} + \sum_{g=1}^{G} r_{i,g} \left(t_{g,m} \cdot Q_i \right) \;;\; i = 2,\ldots,N \;;\; m = 2,\ldots,M \quad (13)$$

$$F_{N,M} \le d_g \quad (14)$$

$$r_{i,g} \in \{0,1\} \;;\; i = 1,\ldots,N \;;\; g = 1,\ldots,G \quad (15)$$

$$\sum_{g=1}^{G} r_{i,g} = 1 \;;\; i = 1,\ldots,N \quad (16)$$

$$\sum_{g=1}^{G} r_{i,g} \cdot Q_i = n_g \;;\; i = 1,\ldots,N \quad (17)$$

$$X_g \begin{cases} 1, & \text{if } \sum_{i=1}^{N} r_{i,g} = 0 \;;\; g = 1,\ldots,G \\ 0, & \text{else} \end{cases} \quad (18)$$

$$N \ge G \quad (19)$$

$$Q_i \ge 1, \text{ integer} \quad (20)$$

Constraint (6) and (7) show the beginning and finishing time of the first batch on machine 1. Constraint (8) and (9) show the beginning and finishing time of batch *i* that is sequenced on machine 1. Constraint (10) and (11) show the beginning and finishing time of the first batch on machine m. Constraint (12) and (13) shows the beginning and finishing time of batch *i* that is sequenced on machine *m*. Constraint (14) to make sure that finishing time of all products is earlier or precisely on their due dates. Constraint (15) is a Binner variable to determine the particular item that is placed in batch where is sequenced on position *i*. Constraint (17) verifies the total of jobs that are processed in a batch equal with the order number. Constraint (18) is a Binner variable to determine the new order, whether accepted or rejected it. Constraint (19) shows that the number of batch is equal or more than the number of item. Constraint (20) is available to make sure that the number of a batch is more than 1 and integer.

4 Problem solution

This paper has considered about the arrival of new orders in the flow shop. An algorithm is proposed to get the solution of this problem. There are two algorithms developed for this problem. Firstly, we will introduce an algorithm to insert a new order to the existing schedule. Secondly, we will show an algorithm to split jobs into batches.

In general, we can explain the idea of the dynamic flow shop algorithm as below:

- Check the availability of machines to perform the new order by re-count the number of jobs and remaining time needed for unfinished works and determine the most possible point of time period to begin the arrival order.
- Group the new work that is identical with the current works and calculate the total number of revised order in a group.
- Re-schedule.

Further, these are the details of the proposed algorithm to solve the problem:

a. Inserting New Order Algorithm

Step 1. Identify the current works that are processing on the shop floor.

Step 2. Set the finishing time of current works on all machines as the availability of machine (A_m).

Step 3. Check the remaining works that have not been processed yet and group them with the similar new orders.

Step 4. Split the group based on due dates. If a group has several different due dates, re-group them into *g* due dates.

Step 5. Set the value of predetermined parameters that are needed in calculation.

Step 6. Set the number of batch equal with the number of item.

Step 7. Solve the problem using the proposed model.

Step 8. Split the batch using Sub Algorithm for Batching and Sequencing.

Step 9. Use the result as revised schedule.

Step 10. Check the status of all new orders. If entire works are rejected, then back to the initial schedule. If at least one of the new orders is accepted, then use the revised schedule.

Step 11. Finish.

b. Sub Algorithm for Batching and Sequencing

Step 1. Set $N = G + 1$.

Step 2. Solve the problem of batching and sequencing using the formulation.

Step 3. Note the result of total production cost.

Step 4. Check feasibility of schedule based on due date of each order. Stop searching if it is not feasible, then go to Step 7. Otherwise, go to Step 5.

Step 5. Check the number of batch. If it has reached the number of item, then go to Step 7. Otherwise, go to Step 6.
Step 6. Set $N = N + 1$. Back to Step 2.
Step 7. Use the schedule with minimum total production cost as the result.
Step 8. Finish.

5 Numerical Experience

There are two kinds of products that will be processed on three stages flow shop, product *A* and product *B*. The values of the parameters as shown in Table 1 and Table 2. This situation presented deterministic circumstance without disturbances that may interrupt the process. The result of this problem is stated as the original schedule, can be seen as shown in Table 3.

