A New Lower Bound for Flexible Flow Shop Problem with Unrelated Parallel Machines

N. Nahavandi[*] & E. Asadi Gangraj

Nasim Nahavandi, *Associate Professor of Industrial Engineering, Department of Industrial Engineering, Tarbiat Modares University, Tehran, Iran,*
Ebrahim asadi Gangraj, *PhD student, Department of Industrial Engineering, Faculty of Engineering, Tarbiat Modares University, Tehran, Iran.*

KEYWORDS

Lower bound;
Flexible flow shop;
Makespan;
Unrelated parallel machines

ABSTRACT

Flexible flow shop scheduling problem (FFS) with unrelated parallel machines contains sequencing in flow shop where, at any stage, there exists one or more processors. The objective consists of minimizing the maximum completion time. Because of NP-completeness of FFS problem, it is necessary to use heuristics method to address problems of moderate to large scale problem. Therefore, for assessment the quality of this heuristic, this paper develop a global lower bound on FFS makespan problems with unrelated parallel machines.

1. Introduction

The scheduling of n jobs through m stages where, at any stage, there exist one or more unrelated processors, is termed as flexible flow shop (FFS) scheduling problem with unrelated parallel machines.

In the classical flow shop problem, a set of jobs flow through multiple stages in the same machine order, where each stage consists of only one machine [1]. But, many production companies need to enhance or balance the capacity, and then it has lead to append some machines to some stages.

This new problem is known as flexible flow shop (FFS), flexible flow line (FFL), hybrid flow shop (HFS), or a flow shop with multiple processors (FSMP). The FFS exists in many real world manufacturing problems, such as semiconductor assembly facilities [2], packaging industries [3], steal manufacturing [4], electronics manufacturing [5], glass container fabrication [6], automobile assembly [7], printed circuit board assembly [8,9], printed circuit board fabrication [10], ceramic tile manufacturing [11], and lead frame manufacturing [12].

Some authors considered this classical problem such as, Santos et al. [13], Sawik [14], Guinet and Solomon [15], Leon and Ramamoorthy [6], Kadipasaoglu et al. [16] , Alisantoso et al. [10], Lee et al. [12], Botta-Genoulaz [17], Cheng et al. [18], Quadt and Kuhn [19], Torabi et al. [20].

In FFS with unrelated machines, the processing times of a job in a stage are different and depend on each specific machine. This may be due to the differences between the machines, to the fact that certain types of machines are better suited for a particular job, whereas others are not, or because the jobs have some special characteristics and can only be assigned to machines that are physically near to them [1]. Some authors consider this characteristic in their research such as Kadipasaoglu et al. [16,21], suresh [22], Hayrinen et al. [9], Low [23], Sawik [14], Jenabi et al. [25], Low et al. [26], and He et al. [27].

One of the effective tools for estimating the optimal makespan for evaluating the quality of heuristics methods is the determination of a strong lower bound. Santos and Deal [28] proposed a global lower bound for Flow shop with multiple processors. The selective objective function was makespan. The procedure for developing a global bound involves determining a lower bound for each stage. This stage-based lower bound calculates for each stage and the greatest stage-

based bound is the bound which can be used for the entire problem.

Haouari and M'Hallah [29] developed a new lower bound for makespan in two-stage Hybrid Flowshop environment. They also compared the lower bound with two phased method based on Simulated Annealing and Tabu Search. These comparisons show the superiority of the derived lower bound.

Soewandy and Elmaghrby [30] developed some lower bound for three-stage FFS problem. Firstly, they compute an auxiliary problem from the original problem.

Based on the processing time of auxiliary problem, they compute some lower bounds for makespan of auxiliary problem and any lower bound to the optimum of auxiliary problem is necessarily a makespan lower bound for original problem.

In this paper, we consider FFS scheduling problem. Despite of many lower bounds for identical FFS scheduling problem, there isn't any lower bound for makespan in flexible flow shop scheduling problem with unrelated parallel processors. In this research a new stage-based lower bound for FFS problem.

