6th International Conference on Industrial Engineering and Industrial Management.
XVI Congreso de Ingeniería de Organización. Vigo, July 18-20, 2012

A makespan minimization in an m-stage flow shop lot streaming with sequence dependent setup times: MILP model and experimental approach.

Gomez-Gasquet P[1], Segura-Andres R, Franco D, Andrés C

Abstract. This paper considers the recently growing interest the scientific community has put over the use of Lot Streaming technics (LS) or lot splitting. Implications are highlighted, from the scheduling point of view. It is presented this consideration and a MILP mathematical model to minimize makespan in an n-job flow shop problem with sequence dependent setup times (SDST). At the same time, the model is resolved, for the two- and three-machine flow shop, with the purpose of analyzing, through a design of experiments, which is the achieved makespan reduction and computation time increment regarding to no LS consideration used. Finally, it will be reflected about the use of non-optimum technics relating makespan values and expected calculation time.

Keywords: flow shop, lot streaming, MILP

1.1 Introduction

Lot Streaming (LS) is a concept that allows splitting jobs in smaller entities, sublots, what makes easier material transfer between stages. In the no hybrid flow shop, LS makes possible makespan or Cmax reduction such as it is showed in Figure 1.1. Nevetheless industrial companies should value if this reduction is enough in order to be worhty the increment of the complexity in the management that supposes the increase of materials flow in the shop. Besides of a LS use analysis in terms of Cmax reduction, it must not be subestimated the difficulty increase that involves job splitting, in a complex problem itself. In the figure 1.1 it is shown an example in what the use or not of LS is compared, in a two-stage flow shop with 3 jobs.

[1] Pedro Gómez Gasquet (✉)
Centro de Investigación de Gestión e Ingeniería de la Producción (CIGIP). Universitat Politècnica de València Cno. De Vera s/n, 46022 Valencia, Spain
e-mail: pgomez@cigip.upv.es

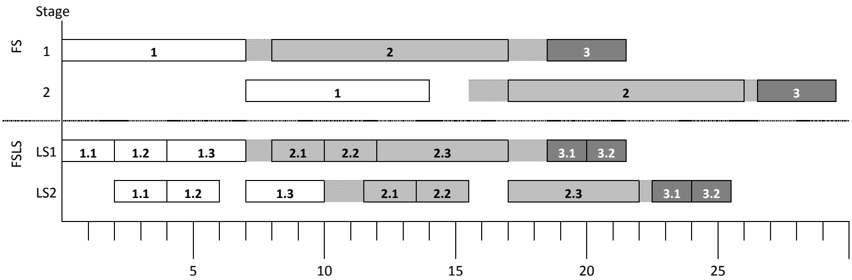

Fig. 1.1 Flow Shop vs. Flow Shop Lot Streaming

In this paper a literature review is carried out in section 1.2. Problem definition for a general case with m stages and n jobs through a MILP model is introduced in section 1.3. A mathematical model for the two- and three-stages problem is instanciated in section 1.4 with the objective of show in a simple case the potential benefits of LS technics, as well as the limitation in the use of optimum technics. Finally, in section 1.5 conclusions are sumarized.

1.2 Literature review

An analysis in a Flow Shop environment (FS) will be presented in this paper. We will compare the use of lot streaming versus not using it. Many years have passed since the first paper about flow shop (FS) was published (Johnson, 1954). From that paper up to now have passed more than five decades in which researchers have published hundreds of papers, some of them presented in a FS review (Gupta & Stafford Jr., 2006). The concept of lot streaming was introduced a decade after Johnson's paper (Reiter, 1966) with the aim of improving the objective function in a certain type of problems through splitting jobs into smaller sublots. Whereas the most difficult task in an *n* jobs FS problem is defining the sequence of jobs, when we combine FS and lot streaming (FSLS) we increase the difficulty of the problem; we need to decide the number of sublots besides the sequence, and even it might be necessary decide the size of each sublot also. One problem of this type, with only two machines, it could be classified as a NP-hard due to its complexity and the possibility of multiple combinations (Glass & Possani, 2011). A simple two-machine example showed that, an optimal solution generally cannot be found when the sequencing approach and the splitting approach are used independently (Potts & Baker, 1989). They suggested that the two approaches should be used simultaneously. There are problems that have not yet been addressed in any paper up to date such as it is showed in a FSLS review (Sarin & Jaiprakash, 2007).

