LOT STREAMING IN MULTI STAGE SHOPS

A THESIS
SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL
ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCES
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Alper Şen
December, 1994

LOT STREAMING IN MULTI STAGE SHOPS

A THESIS
SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL
ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCES
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Alper Şen
December, 1994

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Ömer S. Benli (Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. M. Akif Eyler

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Mustafa Akgül

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Selim Aktürk

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Cemal Akyel

Approved for the Institute of Engineering and Sciences:

Prof. Mehmet Baray
Director of Institute of Engineering and Sciences

ABSTRACT

LOT STREAMING IN MULTI STAGE SHOPS

Alper Şen
M.S. in Industrial Engineering
Supervisor: Assoc. Prof. Ömer S. Benli
December, 1994

In this thesis, a number of lot streaming problems in flow, open and job shops are investigated. *Lot streaming* is the process of splitting a job to allow for overlapping of its operations on various machines resulting in shorter completion times. When there is a single job, the problem is to find the size of the transfer batches ("*sublots*") which minimizes a given performance measure (e.g., makespan, mean flow time). Multi-job problems are harder, since sequencing and sizing decisions must be made simultaneously. Most of the current research in lot streaming is concerned with minimum makespan problems in flow shops. In this study, other performance measures and shop structures are also analyzed. Optimal sublot sizes are derived for the single job two machine flow shop mean flow time problem. Solution methods are proposed for the minimum makespan problem in open shops both for multiple job and single job cases.

Key words: Scheduling, Lot Streaming, Flow Shops, Open Shops, Job Shops

ÖZET

ÇOK MAKİNALI ATELYELERDE KAFİLE AKTARMA

Alper Şen
Endüstri Mühendisliği Bölümü Yüksek Lisans
Tez Yöneticisi: Doç. Dr. Ömer S. Benli
Aralık, 1994

Bu çalışmada çok makinalı atelyelerde kafile aktarma problemleri incelenmiştir . *Kafile aktarma* bir işin bölünerek değişik makinalarda işlemlerinin çakıştırılması yoluyla akış zamanlarının azaltılmasıdır. Sadece bir tek iş olduğunda, problem, verilen performans ölçütünü enazlayan transfer kafilelerinin büyüklüklerinin bulunmasıdır. Sıralama ve büyüklük kararları eşgüdümlü alınması gerektiğinden, çok işli problemlerin çözümü daha güçtür. Bu konuda yapılan araştırmaların çoğunluğu akış tipi atelyelerde çizelge uzunluğu problemlerini incelemektedir. Bu çalışmada ise, değişik performans ölçütleri ve atelye tipleri incelenmektedir. Tek işli, akış tipi, iki makinalı atelyelerde ortalama akış süresini enazlayan transfer kafilesi büyüklükleri hesaplanmaktadır. Çok işli ve çok makinalı atelyelerde çizelge uzunluğu problemleri için çözüm yöntemleri önerilmiştir.

Anahtar sözcükler: Çizelgeleme, Kafile Aktarma, Atelye Tipi Üretim

To my family

ACKNOWLEDGEMENT

I am very grateful to my supervisor, Associate Professor Ömer S. Benli for his invaluable guidance and motivating support during this study. His instruction will be the closest and most important reference in my future research.

I would like to thank to Assistant Professor Cemal Akyel for his efforts, guidance and encouragement in this research.

I am indebted to Professor M. Akif Eyler, Associate Professor Mustafa Akgül and Assistant Professor M. Selim Aktürk for their keen interest in the subject matter and valuable comments.

I would also like to thank to my classmates A. Aydın Selçuk and Abdullah Daşçı for their enlightening discussions.

Special thanks are due to Engin Topaloğlu, for his extensive cooperation.

Finally, I would like to thank to my family and everybody who has in some way contributed to this study by lending moral support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In classical scheduling theory, the job's integrity is preserved while it is processed and transferred. However, especially in batch manufacturing, it is practical to move some portion of the job to the downstream machine, before it is entirely completed on the current machine. *Lot streaming* is the creation of these transfer lots for a job, so that its operations can be overlapped on various machines. Lot streaming is applied by means of *sublots*, which are the groups of items that are transferred from one machine to the next at once. Overlapping operations give the opportunity to start processing earlier on the downstream machines to achieve shorter completion times.

Consider the example, in which we have only two machines and a single job that consists of 100 identical units. Each unit requires processing of 2 minutes on the first machine and 3 minutes on the second machine. If lot streaming is not allowed, the job can be completed in 500 minutes (Figure 1.1.a). But, by simply transferring 50 units (half of the job) to the second machine, after they are complete on the first machine, it is possible to complete all the units in 400 minutes. We can also deliver these 50 units as soon as they are processed on machine 2. Hence 50 units will be delivered at time 250 and the remaining 50 will be delivered at time 400, resulting in an average completion time of 325 minutes (Figure 1.1.b), as compared to 500 minutes in the no lot streaming case. We are further able to reduce completion time to 380 minutes and average completion time to 308 minutes, using sublot sizes 40 and 60 (Figure 1.1.c).

