15th Triennial World Congress, Barcelona, Spain

LOTSTREAMING WITH EQUAL SUBLOTS IN NO-WAIT FLOWSHOPS

Jiyin Liu * **and Edouard Wagneur** **

* *Dept of Ind. Eng. & Eng. Managt., Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong, e-mail: jyliu@ust.hk*
** *Institut de Recherche en Communications et Cybernétique de Nantes, UMR CNRS 6597, Ecole Centrale de Nantes, Université de Nantes, Ecole des Mines de Nantes, BP 92101, 44 321 Nantes Cedex 03, France, e-mail: wagneur@irccyn.ec-nantes.fr*

Abstract:
We consider the problem of minimizing makespan in a no-wait flow-shop with an arbitrary number of machines. We assume that all sublots have the same number of items. We consider the cases with or without loading times. In both cases we show the existence of a critical machine which never remains idle. Then we get a set of general conditions on the processing times and loading times, which bind the number of sublots and the rank of the critical machine. $copyright$ © 2002 $IFAC$.

Keywords: load dispatching, multiprocessors, optimal load flows, optimization problems

1. INTRODUCTION

We consider the problem of lot streaming of a single product through m machines in no-wait flow-shops in order to minimize the makespan. A lot consists of many identical items of the product. Lot streaming is the process of creating sublots to move the completed portion of a production sublot to the next machines. This permits the overlapping of different operations on the same product and may therefore reduce the makespan. In a flowshop, all products or sublots follow the same machine sequence, and each product (eg sublot) has exactly one operation on each machine.
In the no-wait environment, a product (sublot) cannot wait in-between the machines, either because there are no buffers in-between the machines, or because the process does not allow for such processing interruptions. Lotsizing and scheduling problems in no-wait flow-shops arise in chemical processing and petrochemical production environments. Another example of the no-wait situation arises in hot metal rolling industries where the metals have to be processed continuously at high temperature. In such environments, where waiting time induces increased processing time, Wagneur and Sriskandarajah 1993a, 1993b, show that the minimum makespan arises as a trade-off between no-wait (which creates machine idle time) and no idle time for the machines (which creates parts waiting time).

Summaries of literature on lot streaming are given by Baker (1990, 1995), Hall et al, 2000, Potts and Van

Wassenhove (1992), Trietsch and Baker (1989), and Vickson (1995).

The solution where all sublots are equal is not optimal in general (cf Sriskandarajah and Wagneur, 1999, Wagneur 2001), but it is very easy to implement and to manage in practice. It also easily allows for the determination of the optimal number of sublots (cf Goyal 1976), which is usually considered as given in the litterature. Moreover, in many industrial, or chemical processes, the lines have been designed for constant sublot sizes.

The environment considered here for the processing of sublots is *no-wait*: the processing of a sublot on a machine is started as soon as the processing of the sublot on the preceding machine has been completed. This may lead to machine idle times between successive sublots. In recent years, a considerable amount of interest has arisen in no-wait scheduling problems. This interest appears to be motivated as much by applications as by questions of research interest (refer to the survey papers on no-wait scheduling by Hall and Sriskandarajah 1996, Goyal and Sriskandarajah 1988).

The paper is organized as follows. In Section 2, we consider the problem with no loading times prior to the processing of a sublot on the machines.

In Section 3, we analyze the situation when some loading times are required prior to processing each sublot on the machines. This loading time only depends on the machine considered. We show that there is also a critical machine which is never idle. However, this machine may change, as the number of sublots changes. We determine a sequence of intervals for X which relate the index K of the critical machine to some integer interval in which machine K is critical. Then, given that this machine is critical, we find the optimal number n^* of sublots.

A short discussion then concludes the paper.

1.1 *Notations*

X_j : the total number of items in the sublot j; X_j is a rational number.
a_i : the unit processing time on machine i.
n : the number of sublots for the product considered.
$W = \sum_{j=1}^{n} X_j$: the total number of items demanded for the product.
C_{max}: makespan for the single product problem.
I_j^k: machine k idle time prior to the processing of sublot j , $j = 2, \ldots, n$.
τ_i = loading time for machine i , $i = 1, \ldots, m$.

1.2 *Assumptions*

1. All W units of the product are available at time zero.

2. The product can be treated as infinitely divisible.

3. The processing of sublots is *no-wait*.

4. *Consistent* sublots are used to ensure no-wait processing of sublots
(sublot sizes remain the same on all machines).

5. The processing of a sublot is proportional to its size, i.e, the processing time of sublot j on machine i is $a_i X_j$.

