

015-0610

Minimizing Mean Tardiness in a Buffer-Constrained Dynamic Flowshop - A Comparative Study

Ahmed El-Bouri

Department of Operations Management and Business Statistics

College of Commerce and Economics

Sultan Qaboos University

PO Box 20, PC 123

Muscat, Oman

POMS 21st Annual Conference

Vancouver, Canada

May 7 to May 10, 2010

### **Introduction**

A flowshop is a production system characterized by a serial arrangement of machines or servers for processing jobs. All jobs visit each of the machines in the same order, but the processing times required on the machines usually differ from one job to another. When a job is completed on a machine, it is transferred immediately to the next machine in sequence. If the machine is busy, the job waits for it in an intermediate buffer. If that buffer is filled to capacity, then the job cannot be transferred, and it remains occupying (blocking) its current machine until buffer space eventually becomes available. A blocked machine remains idle, and cannot process new jobs until the completed job is cleared.

The type of flowshop considered in this study assumes that intermediate buffers all have a common, limited capacity for storing in-process jobs between the machines. Furthermore, jobs are assumed to arrive at the flowshop continuously, at random points in time during the production run. Once a job arrives, its processing time requirements on each of the machines become known. This type of flowshop is considered a dynamic one, due to the arrival of jobs at times which are not known in advance. Scheduling in a dynamic flowshop is characterized by the need to constantly revise production schedules with each new job arrival. A convenient approach under these circumstances is to utilize dispatching rules, which prioritize waiting jobs according to some criterion or criteria. Dispatching rules are favored in dynamic shop environments because of their simplicity, together with the fact that they can be applied in real-time environments.

The problem of minimizing mean tardiness in a flowshop is NP-hard in the strong sense (Rinooy Kan et al., 1976) and Koulamas (1994). As a result, heuristic approaches are frequently investigated (Sen et. al., 1989, Kim 1993). More recently, meta-heuristics that employ tabu search, simulated annealing, genetic algorithms, and differential evolution algorithms have been examined with varying success for the mean tardiness criterion (Vallada et al., 2008). These techniques, however, were investigated only for static flowshops, in which all of the jobs are available at the start of the scheduling horizon. Published research for dynamic flowshops is much less common, and it has revolved mostly around the use of dispatching rules. Although there is a large body of research in dispatching rules, it is concerned overwhelmingly with job shop environments (Panwalker and Iskander 1977, Blackstone et al. 1982 and Haupt 1989) and flexible manufacturing systems (Jain et al., 2004).

The earliest studies of dispatching rules specifically in flowshops (Scudder and Hoffman, 1987, Hunsucker & Shah, 1992, 1994) investigated simple rules such as the shortest processing time (SPT), longest processing time (LPT) and first-in first-out (FIFO) rules. Rajendran & Holthaus, 1999 compared a number of common dispatching rules for several different performance criteria in both flowshop and jobshop environments. In a later paper, Holthaus and Rajendran proposed a number of new rules for dispatching jobs in a limited buffer capacity flowshop. Their results indicated that one of those new rules, identified as A:PT/TIS, outperformed the SPT, EDD (earliest due-date), FIFO and COVERT (cost over time) rules. In addition, a significant number of studies dealing with limited-capacity buffers in flowshops have been published (Leisten, 1990; Witt and Voss 2007), but these considered the static flowshops only. Holthaus and Rajendran's investigation (2002) remains the most relevant research in the literature with respect to the specific research topic considered here in the present study.

The study presented in this paper compares a cooperative dispatching (CD) approach (El-Bouri et al., 2008) with a number of leading dispatching rules, including one of the rules suggested by Holthaus and Rajendran (2002), for minimizing mean tardiness in a limited buffer capacity flowshop. The comparisons are performed by simulating the different dispatching rules on a computer model of a 5-machine flowshop. The model allows buffer capacities to be specified by the user, and the mean tardiness based on all completed jobs may be computed for different buffer sizes.