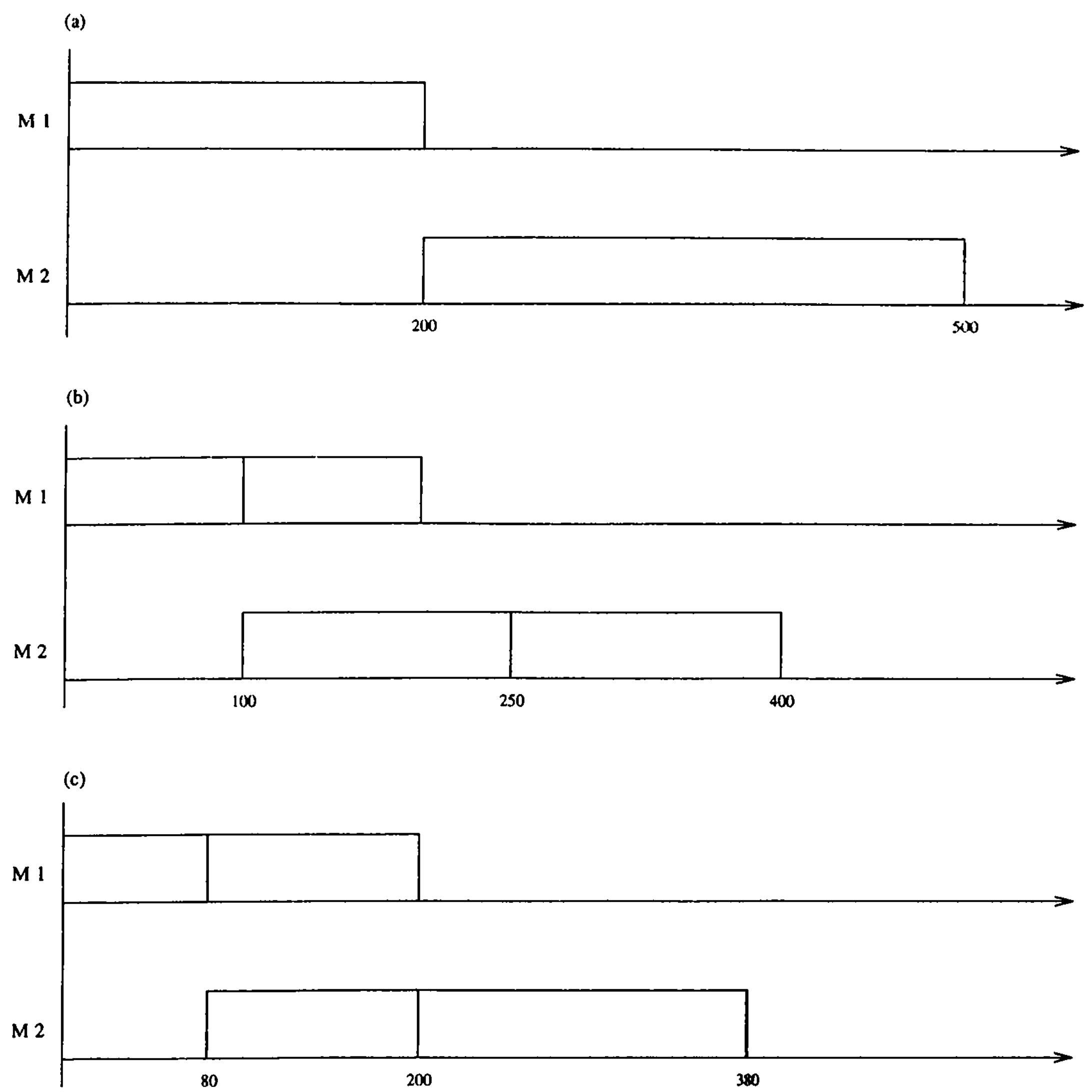

Figure 1.1: Reducing flow times through lot streaming

The use of sublots to accelerate operations is an important aspect in OPT systems. Umble & Srikanth [32], Lundrigan [17] and Browne et. al. [6] discuss that one of the key elements in OPT systems is the distinction between the process and transfer batches. "The transfer batch may not, and many times should not, be equal to the process batch". Fogarty et. al. [11] state the importance of transfer batches in the context of *drum-buffer-rope scheduling*.

Jobs should be streamed on the *non-bottleneck* machines to enable the *bottleneck* machines to start their work as early as possible. The transfer of items is easily maintained by the use of resources (workers, material handling equipment) at non-bottlenecks. Fogarty et. al. [11] also discuss that reducing sublot sizes (thus, increasing number of transfers) may be more efficient, than forcing process batch sizes to equal one as in JIT systems. Swann [25] and Vollmann [35] argue that conventional MRP techniques are no longer applicable, if overlapping operations are allowed. If parts are expedited by use of sublots, there is a need for designing (or revising) scheduling algorithms to get the possible benefits of OPT philosophy in an MRP system.