The remainder of this paper is organized as follows. Section 2 describes mathematical model for FFS scheduling problem with unrelated parallel processors. Section 3 is dedicated to lower bound for FFS scheduling problem with unrelated processors. In section 4, an experimental study is presented to evaluate the lower bound according to some experimental factors. Finally, section 5 is devoted to the main finding of this paper and suggestions for conducting some future researches.

2. Mathematical Model

This section presents a mathematical model for FFS scheduling problem that considers relation between jobs processed in two consecutive stages and machines in each stages. The selected objective function is makespan. The model is based on the following hypothesis:

- All the *n* jobs are independent and available at the initial time.
- All the *m* stages are independent.
- There is infinite buffer capacity between stages in the production line.
- One job can be processed only by one machine at any time and one machine can process only one job at a time.
- The processing time of all jobs on all stages is known and deterministic.
- Jobs processing sequence are known.
- The set up time of all jobs is included in the processing time.
- The travel time between stages is negligible.

To present the mathematical model, the following notations are used:

j, l : Jobs index,

i, h : Stages index,

k : Machines in each stage index,

n : Number of jobs,

m : Number of stages,

s_i : Number of machines in stage i,

p_{ijk} : The processing time of job j on stage i on machine k,

M : Large number,

C_{ij} : The completion time of job j in stage i,

$$X_{ijk} = \begin{cases} 1 & \text{if job } j \text{ is assigned to machine } k \text{ at stage } i \\ 0 & \text{otherwise} \end{cases}$$

$$Y_{ilj} = \begin{cases} 1 & \text{if job } l \text{ is processed earlier than job } j \text{ at stage } i \\ 0 & \text{otherwise} \end{cases}$$

$$W_{ijl} = \begin{cases} 1 & \text{if job } l \text{ and job } l \text{ are processed on same machine at stage } i \\ 0 & \text{otherwise} \end{cases}$$

Therefore the mathematical model of FFS scheduling problem can be formulated as follow:

$$Min \; Z = \max\{C_{mj}\} \qquad \forall j : 1 \le j \le n \qquad (1)$$

Subject to:

$$\sum_{k=1}^{s_i} X_{ijk} = 1 \qquad \begin{matrix} \forall j : 1 \le j \le n \\ \forall i : 1 \le i \le m \end{matrix} \qquad (2)$$

$$C_{1j} \ge \sum_{k=1}^{s_1} p_{1jk}.X_{1jk} \qquad \forall j : 1 \le j \le n \qquad (3)$$

$$C_{ij} \ge C_{i-1,j} + \sum_{k=1}^{s_i} p_{ijk}.X_{ijk} \qquad \begin{matrix} \forall j : 1 \le j \le n \\ \forall i : 2 \le i \le m \end{matrix} \qquad (4)$$

$$\begin{matrix} C_{ij} + M \times (1 - Y_{ilj}) \\ \ge C_{il} + p_{ijk}.X_{ijk} \end{matrix} \qquad \begin{matrix} \forall j, l : 1 \le j, l \le n, \\ \forall k : 1 \le k \le s_i \\ \forall i : 1 \le i \le m. \end{matrix} \qquad (5)$$

$$C_{il} + M \times (1 - W_{ijl} + Y_{ilj}) \qquad \forall j, l : 1 \leq j, l \leq n,$$
$$\geq C_{ij} + p_{ilk}.X_{ilk} \qquad \forall k : 1 \leq k \leq s_i \qquad (6)$$
$$\forall i : 1 \leq i \leq m.$$

$$\forall j, l : 1 \leq j, l \leq n,$$
$$W_{ijl} \geq X_{ijk} + X_{ilk} - 1 \qquad \forall k : 1 \leq k \leq s_i \qquad (7)$$
$$\forall i : 1 \leq i \leq m.$$

$$\forall j, l : 1 \leq j, l \leq n,$$
$$X_{ijk} = \{0,1\} \qquad \forall k : 1 \leq k \leq s_i \qquad (8)$$
$$\forall i : 1 \leq i \leq m.$$

$$Y_{ilj} = \{0,1\} \qquad \forall j, l : 1 \leq j, l \leq n \qquad (9)$$

$$W_{ijl} \in \{0,1\} \qquad \begin{array}{l} \forall j, l : 1 \leq j, l \leq n, \\ \forall i : 1 \leq i \leq m. \end{array} \qquad (10)$$