## Cooperative Dispatching

Consider that a dispatching decision is needed at machine  $s$  in the flowshop, and that the jobs queued for processing at machine  $s$  are represented by the ordered set,  $\Omega$ . Cooperative dispatching consults the downstream machines before deciding which job from  $\Omega$  to load next on machine  $s$ . A consulted machine attempts to influence the dispatching decision such that it may receive the current set of jobs ( $\Omega$ ) in a sequence that minimizes their total tardiness, measured locally at the machine. Local total tardiness at a machine,  $k$ , is determined by treating the machine as a single machine system, with tardiness being measured in reference to operation due-dates. The operation due-date for job  $i$  at machine  $k$ ,  $d_{i,k}$ , is defined by equation (1), where  $p_{i,k}$  is the processing time for job  $i$  on machine  $k$ , and  $D_i$  is due-date for completion of all operations of job  $i$ .

$$d_{i,k} = \left( \frac{\sum_{y=1}^k p_{i,y}}{\sum_{y=1}^m p_{i,y}} \right) \times D_i \quad (1)$$

In computing the local total tardiness at machine  $k$ , it assumed, first, that all in-process jobs between machines  $k$  and  $s$  are cleared in FIFO order. After that, the jobs in set  $\Omega$  are assumed available at machine  $k$ , to be arranged according to a minimum total tardiness sequence. That sequence is obtained by first sorting the jobs ( $\Omega$ ) in order of non-decreasing operation due-dates, and then performing adjacent pairwise interchanges until a local minimum is reached. The lead job in the resulting sequence for  $\Omega$  becomes machine  $k$ 's choice of job for dispatching at upstream machine  $s$ . However, if a different job is dispatched instead, the local total tardiness may increase, as a result of machine  $k$  having a different job now at the head of its preferred sequence for  $\Omega$ . Every consulted machine, therefore, computes the local total tardiness,  $T_{i,k}$ ,

assuming that job  $i \in \Omega$  is dispatched next. This is then weighted according to equation (2), to produce a weighted local total tardiness measure at machine  $k$ ,  $z_{i,k}$ .

$$z_{i,k} = T_{i,k} W_k \alpha^{k-1} \quad (2)$$

$W_k$  is the ratio of a machine's remaining workload (the sum of remaining job processing times) relative to the machine that has the most remaining job processing times. The factor  $\alpha^{(k-1)}$  in equation (2) scales down, exponentially, a machine's influence on the dispatching decision based on its distance downstream. In an earlier paper that studied cooperative dispatching for minimizing mean flowtime (El-Bouri et al. 2008), a value of  $\alpha=0.5$  was found to be suitable. The present study will use the same  $\alpha$  value.

Values of  $z_{i,k}$  are computed according to equation (2) for each of the  $n$  jobs in  $\Omega$ , at each of machines  $s$  through  $k$ . The results are then compiled into a  $n \times (k-s+1)$  matrix,  $\mathbf{Z}$ .

$$\mathbf{Z} = \begin{bmatrix} z_{1,s} & \cdots & z_{1,k} & \cdots & z_{1,m} \\ \vdots & & \vdots & & \vdots \\ z_{i,s} & \cdots & z_{i,k} & \cdots & z_{i,m} \\ \vdots & & \vdots & & \vdots \\ z_{n,s} & \cdots & z_{n,k} & \cdots & z_{n,m} \end{bmatrix}. \quad (3)$$

The entries in column  $k$  of  $\mathbf{Z}$  may be considered as machine  $k$ 's *degree of preference* for each dispatching alternative at machine  $s$ . Machine  $k$ 's top preference is the job associated with the minimum value,  $\mu_k$ , in its respective column. If a different job,  $i$ , is dispatched instead, then machine  $k$  sustains a 'penalty' measured by  $z_{i,k} - \mu_k$ . The sum of the penalties incurred by the consulted machines in the event that job  $i$  is dispatched ( $\rho_i$ ) is computed by.

$$\rho_i = \sum_{k=s}^m (z_{i,k} - \mu_k) \quad i = 1, \dots, n \quad (4)$$

The dispatching candidate which has the minimum value of  $\rho_i$  is the one that is selected and dispatched. Ties are broken in favour of the job which best avoids potential blocking of the machine. Further computational details and a numerical example to illustrate the cooperative dispatching methodology may be found in El-Bouri et al. (2008).