6. All sublots have the same size.

2. THE PROBLEM WITHOUT LOADING TIMES

It is not difficult to see, from Figure 1 below (cf also Wagneur, 2001), that the no-wait constraint induces machine idle times. Thus, the makespan C_{max} of a job of size W can be expressed as (1) for a given number n of sublots,

$$C_{max} = a_1 W + \sum_{j=2}^{n} I_j + X_n \sum_{i=2}^{m} a_i \qquad (1)$$

where I_j is the idle time on machine 1 before the processing of sublot j.

The machine one idle times are derived as follows. Let r_j stand for the release time of sublot j, we have: $r_j \geq r_{j-1} + a_1 X_{j-1}$
$r_j + a_1 X_j \geq r_{j-1} + a_1 X_{j-1} + a_2 X_{j-1}$
. . . .
$r_j + X_j \sum_{i=1}^{k-1} a_i \geq r_{j-1} + a_1 X_{j-1} + X_{j-1} \sum_{i=2}^{k} a_i$
. . . .
Hence
$r_j \geq r_{j-1} + a_1 X_{j-1} +$
$$\max_{2 \leq k \leq m} \{0,\ X_{j-1} \sum_{i=2}^{k} a_i - X_j \sum_{i=1}^{k-1} a_i\}$$
Since this includes all the constraints on r_j, and we want to minimize makespan, we have: $r_j = r_{j-1} + a_1 X_{j-1} + \max_{2 \leq k \leq m} \{0,\ X_{j-1} \sum_{i=2}^{k} a_i - X_j \sum_{i=1}^{k-1} a_i\}$, or
since $r_j - (r_{j-1} + a_1 X_{j-1}) = I_j$:

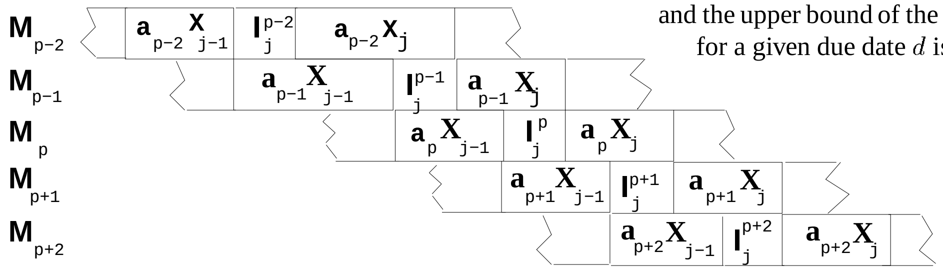

Fig. 1. A schedule for the sequence of sublots

$$I_j = \max_{2 \le k \le m} \left\{ 0, X_{j-1} \sum_{i=2}^{k} a_i - X_j \sum_{i=1}^{k-1} a_i \right\} , \quad j = 2, \ldots, n \qquad (2)$$

Now, since $X_j = X = \frac{W}{n}$, $j = 1, \ldots, n$, we have:
$$I_j = X \max_{2 \le k \le n} \{0, a_k - a_1\}$$
$$= X \max_{1 \le k \le n} \{a_k - a_1\} , \; j = 2, \le \ldots, n .$$

The problem in this case is very simple. We can state the following result.

Proposition 1

If the sublots are required to be equal, then in the optimal solution the slowest machine will never remain idle.

Proof

We have: $C_{max} = a_1 W + (n-1)X \max_{1 \le k \le m} \{a_k - a_1\} + X \sum_{i=2}^{m} a_i = nXa_1 + (n-1)X \max_{1 \le k \le m} a_k - (n-1)Xa_1 + X \sum_{i=2}^{m} a_i$
Let $K = \arg \max_{1 \le k \le n} \{a_k\}$, then
$C_{max} = Xa_1 + (n-1)Xa_K + X \sum_{i=2}^{m} a_i = X \sum_{i=1}^{K-1} a_i + Wa_K + X \sum_{i=K+1}^{m} a_i$

From the above formula, it is clear that the slowest machine is never idle.

The above formula also shows that the makespan decreases with the sublot size. Therefore we have the following result, where $K = \arg \max_{1 \le k \le n} \{a_k\}$.

Proposition 2

When the sublots are equal and there are no loading times, the lower bound of the makespan for producing W units is given by:

$$\underline{C}_{max} = Wa_K ,$$

and the upper bound of the quantity that can be produced for a given due date d is $\overline{W} = \frac{d}{a_K}$.

