LOT STREAM?NG STRATEGY W?TH NON-IDLING CASE ON FLEX?BLE JOB SHOP SCHEDULING PROBLEM

Yunus Demir, Kürsat Isleyen

9[th] International Conference of Modeling, Optimization and Simulation - MOSIM'12
June 06-08, 2012 – Bordeaux - France
"Performance, interoperability and safety for sustainable development"

LOT STREAMİNG STRATEGY WİTH NON-IDLING CASE ON FLEXİBLE JOB SHOP SCHEDULING PROBLEM

Y. DEMİR, S. K. ISLEYEN
Ataturk University
Erzurum-Turkey
demiry@atauni.edu.tr, kursat.isleyen@atauni.edu.tr

ABSTRACT: *In this paper we described scheduling system of press shops in automotive industry. In these shops metal sheets are being shaped as kinds of automotive parts with press machines. These machines can serve for different operations with changing molds on them. The structure basically corresponds to a flexible job-shop scheduling problem with this special characteristic of machines. In this press shop items between operations are transferred with containers those are designed for related item with defined capacity. This structure allows us to use lot streaming strategy to reduce makespan. On the other hand according to corporate policy idle operators that are waiting job is not desired. We established a mathematical model taking into account these circumstances and compared with other modelling approach proposed in the literature.*

KEYWORDS: *Flexible job shop scheduling, Lot streaming, Mathematical modeling*

1 INTRODUCTION

In this paper, we focused on solving the lot streaming problem in a flexible job shop environment. The flexible job-shop scheduling problem (FJSP) is an extension of the classical JSP, where operations are allowed to be processed on any among a set of available machines. FJSP is more difficult than the classical JSP because it contains an additional problem that is assigning operations to machines (Bagheri et al.,2010).

One of the most difficult problems in this area is the Job-shop Scheduling Problem (JSP), where a set of jobs must be processed on a set of machines, each job is formed by a sequence of consecutive operations and each operation requires exactly one machine. Machines are continuously available and can process one operation at a time without interruption. The decision concerns how to sequence the operations on the machines, such as a given performance indicator is optimized. A typical performance indicator for JSP is the makespan, i.e., the time needed to complete all the jobs. JSP is a well-known NP-hard problem (Pezzella et al.,2008).

FJSP is desired to process operations on a machine chosen among a set of available ones. Therefore, the FJSP is more computationally difficult than the JSP (Hmida et al.,2010). The problem of scheduling jobs in FJSP could be decomposed into two sub-problems:

The routing sub-problem that assigns each operation to a machine selected out of a set of capable machines.

The scheduling sub-problem that consists of sequencing the assigned operations on all machines in order to obtain a feasible schedule to minimize the predefined objective function.

Unlike the classical JSP where each operation is processed on a predefined machine, each operation in the FJSP can be processed on one out of several machines. This makes FJSP more difficult to solve due to the consideration of both routing of jobs and scheduling of operations. Moreover, it is a complex combinatorial optimization problem. Therefore FJSP is NP-hard too (Zhang et al.,2011)

With respect to lot streaming, a job is actually a lot composed of identical items. In classical flexible job shop scheduling problems a lot is usually indivisible. The entire lot must be completed before being transferred to its successor operation, which leads to low machine utilization and long completion times. Lot streaming techniques, on the other hand, provide the possibility of splitting a lot into multiple smaller sublots, which can be treated individually and immediately transferred to the next stage once they are completed. Different sublots of the same job can thus be simultaneously processed at different operation stages. As a result of operation overlapping, the production can be considerably accelerated. If the production lot is processed without splitting, the average work in process (WIP) will be equal to production lot size. However, in case of splitting the production lot into sublots, departure of the first sublot reduces the WIP level by its size and the remaining sublots continue to reduce the WIP level by their sublot sizes. Reduction in space requirements and material handling system capacity requirements can be thought as the other benefits of lot streaming (Edis et al.,2007). In the last years, a majority of researches focused on solving lot streaming problems in a flow shop production system (Baker, 1995), (Potts and Baker, 1989), (Vickson and Alfredsson, 1992), (Chen and Steiner, 1998), (Chen and Steiner, 2003), (Biskup and Feldmann, 2006), (Feldmann and