In-between different operations in a FS configuration, it may appear setup times to reconfigure each machine to the next sublot; this sublot may be part of the same job or even though from a different one. To reduce the gap, setup concept has been widely tackled in FS environments (Allahverdi, Gupta, & Aldowaisan, 1999). In this paper, setup times will be sequence dependent (SDST); setups depend not only on the job to be processed next, but also on its immediately preceding job on the same machine. Different authors have published *m* machines FSLS papers with SDST for minimizing makespan as objective function using genetic algorithm (GA) (Lee, Sikora, & Shaw, 1997) or discrete harmony search (Pan, Duan, Liang, Kaizhou Gao, & Junqing Li, 2010). However, none of them achieved an optimal solution.

In a wide range of industrial situations, companies usually work in FS configurations, where in one or more stages it may be available more than one resource to perform their operations. This configuration it is known as Hybrid Flow Shop (HFS) and recently was published a review of the published papers (Ruiz & Vázquez-Rodríguez, 2010).

In this paper, we will analyze two different problems: FS vs. FS lot streaming (FSLS). We will consider also sequence dependent setup times (SDST).

1.3 *A* model definition

Lot streaming problems under study will be defined following the notation described in Sarin and Jaiprakash (2007) as reference. Our model will be $M/N/C/II/DV/\{SDST,C_{max}\}$. This model consists in a flow shop (FS) of *m* stages (M), one machine per stage, where *n* jobs (N) must be processed. Each job consists of U_i identical units. A job can be split into sublots that will be treated as separate entities in production. Each sublot requires processing on any of the machines in all stages. Sublots are consistent (C); the size of each sublot is kept constant on all stages. Intermittent idling (II) is allowed and sublots sizes have discrete values (DV). Setup times will be sequence dependent (SDST) and the objective is to minimize makespan (C_{max}).

The following assumptions are made: (1) all jobs are available at time zero; (2) the processing time of each item is known and deterministic; (3) no preemption is allowed; (4) there is a given smallest allowable size for the sublots of each job; (5) machines are available at any time; (6) each machine can process at most one sublot at a time; (7) each sublot can be processed on one machine at a time; (8) interleaving of sublots of different lots is not allowed.

The problem consists to decide number and size of sublots, for each job, as well as schedule them to minimize makespan. With the aim of constructing a general mathematical model, the information will be presented using the following indexes:

i index set of jobs {0..n}; t, t' index set of jobs {0..n+1}

h index set of jobs {1..n+2}; b, b' index set of jobs {0..n+2}
l, v index set of sublots of *i* job $\{0..L_i\}$
r index set of stages on the shop {1..R}
j, f index set of machines in *r* stage $\{1..m_r\}$

Parameters in the model are the data known beforehand:

Z_i number of units in job *i*
z_i minimum size of each sublot in job *i*
$P_{i,j,r}$ processing time for units of job *i*, on machine *j* at stage *r*
$S^X_{t,t',j,r}$ setup time for a sublot of job *t'*,preceded by a sublot of job *t*,on machine *j* at the stage *r*
$S^m_{i,j,r}$ setup time for a sublot of job *i*,preceded by a sublot of job *i*,on machine *j* at the stage *r*
M a very large positive number (larger than makespan)

The genetic algorithm determines the following variables:

$X_{l,i,j,r}$ (integer) number of units in sublot *l* of job *i* assigned to machine *j* at stage *r*
$C_{l,t,j,r}$ (integer) completion time of sublot *l* of job *t* on machine *j* at stage *r*
K_i (integer) completion time of job *i* (it corresponds to the maximum $C_{l,i}$)

Note that for a given job *i* and sublot *l*, $X_{l,i,j,r} > 0$ for at most one machine *j* (j=1…m_r) in stage *r*, indicating that sublot *l* of job *i* is allocated to machine *j* at the *r* stage.