3. THE MINIMUM MAKESPAN WITH LOADING TIMES

Now we assume that, while remaining in the no-wait environment (eg there are no buffers) a loading time τ_i is required for loading sublot j onto machine i. Thus the processing of a sublot on this machine cannot start before the processing of the sublot on machine $i-1$ finishes, and the sublot has been loaded onto machine i. Then, in this setting, equations (1) and (2) become
$$C_{max} = n\tau_1 + a_1 W + \sum_{j=2}^{n} I_j + \sum_{i=2}^{m} (\tau_i + X_n a_i)$$
$$I_j = \max_{2 \le k \le m} \left\{ 0, X_{j-1} \sum_{i=2}^{k} a_i - X_j \sum_{i=1}^{k-1} a_i + \tau_k - \tau_1 \right\} \quad j = 2, \ldots, n$$

Let $\sum_{i=1}^{o} a_i = 0$, for convenience. As the sublots are equal, $X_j = \frac{W}{n} = X, j = 1, 2, \ldots, n$
$$I_j = I = \max_{1 \le k \le m} \{X(a_k - a_1) + \tau_k - \tau_1\} , \quad j = 2, \ldots, n$$
$$C_{max} = n\tau_1 + a_1 W + (n-1)I + \sum_{i=2}^{m} (\tau_i + Xa_i).$$

We can state:

Proposition 3

When there are loading times, for any given number of sublots n, there is a critical machine that is never idle after starting. This critical machine is the one that maximizes $X(a_k - a_1) + \tau_k - \tau_1$.

Proof

Let $K = \arg \max_{1 \le k \le m} \{X(a_k - a_1) + \tau_k - \tau_1\}$. Then $I = X(a_K - a_1) + \tau_K - \tau_1$

$$C_{max} = n\tau_1 + a_1 W + (n-1)[X(a_K - a_1) + \tau_K - \tau_1] + \sum_{i=2}^{m}(\tau_i + X a_i)$$

$$= \sum_{i=1}^{K-1}(\tau_i + X a_i) + n\tau_K + a_K W + \sum_{i=K+1}^{m}(\tau_i + X a_i)$$

Let I^K stand for machine K idle times. Computing the makespan using machine K, we have :

$$C_{max} = \sum_{i=1}^{K-1}(\tau_i + X a_i) + n\tau_K + a_K W + (n-1)I^K + \sum_{i=K+1}^{m}(\tau_i + X a_i)$$

It follows that $I^K = 0$, i.e. machine K is never idle.

For different values of n, the critical machine may be different. We now try to find special values of n at which the critical machine changes. With these values, all the possible values of n can be divided into some intervals so that the critical machine is the same for all n in an interval. Then we can solve the whole problem by solving the subproblems defined in each interval.
Some machines may never be the critical machine.

Proposition 4

For a machine k, if there is another machine i such that either $a_k < a_i$ and $\tau_k \leq \tau_i$, or $a_k \leq a_i$ and $\tau_k < \tau_i$, then machine k will never be the critical machine.

Proof

Clearly for any positive n, $X(a_k - a_1) + \tau_k - \tau_1 < X(a_i - a_1) + \tau_i - \tau_1$. Therefore machine k will never be the critical machine.

It is clear that, if there are more than one machine having the same unit processing time and the same loading time, then either all become critical for some interval for n, or none of them will become critical. Therefore we only need to consider one of them as the critical machine and we call the others duplicate critical machines. Alternatively, we can assume w.l.o.g. that no two machines have the same unit processing times and loading times. We make this assumption in the sequel.

We now analyze the special values of X (and corresponding n) at which the critical machine changes.

Proposition 5

The possible critical machines can be renumbered as k_1, k_2, ..., so that $\frac{\tau_{k_2}-\tau_{k_1}}{a_{k_1}-a_{k_2}} > \frac{\tau_{k_3}-\tau_{k_2}}{a_{k_2}-a_{k_3}} > \frac{\tau_{k_4}-\tau_{k_3}}{a_{k_3}-a_{k_4}} > \dots$ and k_1 is critical when $X \geq \frac{\tau_{k_2}-\tau_{k_1}}{a_{k_1}-a_{k_2}}$, k_2 is critical when $\frac{\tau_{k_2}-\tau_{k_1}}{a_{k_1}-a_{k_2}} \geq X \geq \frac{\tau_{k_3}-\tau_{k_2}}{a_{k_2}-a_{k_3}}$, k_3 is critical when $\frac{\tau_{k_3}-\tau_{k_2}}{a_{k_2}-a_{k_3}} \geq X \geq \frac{\tau_{k_4}-\tau_{k_3}}{a_{k_3}-a_{k_4}}$, and so on.