Biskup, 2005). The flexible job shop scheduling problem, on the contrary, has received little attention (Buscher and Shen, 2011). (Gomes et al., 2005) proposed time indexed mathematical model for FJSP overlapping in operation with parallel machines (i.e, capable machines with same efficiency for an operation). (Torabi et al., 2005) addressed the common cycle multi-product lot-scheduling problem in deterministic flexible job shops where the planning horizon is finite and fixed by management. First, they developed a new mixed zero-one nonlinear model to solve the problem to optimality. Then, they have suggested an efficient enumeration method to determine an optimal solution. (Alvarez-Valdes et al., 2005) developed a heuristic to schedule flexible job-shop in a glass factory. (Fattahi et al., 2009) formulated a mixed integer linear programming (MILP) and developed a simulated annealing algorithm to solve large scale problems. They tested their algorithm on a set of random test problems generated by them. (Khalife et al., 2010) developed a simulated annealing algorithm to solve multi-objective FJSP with overlapping in operations. They solved test problems generated by (Fattahi et al., 2009) for multi-objective function consist of weighted sum of the three objective values and for each three objectives independently. (Farughia et al., 2011) proposed a hybrid meta-heuristic algorithm called memetic algorithm based on considering overlapping in operation in the FJSP. They solved same test problems of (Fattahi et al., 2009) with their algorithm and compared the results.
This paper presents a mathematical model which is modified from model developed by (Özgüven et al., 2010) for FJSP enabling it to solve the FJSP with overlapping in operations. Proposed mathematical model is tested on a set of test problems and we compared our results with results obtained by (Fattahi et al., 2009).

2 PROBLEM DEFINITION AND ASSUMPTIONS

This paper describes the scheduling systems of press shops producing large automotive parts. Press machines can serve for lot of jobs with changing their molds. Because of this property of machines press shops can be investigated under flexible job shop scheduling topic.
FJSP is a generalization of the traditional JS), in which it is desired to process operations on a machine chosen among a set of available ones (Ben Hmida et al., 2010) FJSP can be formulated as follows (Ho et al., 2007):

1. There is a set of n jobs J = {J1,…,Jn} and a set of m machines M = {M1,..., Mm}.;
2. Let J = {Ji}1≤i≤n, indexed i, be a set of n jobs;
3. Let M= {Mk}1≤k≤m, indexed k, be a set of m machines;
4. Each job consists of a predetermined sequence of operations;
5. Each operation Oij can be processed without interruption on any of machine k. ($k \in M_{ij}$). Therefore p_{kij} is implied processing time of O_{ij} on machine k.

In press shops parts are transferred between operations with containers designed for related part. These containers are with different capacity according to part that contain. This property of production system is allowing the use of lot steaming strategy. As a result of lot streaming, several potential benefits can be obtained (Low et al., 2004):

(a) Reduction of production lead times (thus, better due-date performance),
(b) Reduction of WIP inventory and associated WIP costs,
(c) Reduction of interim storage and space requirements, and
(d) Reduction of material handling system (MHS) capacity requirements.

As an example, we can consider a batch with 500 items, if each item has a processing time of 2 time units on machine 1and 1 time unit on machine 2, then according to traditional batch manufacturing, the makespan will be 500*2+500*1=1500 as shown in figure 1.

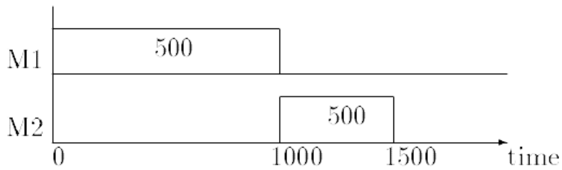

Figure 1: Traditional batch manufacturing

Next, if we consider splitting this batch into two sublots with sizes 300 and 200, then it takes 300*2=600 time units for the items of the first sublot to be ready for machine 2. In the same manner, the items of second sublot will be ready for machine after 300*2+200*2=1000 time units. As seen in figure 2 dividing this batch into two sublots provides a makespan of 1200 time units which is smaller than the makespan under the original case. This small example gives an idea about how lot streaming accelerates the process of the items at the same time reducing the flow time of parts in the process those are the aims of press shops.