$$y_{l,i,j,r} = \begin{cases} 1, \text{if sublot } l \text{ of job } i \text{ is performed on machine } j \text{ at stage } r \\ 0, \text{otherwise} \end{cases}$$

$$q_{l,b,v,b',j,r} = \begin{cases} 1, \text{if sublot } l \text{ of job } b \text{ is performed before sublot } v \text{ of job } b' \text{on machine } j \text{ at stage } r \\ 0, \text{otherwise} \end{cases}$$

With these notations, the problem can be formulated as the following MILP model. The objective is to minimize makespan (1):

$$(1) \qquad F.O.\ min\left\{max_i^n\{C_i\}\right\} = C_{max}$$

The constraints of the model are presented below in three sets, each representing one type of system constraint. The model is subject to:

Precedence constraints: This set of constrains ensures the processing order of sublots.

(2) $C_{l,i+1,j,r} \geq C_{v,t,j,r} + P_{i,j,r} * X_{l,i,j,r} + \left(S^{X}_{t,i+1,j,r} + S^{m}_{i+1,j,r}\right) + M * q_{v,t,l,i+1,j,r} - M \quad \forall l, \forall v, \forall t, \forall i, \forall j, \forall r$

(3) $C_{l,i+1,j,r} - P_{i,j,r} * X_{l,i,j,r} \geq 0 \quad \forall l, \forall i, \forall j, \forall r$

(4) $C_{l,i+1,j,r} - P_{i,j,r} * X_{l,i,j,r} \geq C_{l,i+1,f,r-1} + Y_{l,i,j,r} * M - M \quad \forall l, \forall i, \forall j, \forall f, \forall r > 1$

(5) $\sum_{t=0}^{n+1} \sum_{v=0}^{L_t} q_{v,t,l,i+1,j,r} - Y_{l,i,j,r} = 0 \quad \forall l, \forall i, \forall j, \forall r$

(6) $\sum_{h=1}^{n+2} \sum_{v=0}^{L_h} q_{l,i+1,v,h,j,r} - Y_{l,i,j,r} = 0 \quad \forall l, \forall i, \forall j, \forall r$

(7) $\sum_{h=1}^{n+2} \sum_{v=0}^{L_h} q_{0,0,v,h,j,r} = 1 \quad \forall j, \forall r$

(8) $\sum_{t=0}^{n+1} \sum_{v=0}^{L_t} q_{v,t,0,n+1,j,r} = 1 \quad \forall j, \forall r \quad \forall l, \forall i, \forall j, \forall r$

(9) $C_{0,0,j,r} = 0 \quad \forall j, \forall r$

(10) $q_{l,i+1,l+1,i+1,j,r} = \sum_{j=0}^{m_r} Y_{l+1,i,j,r} \quad \forall l < L_i, \forall i, \forall j, \forall r$

Constraint (2) ensures that a sublot cannot start on machine j at stage r before the previous sublot in the same machine j at the same stage r has been completely processed for any job at any stage. Constraint (3) ensures that the first sublot on machine j at stage r can be completed only after it has been on the machine for the necessary processing time. Constraint (4) ensures that a sublot of a job cannot start in the next stage before it has been completed in the actual stage. Constraints (5) and (6) ensures that every sublot must have a previous one (5) and another one which comes later (6). Constraints (7), (8) and (9) define all the fictitious jobs so each job will have an initial and a last job. Constraint (10) is it used to avoid interleaving on the problem. It makes that every job must be performed continuosly; sublot *l+1* will be processed just after it is finished sublot *l* in the same machine.

Constraints related to sublot sizes:

(11) $\sum_{j=0}^{m_r} Y_{l,i,j,r} \leq 1 \quad \forall l, \forall i, \forall r$

(12) $X_{l,i,j,r} \geq z_i * Y_{l,i,j,r} \quad \forall l, \forall i, \forall j, \forall r$

(13) $X_{l,i,j,r} \leq Z_i * Y_{l,i,j,r} \quad \forall l, \forall i, \forall j, \forall r$

(14) $\sum_{l=0}^{L_i} \sum_{j=0}^{m_r} X_{l,i,j,r} = Z_i \quad \forall i, \forall r$