The objective function (1) minimizes makespan and constraint (2) indicates that each job can be assigned to one machine at each stage. Constraint (3) ensures that completion time of job j in the first stage is greater than or equal to its processing time in this stage. Relation between completion times in two consecutive stages for job *j* can be seen in constraint (4). Constraint sets (5) and (6) preclude the interference between the processing operations of any two jobs on a machine. At most, one of these two constraint sets is active for each pair of jobs.

If job *l* is processed before job l on the same machine in stage i, constraint set (5) is activated to prevent interference between the processing operations of these two jobs and constraint (6) is satisfied for all values of j and *l* which have the stated condition. In the opposite situation, the roles of these two constraint sets are changed. Constraint (7) determines the jobs which are processed on the same machine in stage i. Finally both constraints (8), (9) and (10) force variables X_{ijk}, Y_{ilj} and W_{ijl} to assume binary values 0 or 1.

3. Lower Bound

As mentioned above, because of NP-completeness of FFS problem with unrelated parallel machines in each stage, we can't attain optimum solution for large instances. On the other hand, we need a datum to evaluate the proposed heuristic in large scale instances. A new stage-based lower bound for FFS scheduling problem with unrelated parallel machines is explained in this section.

It consists of three sections: the first section compute a lower bound for machine waiting time in each stage, the second section calculate a lower bound for each machine workload and the third section compute an estimated time for last job at sequence on each machine to pass from each stage to last stage.

Preposition 1: a stage-based lower bound for FFS scheduling problem with C_{max} objective function is equals:

$$LB_i^1 = \frac{1}{S_i} \left[\sum_{j=1}^{SPT(S_i)} \sum_{o=1}^{i-1} \min_k P_{ojk} + \sum_{o=1}^{i-1} \min_j \left\{ \min_k P_{ojk} \right\} (S_i - S_{i-1})^+ + \sum_{j=1}^{n} \sum_{k=1}^{S_i} \min_k P_{ijk} + \sum_{j=1}^{SPT(S_i)} \sum_{o=i+1}^{m} \min_k P_{ojk} \right] \qquad (11)$$

Proof: Minimum time to receiving job j to stage i equals $\sum_{o=1}^{i-1} \min_k P_{ojk}$. If there is S_i machines at stage i, total idle time at stage i equals $\sum_{j=1}^{SPT(S_i)} \sum_{o=1}^{i-1} \min_k P_{ojk}$.[2] Therefore minimum waiting time of stage i tasks equals $\sum_{j=1}^{SPT(S_i)} \sum_{o=1}^{i-1} \min_k P_{ojk}$ in addition $\sum_{o=1}^{i-1} \min_j \{\min_k P_{ojk}\} (S_i - S_j)^+$.[3] So there is a machine at stage i so that its idle time is greater than or equal to

$$\frac{1}{S_i} \left[\sum_{j=1}^{SPT(S_i)} \sum_{o=1}^{i-1} \min_k P_{ojk} + \sum_{o=1}^{i-1} \min_j \left\{ \min_k P_{ojk} \right\} (S_i - S_{i-1})^+ \right]$$

---

[2] $\sum_{j=1}^{SPT(S_i)} \{h_j\}$ is sum of h_j of S_j orders whose h_j are shortest.