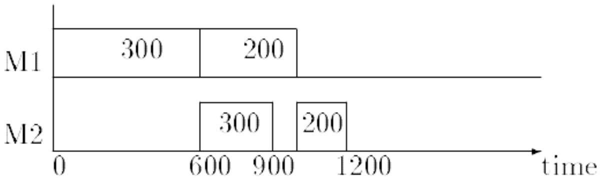

Figure 2: Lot streaming strategy

Additionally in press shops idle operator is not allowed. This is called in lot streaming literature "Non-Idling Case". If prior operation's processing time is slower than the successor operation there will be idling time for operator. For instance we have two consecutive operations. Prior operation's processing time 25 time units and successor operation' processing time 15 time units. In such a case there will idling time shown with red area in figure 3.

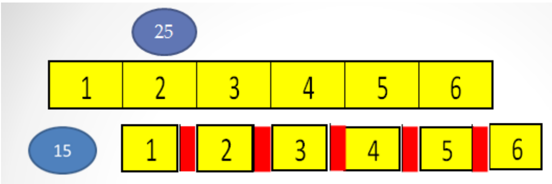

Figure 3: Idling Time

(Fattahi et al., 2009) limited extent of overlapping by a coefficient, ov_{ij} defined as the proportion of operation O_{ij} that has to be processed before its successor O_{ij+1} can start. In our model we divide entire demand of a product into sublots (predefined sized containers) and schedule them individually. So we do not need any coefficient. (Fattahi et al., 2009) solved test problems with 0.1 overlapping coefficient (ov_{ij}=0.1) for all operations. To ensure the equal terms to compare with Fattahi's model we assumed demand for each part is 10 and sublot size is 1. (it corresponds 0.1 overlapping coefficient)

In this paper we tried to consist a mathematical model ensures circumstances above with these assumptions (Xia and Wu, 2005):

1) All machines are available at time 0 and independent from each other.
2) All jobs are released at time 0 and independent from each other.
3) Setting up times of machines and transportation times between operations are negligible.
4) Maintenance activities and machine break downs are neglected.
5) At a given time, a machine can execute at most one operation.
6) There are no precedence constraints among the operations of different jobs.
7) More than one operation of the same job can be executed at a time.

3 PRESENTATION OF MODEL

The following notation is used for the formulation of our mathematical model:

<u>Indices and sets</u>

i jobs ($i,f \in J$)
j operations ($j,g \in O$)
k machines ($k \in M$)
J the set of jobs
M the set of machines
O the set of operations
O_i ordered set of operations of job i
$O_{if(i)}$ first operation of job i
$O_{il(i)}$ last operation of job i
$l_{f(ij)}$ first sublot of operation O_{ij}
$l_{l(ij)}$ last sublot of operation O_{ij}
M_{ij} the set of alternative machines on which operation O_{ij} can be processed,

<u>Parameters</u>

t_{kij} the processing time of operation Oij on machine k
M a large number
d_i demand for job i
Pt_i lot size of job i
pr_i total lot number of job i
$pr_i = [d_i/pt_i]^+$
tp_{ijlk} the processing time of sublot of operation Oij on machine k
If l. sublot is the last sublot ($l=pr_i$);

$$(d_i\text{-}pt_i*(pr_i\text{-}1))*\ t_{kij}$$

Otherwise;

$$pt_i*\ t_{kijk}$$

E_k: The set of operations which can be performed on machine k

This calculation of sublot processing time prevents the undue idling time between operations as seen in figure 4. As an example, we have 112 units demand for product p and sublot size of this product is 25 units. In such a situation the last sublot's size will be 12 units not same as the prior sublots (25 units).