### **Computational Experience**

The performance of cooperative dispatching is evaluated in comparison with a number of competing dispatching rules that have been used in previous studies for minimizing tardiness-based criteria in flowshops and job shops. The selected dispatching rules are:

1. Shortest processing time first (SPT) rule, because of its proven performance under tight due-dates (Haupt, 1989)
2. Earliest due-date (EDD) rule, which dispatches the waiting job that has the lowest  $D_i$ .
3. Modified due-date (MDD) rule (Baker and Bertrand, 1982).
4. The COVERT rule (Carroll, 1965);
5. The A:PT/TIS rule proposed by Holthaus and Rajendran (2002). This rule is oriented towards low capacity buffers, and it seeks to minimize the amount of time machines remain idle (blocked) when completed jobs cannot be unloaded because the buffer at the next machine is filled to capacity. The A:PT/TIS rule is identified henceforth in this study as the APT rule.

Comparison of CD relative to the other dispatching rule is performed by simulation on a 5-machine flowshop. A set of twenty randomly-generated test problems is used. Each test problem consists of 10,000 job arrivals, with processing times for each job randomly generated from the uniform distribution  $U[1,99]$ . The due-date assigned to each job on arrival is a multiple,  $Q$ , of the sum of the job's processing times on the five machines. The arrival time for each job is also randomly generated, but from an exponential distribution, with a mean arrival rate  $\lambda=0.018$  per unit time. The test problems are simulated by using buffer capacities of 2, 5 and 8 units, and for three levels of due-date factor,  $Q=4$ ,  $Q=6$  and  $Q=8$ . Problems tested with a buffer capacity of 2 units represent severe congestion in the shop, while those associated with buffer capacity of 8 units reflect a lower level of job congestion. For a given buffer capacity and due-date factor, each of the twenty test problems is simulated six times, once for each of the five dispatching rules, in addition to a run for CD. The mean tardiness produced in each instance is compiled, and the results are summarized in Table 1 below.

## **Results and Conclusions**

The experimental results from Table 1 clearly show that the average mean tardiness in the test problems obtained by CD is generally lower, by a wide margin, than those produced by the other dispatching rules. In some categories, CD reduces the average mean tardiness by as much as about 85-90% compared to APT, the next closest dispatching rule. However, CD's performance appears to weaken progressively as buffer size increases and due-dates become looser (high  $Q$ ). On the other hand, the huge deterioration in performance exhibited by the other dispatching rules when buffer size is reduced from 5 to 2 units is quite remarkable. The results in Table 1 illustrate

the superiority of CD in adapting to tighter buffer sizes and due-dates, in comparison with the competing job dispatching techniques considered in this study.

Table 1: Average mean tardiness for 20-problem test set

Buffer Size	Due-date Factor (Q)	Dispatching Method					
		APT	CD	COVERT	EDD	MDD	SPT
2	4	6266.1	909.9	13067.7	19641.6	10112.9	13419.8
	6	5807.1	751.9	12644.0	19248.6	9981.2	13316.5
	8	5597.0	615.1	12243.7	19138.4	9873.5	13184.5
5	4	186.4	130.5	209.7	715.6	241.4	290.6
	6	70.6	40.9	73.7	491.5	90.7	217.6
	8	33.0	16.2	31.4	409.6	46.4	172.2
8	4	165.0	117.8	163.4	290.0	194.1	238.0
	6	52.7	37.3	50.6	85.5	57.4	171.0
	8	17.7	14.6	11.0	17.5	11.8	129.0

The study presented in this paper illustrates the significant disadvantage of applying common and popular dispatching rules in a flowshop that has severely limited-capacity buffers. Although the APT rule, which works specifically to minimize machine blockage times, shows a vast improvement over the other traditional dispatching rules for low capacity buffers, it still underperforms substantially when compared with CD. In conclusion, the collaborative method of job dispatching employed by CD produces exceptional results, in comparison to alternative dispatching rules, in the presence of severe capacity limitations in the flowshop's buffers.