Table 1 Parameters for machines

m	t_{gm}		s_m	$C_{1,m}$
	A	B		
1	4	6	2	2
2	10	12	8	3
3	8	10	4	2

Table 2 Parameters for products

Product	n_g	d_g	$C_{2,g}$	$C_{3,g}$
A	5	800	5	1
B	10	1000	8	4

Table 3 The resulting batch for static environment

N	TC	N	TC
2	56.502	9	48.350
3	52.058	10	48.258
4	50.672	11	48.198
5	49.548	12	48.170
6	48.960	13	48.158
7	48.564	14	48.260
8	48.456	15	48.390

The solution shows that the relation between total production cost and batch size is convex. Splitting batch into small size will reduce the total production cost until the minimum point is reached and after that the number of batch will increase the total production cost, as shown in Fig.2. The original schedule of this problem is performed as shown in Table 4.

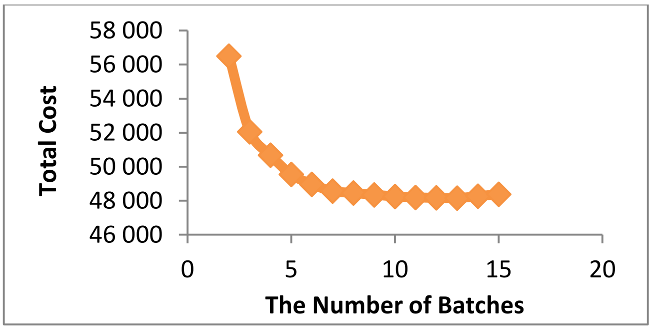

Fig. 2 Total production cost for different batch size

Table 4 The original schedule

N	*Q*	*Item*	*Starting Point*		*F*
			M	*B*	
1	*1*	*A*	*1*	*2*	*6*
			2	*8*	*18*
			3	*18*	*26*
2	*2*	*A*	*1*	*8*	*16*
			2	*26*	*46*
			3	*46*	*62*
3	*2*	*A*	*1*	*18*	*26*
			2	*54*	*74*
			3	*74*	*90*
4	*1*	*B*	*1*	*28*	*34*
			2	*82*	*94*
			3	*94*	*104*
5	*1*	*B*	*1*	*36*	*42*
			2	*102*	*114*
			3	*114*	*124*
6	*1*	*B*	*1*	*44*	*50*
			2	*122*	*134*
			3	*134*	*144*

…continued

7	*1*	*B*	*1*	*52*	*58*
			2	*142*	*154*
			3	*154*	*164*
8	*1*	*B*	*1*	*60*	*66*
			2	*162*	*174*
			3	*174*	*184*
9	*1*	*B*	*1*	*68*	*74*
			2	*182*	*194*
			3	*194*	*204*
10	*1*	*B*	*1*	*76*	*82*
			2	*202*	*214*
			3	*214*	*224*
11	*1*	*B*	*1*	*84*	*90*
			2	*222*	*234*
			3	*234*	*244*
12	*1*	*B*	*1*	*92*	*98*
			2	*242*	*254*
			3	*254*	*264*
13	*1*	*B*	*1*	*100*	*106*
			2	*262*	*274*
			3	*274*	*284*

For dynamic situation, suppose that there is an order of eight units item *B* arrives at 154 of time period and needed to deliver on 450 of the time period. The proposed algorithm will be used to solve this situation.

Step 1. The current work that are processing on the shop floor at minute 154 is Q_7 on the machine-3.

Step 2. The availabilities of machine are A_1 = 58, A_2 = 154, and A_3 = 164.

Step 3. The remaining work that have not been finished to process is item *B*, similar type with the new order.

Step 4. Due date of the new order is tighter than the current work. Then, to differ it, the new order will be labelled as item *C*.

Step 5. Set the value of predetermined parameters that are needed in calculation.

Step 6. Set $N = 2$

Step 7. The resulting value of total production cost for N = 2 are showed in Table 5.

Step 8. The resulting batch is presented on Table 5.

Step 9. The resulting schedule is presented on Table 6.

Step 10. Check the status of all new orders. If entire works are rejected, then back to the initial schedule. If at least one of the new orders is accepted, then use the revised schedule.

Step 11. Finish.

The graph to show the convex objective function for dynamic flow shop environment as shown in Fig.3.