<u>Decision variables</u>

y_{ijk} 1, if machine k is selected for operation O_{ij};
0, otherwise
x_{ijlk} 1, if l. sublot of operation O_{ij} is assigned on machine k
0, otherwise

s_{ijlk} the starting time of lot l of operation O_{ij} on machine k
c_{ijlk} the completion time of lot l of operation O_{ij} on machine k
z_{ijfgk} 1, if operation O_{ij} precedes operation Ofg on machine k;
0, otherwise

C_{ijlk} the completion time of l. sublot of operation O_{ij} on machine k
$CO_{iO_{il(i)}l_{l(ij)}}$ completion time of last sublot of final operation of job i
C_{max} maximum completion time over all jobs (makespan)

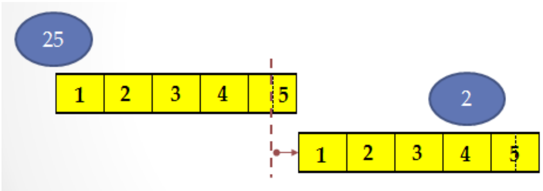

Figure 4: ⟶ Undue idling time

The proposed mathematical model is defined as follows:

Minimize C_{max}

$$\sum_{k\epsilon Mij} C_{iO_{il(i)}l_{l(ij)}k} = CO_{iO_{il(i)}l_{l(ij)}} \quad \forall i \qquad (1)$$

$$Cmax > CO_{iO_{il(i)}l_{l(ij)}} \quad \forall i \qquad (2)$$

$$\sum_{k\epsilon Mij} y_{ijk} = 1 \quad \forall i,j \qquad (3)$$

$$\sum_{l} x_{ijlk} = pr_i * y_{ijk} \quad \forall i,j,\forall k\epsilon Mij \qquad (4)$$

$$s_{ijlk} + C_{ijlk} \leq x_{ijlk} * M$$
$$\forall i,j,\forall k\epsilon Mij, \forall l = 1,\ldots,pr_i \qquad (5)$$

$$C_{ijlk} \geq s_{ijlk} + tp_{ijlk} - (1 - x_{ijlk}) * M$$
$$\forall i,j,\forall k\epsilon Mij, \forall l = 1,\ldots,pr_i \qquad (6)$$

$$s_{ijl_{f(ij)}k} \geq C_{fgl_{l(ij)}k} - z_{ijfgk} * M$$
$$\forall i,j,f,g \in E_k, O_{ij} \neq O_{hg} \qquad (7)$$

$$s_{fgl_{f(ij)}k} \geq C_{ijl_{l(ij)}k} - (1 - z_{ijfgk}) * M$$
$$\forall i,j,f,g \in E_k, O_{ij} \neq O_{hg} \qquad (8)$$

$$\sum_{k\epsilon Mij} s_{ijlk} \geq \sum_{k\epsilon Mij-1} C_{ij-1lk}$$
$$\forall i,\forall j\epsilon O_i - O_{if(i)}, \forall l = 1,\ldots,pr_i \qquad (9)$$

$$\sum_{k\epsilon Mij} s_{ijlk} \geq \sum_{k\epsilon Mij} C_{ijl-1k}$$
$$\forall i,j,\forall l = 2,\ldots,pr_i \qquad (10)$$

$$C_{ijlk} - s_{ijl_{f(ij)}k} \leq d_i * t_{kij}$$
$$\forall i,j,\forall k\epsilon Mij, \forall l = 1,\ldots,pr_i \qquad (11)$$

Constrain (1) computes the completion times of last sublot of final operation of a job. Constraint (2) determines the makespan. Constraint (3) ensures that operation O_{ij} is assigned to only one machine. Constraint (4) ensures the assigned sublots of O_{ij} to machine k equal to total sublot number of O_{ij}. Constraint (5) set the starting and completion times of l. sublot of operation O_{ij} on machine k equal to zero if it is not assigned to machine k. Otherwise, the constraints (6) guarantee that the difference between the starting and the completion times is equal at least to the processing time of l. sublot of operation O_{ij} on machine k. Constraints (7) and (8) force operation O_{ij} and operation O_{fg} cannot be done at the same time on any machine k in the set E_k and ensures that he case no interruption in the sequence of sublots of an operation O_{ij}. Constraint (9) ensures that the precedence relationships between the operations of a job are not violated, i.e. the l. sublot of operation O_{ij} is not started before the l. sublot of operation O_{ij-1} has been completed. Constraint (10) takes care of precedence relationship sublots of operations i.e. the l. sublot of operation O_{ij} is not started before the (l-1). sublot of operation O_{ij} has been completed. Constraint (11) ensures that the case where there is no production interruption time (i.e., the idle time) between any two adjacent sublots at the same stage. In other words, sublots on a particular stage have to be processed directly one after the other.