[3] $(S_i - S_{i-1})^+ = \max\{0, S_i - S_{i-1}\}$

The third term of equation presents a lower bound for workload at stage i. The total workload for entire jobs at stage i equal to $\sum_{j=1}^{n} \sum_{k=1}^{S_i} \min_k P_{ijk}$. Therefore there is a machine at stage i so that its workload is greater than or equal to $\frac{1}{S_i} \sum_{j=1}^{n} \sum_{k=1}^{S_i} \min_k P_{ijk}$. The last term shows necessary time to finish processing the last order on sequence on the remaining stages. If there is S_i machines at stage i, the total processing time for S_i orders equal to $\sum_{j=1}^{SPT(S_i)} \sum_{o=i+1}^{m} \min_k P_{ojk}$. So there is a machine at stage i so that the minimum necessary time to finish processing the last order on this machine on

the remaining stages is greater than or equal to $\frac{1}{S_i} \sum_{j=1}^{SPT(S_i)} \sum_{o=i+1}^{m} \min_{k} P_{ojk}$. Notice that if $i = 1$ the first term equal to 0 and if $i = m$ the last term equal to 0.

$$LB_i^2 = \frac{1}{S_i} \left[\sum_{j=1}^{SPT(S_i)} \sum_{o=i+1}^{m} \min_{k} P_{ojk} + \sum_{o=i+1}^{m} \min_{j} \left\{ \min_{k} P_{ojk} \right\} (S_{i-1} - S_i)^+ + \sum_{j=1}^{n} \sum_{k=1}^{S_i} \min_{k} P_{ijk} + \sum_{j=1}^{SPT(S_i)} \sum_{o=1}^{i-1} \min_{k} P_{ojk} \right] \quad (12)$$

Proof: obviously, by looking at problem backward and last proof, we can simply conclude this lower bound.

Preposition 3: a stage-based lower bound for FFS scheduling problem with C_{max} objective function is equals:

$$LB = \max\{roundup\{LB_i^1, LB_i^2\}\}, i = 1, 2, ..., m \quad (13)$$

Based on the integer processing time in each stage, the lower bound must be integer; therefore the resulted lower bound must be roundup.

4. Computational Study

In this section, performance of lower bound is evaluated. For evaluation purpose, some test problems is produced. These instances are dedicated to small size problems. As mentioned above, because of NP-completeness of FFS scheduling problem, it is very expensive to receive the optimal solution for the medium and large problem. Then test problems in this section are limited to the small size problem. For this purpose, 20 test problems with following features are generated:

Preposition 2: a stage-based lower bound for FFS scheduling problem with C_{max} objective function is equals:

Tab. 1. Experimental factor for small size problem

Experimental factor	Feature
Number of jobs	U[3,5]
Number of stages	U[2,4]
Number of machines each stage	U[1,3]
Processing time	U[5,10]

The results of comparison between lower bound and optimal solution presents in table 2. The optimal gap between the lower bound and the optimal solution is calculated as follows:

$$Optimal\ Gap = \frac{optimal\ solution - lower\ bound}{optimal\ solution} \quad (14)$$

Tab. 2. Lower bound performance evaluation

Test problem	No. of jobs	No. of stages	No. of machines each stages	Lower bound	Optimal solution	Optimal Gap
1	4	3	2 2 3	26	29*	---
2	4	3	2 1 3	37	40	8.11%
3	5	4	2 2 1 2	53	53	0.00%
4	3	4	2 2 2 1	37	38	2.70%
5	4	3	3 2 2	26	29*	---
6	5	2	2 2	22	26*	---
7	3	3	2 2 3	24	25	4.17%
8	4	2	1 3	27	27	0.00%
9	5	3	1 2 2	55	55	0.00%
10	4	3	2 1 3	46	47	2.17%
11	4	3	2 2 2	25	25	0.00%
12	3	3	3 3 2	23	26	13.04%
13	4	3	2 2 1	42	42	0.00%
14	4	4	3 1 2 2	50	52	4.00%
15	5	2	3 1	53	53	0.00%
16	3	3	3 2 2	23	25	8.70%
17	4	2	2 2	21	22	4.76%
18	4	2	1 2	29	29	0.00%
19	3	3	2 3 3	25	26	4.00%
20	4	4	1 1 2 2	50	50	0.00%

*After 600 seconds, solver is interrupted.

According to table 2, lower bound can achieve optimal solution in 53% of test problems and solver[4] can't attain optimal solution after 10 minutes in three

problems. Differences between lower bound and optimal solution (optimal gap) equal 3%. Therefore we can conclude lower bound have good quality to achieve optimal solution.