Proof

Let k_1, k_2, ..., be such that $a_{k_1} > a_{k_2} > a_{k_3} > \dots$ and $\tau_{k_1} < \tau_{k_2} < \tau_{k_3} < \dots$

For $i = 1, 2, \dots$, let $\xi_{i,i+1} = \frac{\tau_{k_{i+1}}-\tau_{k_i}}{a_{k_i}-a_{k_{i+1}}}\ (>0)$.

If $X \geq \xi_{i,i+1}$, then
$X(a_{k_{i+1}} - a_1) + \tau_{k_{i+1}} - \tau_1 \leq X(a_{k_i} - a_1) + \tau_{k_i} - \tau_1$, i.e., k_i (resp k_{i+1}) is (resp is not) likely to become critical.

If $X \leq \xi_{i,i+1}$, then
$X(a_{k_{i+1}} - a_1) + \tau_{k_{i+1}} - \tau_1 \geq X(a_{k_i} - a_1) + \tau_{k_i} - \tau_1$, i.e., k_{i+1} (resp k_i) is (resp is not) likely to become critical.

From
$\xi_{i,i+2} = \frac{\tau_{k_{i+2}}-\tau_{k_i}}{a_{k_i}-a_{k_{i+2}}} = \frac{\tau_{k_{i+2}}-\tau_{k_{i+1}}}{a_{k_i}-a_{k_{i+2}}} + \frac{\tau_{k_{i+1}}-\tau_{k_i}}{a_{k_i}-a_{k_{i+2}}} = \xi_{i+1,i+2}\frac{a_{k_{i+1}}-a_{k_{i+2}}}{a_{k_i}-a_{k_{i+2}}} + \xi_{i,i+1}\frac{a_{k_i}-a_{k_{i+1}}}{a_{k_i}-a_{k_{i+2}}}$
it follows that $\xi_{i,i+2} \geq \min\{\xi_{i,i+1}, \xi_{i+1,i+2}\}$ and $\xi_{i,i+2} \leq \max\{\xi_{i,i+1}, \xi_{i+1,i+2}\}$, i.e., $\xi_{i,i+2}$ is always between $\xi_{i,i+1}$ and $\xi_{i+1,i+2}$.

Assume that $\xi_{i,i+1} \leq \xi_{i+1,i+2}$ for some i, then for $X \in [\xi_{i,i+1}, \xi_{i+1,i+2}]$, the index k_{i+1} will never become critical. Indeed for $X \leq \xi_{i+1,i+2}$, k_{i+2} is more likely to be critical than k_{i+1} and for $X \geq \xi_{i,i+1}$, k_i is more likely to be critical than k_{i+1}. Hence, the interval of X in which k_{i+1} is potentially critical is empty. So k_{i+1} is not the index of a possible critical machine, and thus can be removed from the list. The remaining machines can be renumbered: $k_{i+2} \to k_{i+1}$, $k_{i+3} \to k_{i+2}$, a.s.o.

We can do this until all the remaining machines are true potential candidates for becoming critical, and $\xi_{i,i+1} > \xi_{i+1,i+2}$ for all i. This completes the proof.

From the above result, we can obtain the special values of X: $\xi_{1,2} > \xi_{2,3} > \dots > \xi_{p-1,p}$, and the corresponding special values of n: $n_{1,2} < n_{2,3} < \dots < n_{p-1,p}$. Clearly, the total number of these values, p, is less than or equal to m.

These special n values divide the interval $]0, \infty[$, into $p+1$ intervals:

$]0, n_{1,2}]$, $[n_{1,2}, n_{2,3}]$, $\dots$, $[n_{p-1,p}, \infty[$.

For each interval, $[n_{i,i+1}, n_{i+1,i+2}]$, the critical machine $K = k_{i+1}$, we define the constrained subproblem

$$\min\ C_{max} = \sum_{i=1}^{m} \tau_i + (n-1)\tau_K + a_K W + (A - a_K)\frac{W}{n} \quad \text{s.t.} \quad n_{i,i+1} < n < n_{i+1,i+2},$$

where $A = \sum_{i=1}^{m} a_i$.

Now we try to solve the subproblem:

$\frac{dC_{max}}{dn} = \tau_K - W\frac{(A-a_K)}{n^2}$
$\frac{d^2C_{max}}{dn^2} = 2W\frac{(A-a_K)}{n^3} > 0,$ for $n \geq 0$

Therefore, the n making $\frac{dC_{max}}{dn} = 0$ is the optimal solution to the continuous version of the subproblem without the constraint.

Solving $\tau_K - W(A - a_K)/n^2 = 0$ we get $n_0 = \sqrt{W(A - a_K)/\tau_K}$

Since C_{max} is a convex function of n, the integer unconstrained solution n^* is either $\lfloor n_0 \rfloor$ or $\lceil n_0 \rceil$, whichever yields the least C_{max}.