4 COMPARISON OF MODELS

In this section results obtained by proposed model by us and (Fattahi et al.,2009) (it will be referred to as Model F) are compared. We used some of the test problems randomly created by (Fattahi et al., 2009). These problems divided into three groups: Small, medium and large size FJSPs. Problems are coded in the mathematical language GAMS and used CPLEX solver. Test problems are run on PC with Core(TM) 2 Quard CPU, 2.66 GHz processor and 4 GB RAM. The runs are terminated after 86400 seconds (24 hours). As shown in Table 1. for small sized problems Cmax values obtained by each model is same and optimum. But proposed model obtained these results at lower CPU time. Proposed model also finds the optimal solution of first three medium sized problems which Model F does not. For the remaining problems proposed model found best integer solutions however Model F cannot found any best integer solution.

Table 1
The computational results of Model F and proposed model

	Model F					Proposed Model				
Size (i.j.k)	Integer variable	Non integer variable	Number of Const.	CPU time (s)	Cmax	Integer variable	Non integer variable	Number of Const.	CPU time (s)	Cmax
Small size										
2.2.2.	40	26	136	1	66	100	201	320	0.28	66
2.2.2.	32	24	110	1	107	92	201	284	0.16	107
3.2.2.	72	36	237	16	221	150	301	460	0.59	221
3.2.2	84	38	239	15	355	151	301	462	0,78	355
3.2.2	84	38	275	250	119	162	301	504	1,5	119
3.3.3	189	50	503	452	256	317	631	781	0,7	256
3.3.5	225	55	604	63	233.5	492	991	984	1.39	233.5
3.3.4	216	55	595	1303	193	411	811	912	1.67	193
3.3.3	243	56	590	1850	171.7	336	631	852	2.68	171.7
4.3.5	300	66	870	1179	419.5	650	1321	1260	1	419.5
Medium Size										
4.3.5	360	71	983	>3600	(357.5;419.5)	589	1681	1678	0.94	419.5
5.3.7	840	106	1996	>3600	(196.8;380.7)	1185	2251	2160	1232	325.1
6.3.7	1260	131	2831	>3600	(196.8; -)	1462	2701	2684	64806	371.6
7.3.7	1617	149	3803	>3600	(273.4; -)	1730	3151	3180	86400	(300.2; 454.9)
7.3.7	1617	149	3740	>3600	(2370.1; -)	1725	3151	3160	86400	(340.8; 443.3)
Large Size										
8.3.7	2184	174	4766	>3600	(- ; -)	1999	3601	3658	86400	(347.3; 565.4)
8.4.7	3584	219	7883	>3600	(- ; -)	2746	4801	4992	86400	(325.2; 803.0)
9.4.8	4896	256	9778	>3600	(- ; -)	3426	6121	5912	86400	(308.9; 832.5)
11.4.8	7040	308	14190	>3600	(- ; -)	4303	7481	7436	86400	(308.0; 1157.0)
12.4.8	8832	346	16784	>3600	(- ; -)	4775	8161	8270	86400	(459.4; 1478.7)

5 CONCLUSIONS

In this paper a mathematical model is developed for FJSP with overlapping in operations. Then proposed model is compared with alternative model proposed (Fattahi et al., 2009) and superiority of our model is displayed in terms of Cmax and CPU time. Results also showed that it is quite difficult to reach an optimal solution to this problem in real life with mathematical modeling technique because of its NP-hard structure. Therefore authors of this paper are planning as a future study to develop a solution approach with genetic algorithm to solve large problem instances.

REFERENCES

Alvarez-Valdes R., Fuertes A., Tamarit J.M., Gim_enez G., Ramos R., 2005. A heuristic to schedule flexible job-shop in a glass factory. European Journal of Operational Research, 165, p. 525–534

Bagheri, A. et al., 2010. An artificial immune algorithm for the flexible job-shop scheduling problem. Future Generation Computer Systems, 26(4): p. 533-541.