---

[4] All the optimal solutions are obtained by Lingo 9.0 software.

5. Conclusion

In this research, a new mathematical model is presented for flexible flow shop scheduling problem with unrelated parallel machines in each stage. The selective objective function is makespan. Because of NP-completeness, it is expensive to achieve the optimal solution in large scale problems. Therefore, we proposed a new global lower bound as a datum for large scale problem. The results show that, the lower bound has 3% difference from optimal solution in small instances. Therefore it can be used to evaluate other heuristic.

Future works can consider other characteristics of FFS environments, such as availability constraints, sequence dependent set up time (cost), and identical machines. Metaheuristics algorithms (SA, TS, GA …) can also be applied for this problem, and we can compare them with the proposed lower bound. Afterwards, lower bound can be changed for other environments such as job shop and open shop.

References

[1] Ribas, I., Leisten, R., Framinan, J.M. *"Review and Classification of Hybrid Flow Shop Scheduling Problems from a Production System and a Solutions Procedure Perspective"* Computers and Operation Research, Vol. 37, 2009, pp.1439–1454.

[2] Quadt, D., Kuhn, H. *"Conceptual Framework for Lot-Sizing and Scheduling of Flexible Flow Lines"* International Journal of Production Research, Vol. 43(11), 2005, pp. 2291–2308.

[3] Adler, L., Fraiman, N., Kobacker, E., Pinedo, M., Plotnicoff, J.C., Wu, T.P., *"BPSS: a Scheduling Support System for the Packaging Industry"* Operations Research, Vol. 41, 1993, pp. 641–648.

[4] Voss, S., Witt, A. *"Hybrid Flow Shop Scheduling as a Multi-Mod Multi-Project Scheduling Problem with Batching Requirements: a Real-World Application"* International Journal of Production Research, Vol. 105(2), 2007, pp. 445–458.

[5] Wittrock, R.J. *"An Adaptable Scheduling Algorithm for flexible flow lines"* Operations Research, Vol. 36, 1988, pp. 445–453.

[6] Leon, V.J., Ramamoorthy. B., *"An Adaptable Problem Space-Based Search Method for Flexible Flow Line Scheduling"* IIE Transactions, Vol. 29, 1997, pp. 115–125.

[7] Agnetis, A., Pacifici, A., Rossi, F., Lucertini, M., Nicoletti, S., Nicolo, F., *"Scheduling of Flexible Flow Shop in an Automobile Assembly Plant"* European Journal of Operation Research, Vol. 97(2), 1997, pp. 348–362.

[8] Jin, Z.H., Ohno, K., Ito, T., Elmaghraby, S.E., *"Scheduling Hybrid Flowshops in Printed Circuit Board Assembly Lines"* Production and Operation Management, Vol. 11(2), 2002, pp. 216–230.

[9] Hayrinen, T., Johnsson, M., Johtela, T., Smed, J., Nevalainen, O., *"Scheduling Algorithms for Computer-Aided Line Balancing in Printed Circuit Board Assembly"* Production Planning and Control, Vol. 11, 2000, pp. 497–510.

[10] Alisantoso, D., Khoo, L.P., Jiang, P.Y., *"An Immune Algorithm Approach to the Scheduling of a Flexible PCB Flow Shop"* International Journal of Advanced Manufacturing Technology, Vol. 22(11–12), 2003, pp. 819–827.

[11] Ruiz, R., Maroto., C., *"A Genetic Algorithm for Hybrid Flowshops with Sequence Dependent Setup Times and Machine Eligibility"* European Journal of Operation Research, Vol. 169, 2006, pp. 781–800.

[12] Lee, G.C., Kim, Y.D., Choi, S.W., *"Bottleneck-focused scheduling for a hybrid flowshop"* International Journal of Production Research. Vol. 42, 2004, pp. 165–181.

[13] Santos, D.L., Hunsucker, J.L., Deal, D.E., *"On Makespan Improvement in Flow Shops with Multiple Processors"* Production Planning Control, Vol. 12 (3), 2001, pp. 283–295.