(15) $\sum_{j=0}^{m_r} Y_{l,i,j,r} \geq \sum_{j=0}^{m_r} Y_{l+1,i,j,r} \quad \forall l < L_i, \forall i, \forall r$

(16) $\sum_{j=0}^{m_r} X_{l,i,j,r} = \sum_{j=0}^{m_{r+1}} X_{l,i,j,r+1} \quad \forall l, \forall i, \forall r < R$

(17) $\sum_{l=0}^{L_i} \sum_{j=0}^{m_r} Y_{l,i,j,r} = \sum_{l=0}^{L_i} \sum_{j=0}^{m_{r+1}} Y_{l,i,j,r+1} \quad \forall i, \forall r < R$

(18) $q_{l,i+1,l,i+1,j,r} = 0 \quad \forall l, \forall i, \forall j, \forall r$

Constraints (11), (12) and (13)define Y variables. Constraint (14) ensures that all the units are processed for all job at all stage. Constraint (15) ensures that sublots with a null number of units are required to follow the ones with units for each job at each stage. Constraint (16) ensures that the size of each sublot will remain equal in all job at all the stages. Constraint (17) ensures that the number of sublots will remain equal in all job at all stage. Constraint (18) avoids redundancy to process sublots that they have been already processed.

1.4 Experimental analysis and results

With the experimental analysis is it possible to notice makespan minimization using lot streaming technics (LS) in the case of m/6/C/II/DV/{SDST,C_{max}} with m=2 and m=3. Besides it is possible to assess the existing percentage of makespan reduction. Due to the theoretical difficulty the problem arise, it is proposed to analyze in what degree the increase in the computational effort is profitable in an industrial environment, and if substituting the optimizadory technic by a sub-optimum one it would be recommendable in terms of efficiency because, obviously, it will not be in terms of inverted time.

The experiment is based in the resolution of the mathematical model discussed in the section 1.3, using Gurobi Optimizer 4.6 in different scenarios. In all cases, unitary process times per stage and size of jobs are generated with a uniform distribution between [1…6] and [1…30] respectively. Minimum sublot size considered is 1 unit. For its analysis, design of experiments (DoE) has been performed with two factors: setup times (SDST) and number of stages (Stg). In the case of SDST, three different values have been considered, 10%, 30% and 50% time of the process time, all of them generated with a uniform distribution. Stg factor analyses of two- and three-stage shops. In every scenario a total of 5 replications have been performed. Dependent variable of the experiment is Cmax reduction percentage (rCmax), expressed in (Ec. 19) $rCmax = (CmaxLS - Cmax)/Cmax$, been Cmax the obtained value without LS and CmaxLS same example with LS. Due to the required computational effort for the LS case, analyzed values in the problem correspond to the 7.200 seconds, hence sub-optimum. In the case without LS optimal values have been always achieved.

Table 1.1 Analysis of Variance for rCmax - Type III Sums of Squares

Source	Sum of Squares	Df	Mean Square	F-Ratio	P-Value
MAIN EFFECTS					
A:SDSS	6,60527	2	3,30263	0,13	0,8758
B:Stg	1031,03	1	1031,03	41,62	0,0000
INTERACTIONS					
AB	12,5672	2	6,28362	0,25	0,7780
RESIDUAL	594,484	24	24,7702		
TOTAL (CORRECTED)	1644,69	29			

All F-ratios are based on the residual mean square error.

The results shown that in all cases the Cmax was bigger than CmaxLS, in other words LS approach is suitable for the problem. But in the table 1.1 ANOVA analysis shows as results using LS are not dependent on setup time values. However the increment of stage from 2 to 3 influences in the makespan, the mean of rCmax passes form -7.8% to -19.5%. In terms of an industrial application an improvement of 19.5%, even if setup times were high, could be consider excellent.

In the same experiment results of makespan for LS at 200 seconds were recorded, but the ANOVA of this new rCmax(200) shown no significant factors. It indicates the problem presented is very difficult to solve specially when LS is applied.

1.5 Conclusions

After the comparison between a flow shop with and without lot streaming, we can conclude that problem presented is, as it is highlighted by apart 1.2, an interesting challenge from research point of view, where further research needs to be carried out in order to exploit this technic/concept properly. This technic might improve companies' efficiency when it is possible to apply. And we should not forget that these problems, once we introduce SDST combined with lot streaming, are NP-hard.