Baker K.R., 1995. Lot Streaming in the Two-Machine Flow Shop with Setup Times. Annals of Operations Research, 57, p.1-11.

Ben Hmida A. et al., 2010. Discrepancy search for the flexible job shop scheduling problem. Computers & Operations Research, 37(12), p. 2192-2201.

Biskup D. and Feldmann M., 2006. Lot Streaming with Variable Sublots: An Integer Programming Formulation. Journal of Operational Research Society, 57, p. 296-303.

Buscher U. and Shen L., 2011. An Integer Programming Formulation for the Lot Streaming Problem in a Job Shop Environment with Setups. Proceedings of the international MultiConference of Engineers and computer Scientist Vol. 2, IMECS 2011, 2011 Hong Kong

Chen J. and Steiner G., 1998. Lot Streaming with Attached Setups in Three-Machine Flow Shops. IIE Transactions 30, p. 1075-1084.

Chen J.and Steiner G., 2003. On Discrete Lot Streaming in No-Wait Flow Shops. IIE Transactions, 35, p. 91-101.

Edis R., Ornek A., Eliiyi D., 2007. A Review On Lot Streaming Problems With Transportation Activities, İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi, 6(11), p. 129-142

Farugi H., Babak Y.Y, Hiresh S., Fyagh Z., Foruzan N., 2011. Considering The Flexibility and Overlapping in Operation in Job Shop Scheduling Based on Meta-heuristic Algorithms. Australian Journal of Basic and Applied Sciences, 5(11), p. 526-533

Fattahi P., Jolai F., Arkat J., 2009. Flexible job shop scheduling with overlapping in operations. Applied Mathematical Modelling, 33, p. 3076–3087

Feldmann M. and Biskup D., 2005. On Lot Streaming with Multiple Products. Discussion paper, No. 542, Department of Business Administration and Economics, Bielefeld University, Germany.

Gomes M. C., Barbosa-Póvoa A. P., Novais A. Q., 2005. Optimal scheduling for flexible job shop operation. International Journal of Production Research, 43(11), p. 2323–2353

Ho N. B, Tay J. C, Lai E., 2007. An effective architecture for learning and evolving flexible job-shop schedules. European Journal of Operations Research. 179(2), p. 333-316.

Khalife M. A., Abbasi B., Abadi A. H. K. D., 2010. A Simulated Annealing Algorithm for Multi Objective Flexible Job Shop Scheduling with Overlapping in Operations. Journal of Industrial Engineering, 5, p. 17-28

Low C., Hsu C., Huang K., 2004. Benefits of lot splitting in job-shop scheduling.The International Journal of Advanced Manufacturing Technology, 24(9), p. 773-780

Özgüven C., Özbakır L., Yavuz Y., 2010. Mathematical models for job-shop scheduling problems with routing and process plan flexibility. Applied Mathematical Modelling, 34, p. 1539–1548.

Pezzella, F., G. Morganti, and G. Ciaschetti, 2008. A genetic algorithm for the Flexible Job-shop Scheduling Problem. Computers & Operations Research, 35(10), p. 3202-3212.

Potts C.N. and Baker K.R., 1989. Flow Shop Scheduling with Lot Streaming. Operations Research Letters,8, pp. 297-303.

Torabi S.A., Karimi B., Fatemi Ghomi S.M.T., 2005. The common cycle economic lot scheduling in flexible job shops: The finite horizon case. Int. J. Production Economics, 97, p. 52–65.

Xia W, Wu Z., 2005. An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems. Computer and Industrial Engeneering, 48(2), p. 425-409.

Vickson R.G. and Alfredsson B. E., 1992. Two- and Three-Machine Flow Shop Scheduling Problems with Equal Sized Transfer Batches. International Journal of Production Research, 30, p. 1551-1574.

Zhang, G., L. Gao, and Y. Shi, 2011. An effective genetic algorithm for the flexible job-shop scheduling problem. Expert Systems with Applications, 38(4), p. 3563-3573.