If n^* lies within the interval constraint of the subproblem, it is optimal for the subproblem; if it is less than the lower bound of the interval, then the smallest integer in the interval is the optimal solution; if it is greater than the upper bound of the interval, then the largest integer in the interval is the optimal solution. If there is no integer in the interval, then this subproblem is not feasible.

After all these subproblems have been solved (note that at least one subproblem has a feasible solution), then the best solution (ie the one which yields the minimum makespan) will be the optimal solution for the entire problem.

4. CONCLUSION

In industrial practice, the sublot sizes are usually taken to be a constant. However, the number of sublots has to be optimized. We have seen in this paper that the slowest machine never remains idle in case there are no loading times inbetween the processing of two sublots.
In case some loading times are required, then we have shown the existence of a critical machine, whose rank depends on the number of sublots. We show also how the optimal number of sublots can be determined by solving a sequence of subproblems, and the best solution to all these subproblems will yield the optimal solution to the problem.
The results in this paper can be very useful for the optimization of some industrial processes. Moreover, they can be seen as a first, simple approach to the general problem of finding the optimal number of sublots when the sublot sizes are also decision variables (cf Sriskandarajah and Wagneur, 1999).
There are many practical reasons to use equal sublots. However, from a theoretical point of view, if the sublot sizes are not restricted to be equal, the makespan may be further reduced. Therefore, it is interesting to know how far the optimal equal sublot solution is from the optimal general solution.
Usually, in practice customers require a given amount of product Y_j, at some due date D_j , $j = 1, 2, \ldots, N$. Then it is of crucial importance to know if it is possible to satisfy the demand at these due dates. One way of doing this is to process sublots of equal sizes, and to verify that they meet the demand requirements. If this is not the case, then we have to look at the problem with with variable sublots sizes. In this sense, the contribution of the paper may be seen as preliminary contribution towards this goal.

Aknowledgements: This research has been supported by the PROCORE programme.

References

Baker, K.R. 1990. Solution Procedures for the Lot-Streaming Problem, *Decision Sciences,* 21, 475–491.

Baker, K.R. 1995. Elements of Sequencing and Scheduling, book published by the Amos Tuck School of Business Administration, Dartmouth College, Hanover, New Hampshire.

Goyal, S.K. 1976. Note on Manufacturing Cycle Time Determination for a Multi-stage Economic Production Quantity Model, *Management Science*, 23, 332–338.

Goyal, S.K. and C. Sriskandarajah. 1988. No-wait Shop Scheduling: Computational Complexity and Approximate Algorithms, *Opsearch*, 25, 220–244.

Hall, N.G. and C. Sriskandarajah. 1996. A Survey of Machine Scheduling Problems with Blocking and No-wait in Process, *Operations Research*, 44, 510–525.

Hall, N.G., Laporte, G., Selvaraj, E., and Sriskandarajah, C., Scheduling and Lotstreaming in Machine Shops with No-Wait In Process, Working paper, Feb 2000, the University of Texas at Dallas.

Potts, C.N. and K.R. Baker. 1989. Flowshop Scheduling with Lot Streaming, *Operations Research Letters*, 8, 4, 297–303.

Potts, C.N. and L.N. Van Wassenhove. 1992. Integrating Scheduling with Batching and Lot-sizing : A Review of Algorithms and Complexity, *Journal of Operational Research Society*, 43, 5, 395–406.

Sriskandarajah, C., and Wagneur, E., 1999. Lotstreaming and Scheduling Multiple products in two-machine No-Wait Flow-Shops, IIE Transactions, Scheduling and Logistics, 31, 8, 695-708.

Trietsch, D. and K. R. Baker. 1993. Basic Techniques for Lot Streaming, *Operations Research*, 41, 6, 1065–1076.

Vickson, R.G. 1995, Optinmal Lotstreaming for Multiple Products in a Two-Machine Flow-Shop, *European Journal of Operations Research*, 85, 556-575.

Wagneur and Sriskandarajah 1993a. The 2-Machine Permutation Flow-Shop with State Dependent Processing Times, Naval Res. Logistics, 40, 697-717.

Wagneur and Sriskandarajah 1993b. Optimal Control of a Class of Discrete Event Systems, Journal of Dynamic Discrete Event systems, 3, 397-425.

Wagneur, E., 2001. Lotstreaming single product in 3-machine no-wait Flow-Shop, Decision & Control in Management Sciences" in honor of Professor Alain Haurie, G. Zaccour Ed., Kluwer, 319-334.