Remaining work in this research project will consider the effect of  $\alpha$  from equation (2). The value of  $\alpha = 0.5$  used in this study is based on results from previous research for the objective of minimizing mean flowtime. It is likely here that an optimal value for  $\alpha$  may bear

some relationship to the average due-date tightness for the jobs. Furthermore, the experiments should be expanded to include a flowshop that has more than five machines, to determine how the performance of the CD in minimizing mean tardiness is affected by the size of the shop.

## References

Baker, K. and Bertrand, J., 1982, A dynamic priority rule for scheduling against due-dates.

*Journal of Operations Management*, **3**, 37-42.

Blackstone, J. H., Phillips, D. T. and Hogg, G.L., 1982. A state-of-the-art survey of dispatching rules for manufacturing job shop operations. *International Journal of Production*

*Research*, **20**(1), 27-45.

Carroll, D.C., 1965. Heuristic sequencing of single and multiple component jobs. Ph.D. Thesis,

Sloan School of Management, M.I.T., Cambridge, MA.

El-Bouri, A., Popplewell, N. and Balakrishnan, S., 2008. Cooperative dispatching for minimizing mean flowtime in a dynamic flowshop. *International Journal of Production Economics*,

**113**(2), 819-833.

Haupt, R., 1989. A survey of priority rule-based scheduling. *OR Spektrum*, **11**(1), 3-16.

Holthaus, O. and Rajendran, C., 2002. A study on the performance of scheduling rules in buffer-constrained dynamic flowshops. *International Journal of Production Research*, **40**(13),

3041-3052.

Hunsucker, J.L. and Shah, J.R., 1992. Performance of priority rules in a due date flow shop.

*OMEGA*, **20**(1), 73-89.

- Hunsucker, J.L. and Shah, J.R., 1994. Comparative performance analysis of priority rules in a constrained flow shop with multiple processors environment. *European Journal of Operational Research*, **72**(1), 102-114.
- Jain, A., Jain, P.K. and Singh, I.P., 2004. An investigation on the performance of dispatching rules in FMS scheduling. *International Journal of Simulation Modelling*, **3**(2-3), 49-60.
- Kim, Y. D., 1993. Heuristics for flowshop scheduling problems minimizing mean tardiness. *Journal of the Operational Research Society*, **44**, 19-28.
- Koulamas, C., 1994. Total tardiness problem: review and extensions. *Operations Research*, **42**(6), 1025-1041.
- Leitsten, R., 1990. Flowshop sequencing problems with limited buffer storage. *International Journal of Production Research*, **28**(11), 2085-2100.
- Panwalkar, S.S. and Iskander, W., 1977. A survey of scheduling rules. *Operations Research*, **25**(1), 45-61.
- Rajendran, C. and Holthaus, O., 1999. A comparative study of dispatching rules in dynamic flowshops and jobshops. *European Journal of Operational Research*, **116**(1), 156-170.
- Rinnooy Kan, A. H. G. (1976). Machine scheduling problems: classification, complexity and computations. The Hague:Martinus Nijhoff.
- Scudder, G. D. and Hoffmann, T.R., 1987. The use of cost-based priorities in random and flow shops. *Journal of Operations Management*, **7**(2), 217-232.
- Vallada, E., Ruiz, R. And Minella, G., 2008. Minimising total tardiness in the m-machine flowshop problem: A review and evaluation of heuristics and metaheuristics. *Computers and Operations Research*, **35**(4), 1350-1373.

Witt, A. and Voss, S., 2007. Simple heuristics for scheduling with limited intermediate storage.

*Computers and Operations Research*, **34**(8), 2293-2309.