Table 5 The resulting batch for dynamic environment

N	*TC*	*N*	*TC*
2	45.720	9	42.964
3	44.316	10	42.984
4	43.600	11	43.036
5	43.268	12	43.120
6	43.096	13	43.236
7	43.024	14	43.384
8	43.020		

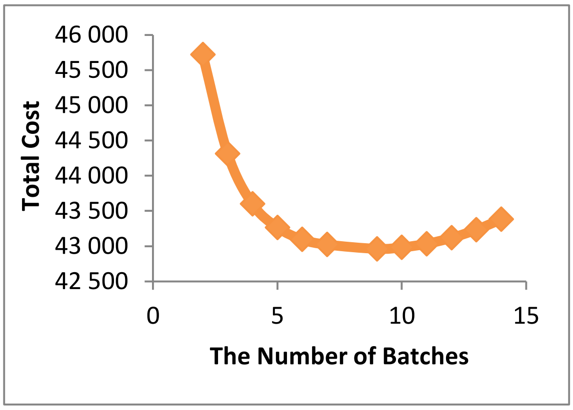

Fig.3 Total production cost for dynamic environment

Table 6. The revised schedule

N	Q_i	*Item*	*Starting Point*		$F_{i,m}$
			m	$B_{i,m}$	
1	2	C	1	60	72
			2	162	186
			3	186	206
2	2	B	1	74	86
			2	194	218
			3	218	238
3	2	C	1	88	100
			2	226	250
			3	250	270
4	2	C	1	102	114
			2	258	282
			3	282	302
5	2	B	1	116	128
			2	290	314
			3	314	334
6	1	B	1	130	136
			2	322	334
			3	338	348
7	1	C	1	138	144
			2	342	354
			3	354	364
8	1	B	1	146	152
			2	362	374
			3	374	384
9	1	C	1	154	160
			2	382	394
			3	394	404

6 Conclusions

This paper addresses a multi due date batch scheduling model on dynamic circumstance for similar steps of the production process to minimize total production cost. There are several cost elements that is considered in the proposed model, they are setup cost, work-in-process (WIP) cost, inventory cost, and lost sale cost. Numerical experience shows that the proposed model and algorithm are reliable to solve the dynamic scheduling problem for flow shop manufacturing. Future research as development of this model and algorithm is needed to solve the more complex situation, e.g. involving penalty cost for the late finished jobs, considering another disturbance situation like unavailability of machines, etc.

References

[1] D. Ouelhadj and S. Petrovic, A Survey of dynamic scheduling in manufacturing systems, *Journal of Scheduling*, **12** (2009), no. 4, 417-431. http://dx.doi.org/10.1007/s10951-008-0090-8

[2] M. Gholami, M. Zandieh and A.A. Tabriz, Scheduling hybrid flow shop with sequence-dependent setup times and machines with random breakdowns, *The International Journal of Advanced Manufacturing Technology* **42** (2009), no. 1-2, 189-201. http://dx.doi.org/10.1007/s00170-008-1577-3

[3] L. Tang, W. Liu and J. Liu, A neural network model and algorithm for the hybrid flow shop scheduling problem in a dynamic environment, *Journal of Intelligent Manufacturing* **16** (2005), no. 3, 361-370. http://dx.doi.org/10.1007/s10845-005-7029-0

[4] S.Q. Liu, H.L. Ong and K.M. Ng, Metaheuristics for minimizing the makespan of the dynamic shop scheduling problem, *Advances in Engineering Software*, **36** (2005), no. 3, 199-205. http://dx.doi.org/10.1016/j.advengsoft.2004.10.002

[5] R. Aggoune, Minimizing the makespan for the flow shop scheduling problem with availability constraints, *European Journal of Operational Research*, **153** (2004), no. 3, 534-543. http://dx.doi.org/10.1016/S0377-2217(03)00261-3

[6] G. Guo, B. Wu and S. Yang, A job-insertion heuristic for minimizing the mean flowtime in dynamic flowshops, *Front. Mech. Eng*., **6** (2011), no. 2, 197-202. http://dx.doi.org/10.1007/s11465-011-0211-5

[7] E. Lodree, W. Jang and C.M. Klein, A new rule for minimizing the number of tardy jobs in dynamic flow shops, *European Journal of Operational Research*, **159** (2004), no. 1, 258-263. http://dx.doi.org/10.1016/S0377-2217(03)00404-1

[8] K. Bulbul, P. Kaminsky and C. Yano, Flow shop scheduling with earliness, tardiness, and intermediate inventory holding costs, *Naval Research Logistics (NRL)*, **51** (2004), 407-445. http://dx.doi.org/10.1002/nav.20000

[9] A.H. Halim and H. Ohta, Batch-scheduling problems to minimize inventory cost in the shop with both receiving and delivery just-in-times, *International Journal of Production Economics*, **33** (1994), 185-194. http://dx.doi.org/10.1016/0925-5273(94)90131-7

Received: February 12, 2016; Published: March 24, 2016