[14] Santos, D.L., Hunsucker, J.L., Deal, D.E., *"On makespan improvement in flow shops with multiple processors"* Production Planning Control, Vol. 12 (3), 2001, pp. 283–295.

[15] Sawik, T., *"An Exact Approach for Batch Scheduling in Flexible Flow Lines with Limited Intermediate Buffers"* Mathematical Computation Model. Vol. 36(4–5), 2002, pp. 461–471.

[16] Guinet, A.G.P., Solomon, M.M., *"Scheduling Hybrid Flowshops to Minimize Maximum Tardiness or Maximum Completion Time"* International Journal of Production Research, Vol. 34(6), 1996, pp. 1643–1654.

[17] Kadipasaoglu, S.N., Xiang, W., Khumawala, B.M., *"A Comparison of Sequencing Rules in Static and Dynamic Hybrid Flow Systems"* International Journal of Production Research, Vol. 35(5), 1997, pp. 1359–1384.

[18] Botta-Genoulaz, V., *"Hybrid Flow Shop Scheduling with Precedence Constraints and Time Lags to Minimize Maximum Lateness"* International Journal of Production Economic, Vol. 64(1–3), 2000, pp. 101–111.

[19] Cheng, J., Yoshiyuki, K., Kise, H., *"A Shifting Bottleneck Approach for Parallel-Machine Flowshop Scheduling Problem"*, Journal of the

Operation Research Society of Japan, Vol. 44(2), 2001, pp. 141–154.

[20] Quadt, D., Kuhn, H., "*A Taxonomy of Flexible Flow Line Scheduling Procedures*" European Journal of Operation Research, Vol. 178(3), 2007, pp. 686–698.

[21] Torabi, S.A., Ghomi, S.M.T.F., Karimi, B., "*A Hybrid Genetic Algorithm for the Finite Horizon Economic Lot and Delivery Scheduling in Supply Chains*, European Journal of Operation Research, vol. 173(1), 2006, pp. 173–189.

[22] Kadipasaoglu, S.N., Xiang, W., Khumawala, B.M., "*A Note on Scheduling Hybrid Flow Systems*" International Journal of Production Research, Vol. 35(5), 1997, pp. 1491–1494.

[23] Suresh, V., "*A Note on Scheduling of Two-Stage Flow Shop with Multiple Processors*", International Journal of Production Economic, Vol. 49(1), 1997, pp. 77–82.

[24] Low, C., "*Simulated Annealing Heuristic for Flow Shop Scheduling Problems with Unrelated Parallel Machines*", Computer and Operation Research, Vol. 32, 2005, pp. 2013–2025.

[25] Sawik, T., "*Integer Programming Approach to Production Scheduling for Make- to-Order Manufacturing*", Mathematical and Computer Model, Vol. 41(1), 2005, pp. 99–118.

[26] Low, C.Y., Hsu, C.J., Su, C.T., "*A Two-Stage Hybrid Flowshop Scheduling Problem with a Function Constraint and Unrelated Alternative Machines*", Computer and Operation Research, Vol. 35(3), 2008, pp. 845–853.

[27] He, L.M., Sun, S.J., Luo, R.Z., "*A Hybrid Two-Stage Flowshop Scheduling Problem*", Asia Pacific Journal of Operation Research, Vol. 24(1), 2007, pp. 45–56.

[28] Snanos, D.L., Hunsucker, J.L., Deal, D.E., "*Global Lower Bounds for Flow Shops with Multiple Processors*", European journal of operational research, Vol. 80, 1995, pp. 112-120.

[29] Haouari, M., M'Hallah, R., "*Heuristic Algorithm for Two Stage Hybrid Flow Shop*", Operation Research Letter, Vol. 21, 1997, pp. 43-53.

[30] Soewandy, H., Elmaghrby, S.E., "*Sequencing Three-Stage Flexible Flowshops with Identical Machines to Minimize Makespan*", IIE transactions, Vol. 33, 2001, pp. 985-993.