At the same time, the experimental analysis that deal with a representative industrial situation, although not too complex, it makes clear that in all cases, makespan is minimized with the use of lot streaming technics although it is not better as longer setup times are as initially thought. However, its industrial application needs to be validated due to it persist the doubt if the makespan minimization compensates the complexity in the materials flow management that entails its startup over all for small shops.

Finally, it is worth pointing out the makespan difference achieved in 7200 seconds is lightly better than the one in 200 seconds obtained for 2-stage problem but much better form 3-stage problem. This means, from our point of view, optimal approaches need so much time in order to reach a solution where the problem is complex but well-oriented suboptimal approach could be a well balanced method in terms of makespan and computational effors.

In the future, our research will introduce and consider more realistic approaches such as Hybrid FS environments (HFS). When we combine HFS and lot streaming it increases the difficulty to find some relevant papers on the topic.

This work has been carried out as part of the project "PAID-06-10-2396 (NegoSol-MAS)" funded by Vicerrectorado de Investigación of Universitat Politècnica de València.

1.6 References

Allahverdi, A., Gupta, J. N. ., & Aldowaisan, T. (1999). A review of scheduling research involving setup considerations. *Omega*, *27*(2), 219–239. doi:10.1016/S0305-0483(98)00042-5

Feldmann, M., & Biskup, D. (2007). Lot streaming in a multiple product permutation flow shop with intermingling. *International Journal of Production Research*, *46*(1), 197–216. doi:10.1080/00207540600930065

Glass, C. A., & Possani, E. (2011). Lot streaming multiple jobs in a flow shop. *International Journal of Production Research*, *49*(9), 2669–2681. doi:10.1080/00207543.2010.532935

Gupta, J. N. D., & Stafford Jr., E. F. (2006). Flowshop scheduling research after five decades. *European Journal of Operational Research*, *169*(3), 699–711. doi:10.1016/j.ejor.2005.02.001

Johnson, S. M. (1954). Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, *1*(1), 61–68. doi:10.1002/nav.3800010110

Lee, I., Sikora, R., & Shaw, M. J. (1997). A genetic algorithm-based approach to flexible flow-line scheduling with variable lot sizes. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, *27*(1), 36–54. doi:10.1109/3477.552184

Liu, J. (2008). Single-job lot streaming in m − 1 two-stage hybrid flowshops. *European Journal of Operational Research*, *187*(3), 1171–1183. doi:10.1016/j.ejor.2006.06.066

Martin, C. H. (2009). A hybrid genetic algorithm/mathematical programming approach to the multi-family flowshop scheduling problem with lot streaming. *Omega*, *37*(1), 126–137. doi:10.1016/j.omega.2006.11.002

Pan, Q.-K., Duan, J., Liang, J. J., Kaizhou Gao, & Junqing Li. (2010). A novel discrete harmony search algorithm for scheduling lot-streaming flow shops. *Control and Decision Conference (CCDC), 2010 Chinese* (pp. 1531–1536). Presented at the Control and Decision Conference (CCDC), 2010 Chinese, IEEE. doi:10.1109/CCDC.2010.5498265

Potts, C. N., & Baker, K. R. (1989). Flow shop scheduling with lot streaming. *Operations Research Letters*, *8*(6), 297–303. doi:10.1016/0167-6377(89)90013-8

Reiter, S. (1966). A System for Managing Job-Shop Production. *The Journal of Business*, *39*(3), 371–393.

Ruiz, R., & Vázquez-Rodríguez, J. A. (2010). The hybrid flow shop scheduling problem. *European Journal of Operational Research*, *205*(1), 1–18. doi:10.1016/j.ejor.2009.09.024

Sarin, S. C., & Jaiprakash, P. (2007). *Flow Shop Lot Streaming*. Springer.

Zhang, W., Yin, C., Liu, J., & Linn, R. J. (2005). Multi-job lot streaming to minimize the mean completion time in m-1 hybrid flowshops. *International Journal of Production Economics*, *96*(2), 189–200. doi:10.1016/j.ijpe.2004.04.005