Overlapping of operations in scheduling is first considered by Mitten [18]. He proposed an algorithm to sequence multi jobs in a two-stage flow shop, in which each job may start processing on the second machine, a certain amount of time after it has started processing on the first machine.

Szendrovits [26] allowed for equal sized transfers between the stages and proposed a model to minimize the sum of setup, finished products inventory and work-in-process inventory costs, while meeting the continuous demand. The Economic Production Quantity of identical items that he optimized is processed uninterrupted on all machines. Truscott [31] studied the case where the sublot sizes can be multiples of a certain number and developed a model to minimize makespan in the presence of setup times, equal sized transfers and transfer times, again with the restriction that the machines should work continuously.

Baker [1] and Trietsch [29] relaxed the assumption that the sublots should be equal sized and proposed solution procedures to find the sublot sizes which minimize the makespan of a single job, with exogenously assigned maximum number of sublots. Since then, there is a considerable interest in lot streaming problems, of which the related portions are reviewed in the succeeding chapters.

The following section defines the lot streaming problem along with various models and restrictions.

1.1 Problem Definition

A resource that performs at most one activity at a time is called a *machine*. A *shop* is a collection of machines. An m–machine shop consists of m machines, $M_1, M_2, \ldots, M_m$. The activities are called *jobs*. There are n jobs, $J_1, J_2, \ldots, J_n$. Each J_j has m operations $O_{1j}, O_{2j}, \ldots, O_{mj}$. O_{ij} has a processing of duration P_{ij} to be performed on M_i. No two operations can be processed simultaneously on a machine. A *routing* $R = (M_{[1]}, M_{[2]}, \ldots, M_{[m]})$ for a job is the order of machines that will process the job. If this order is fixed for all jobs, the shop is called a *flow shop*. In an *open shop*, there are no such restrictions. In *job shops*, each job may require more than m operations (hence each job may require same machine at different stages of its processing). Each job has distinct but a fixed routing in job shops. A job J_j consists of U_j identical units. Hence, each operation O_{ij} is composed of U_j identical sub-operations, each of length $p_{ij} = P_{ij}/U_j$.

For a job, the group of units that are transferred at the same time from one machine to the next machine in the routing, forms a sublot of that job. For each M_i and for each J_j, there can be at most s_{ij} sublots. We assume that the number of sublots is fixed in the shop for each job, i.e. $s_{ij} = s_j$, $i = 1, \ldots, m$. In one extreme, $s_j = U_j$ for each j, which implies a continuous flow production line, if the shop is a flow shop. In the other extreme, $s_j = 1$ for each j, which implies a classical scheduling model where each job's integrity is preserved while it is transferred. The processing time of the kth sublot for J_j on M_i is $p_{ij}L_{ijk}$, where L_{ijk} denotes the number of units in sublot. Clearly, $\sum_{k=1}^{s_j} L_{ijk} = U_j$ for each J_j on each M_i. C_{ijk} is the completion time of the kth sublot of J_j on M_i. A job is completed, if all of its sublots are completed on all machines, that is, the completion time of J_j, $C_j = \max_{i,k} C_{ijk}$.

If the number of units that form a sublot remains the same throughout the shop, then the sublots are called *consistent*, i.e. $L_{ijk} = L_{jk}$ for each M_i. Otherwise, they are called *variable* sublots. The size of the sublots may be restricted to take integer values, i.e. *discrete* case or the job can be assumed to be infinitely divisible, i.e. *continuous* case.

If each J_j is processed P_{ij} consecutive time units on M_i, over the time the machine is busy, then the shop is called a *non-preemptive* shop. If jobs can be processed with interruptions to allow for processing of units of a some other job, then the shop is called a *preemptive* shop. The shop is still a non-preemptive shop if the processing of a job is interrupted, but the machine is idle during the interruption. In any of these models, we do not allow for interruption of sublots on any machine. If a machine is not allowed to have idle time from the start of its first operation to the completion of its last operation, we say that the model is a *continuous work* model. Otherwise, we say that *intermittent idling* is allowed.

There may be several objectives, depending on the completion times of individual units (*items*), sublots or jobs. The *job completion time* may be critical for a system, in which each job is delivered as a whole. Items in a sublot may be assumed to be completed, when the sublot to which they belong is completed, resulting in a *sublot completion time* model. In *item completion time* models, each item is completed as soon as its operations are completed on last machine.

Under these models, the objective is a *regular measure of performance*, i.e. a monotone non-decreasing function of completion times. This may be the *makespan*, i.e. the time at which all the jobs (with all of their sublots and units) are completed, $C_{max} = \max_j C_j$. *Total flow time* can be another objective, where we want to minimize the sum of job completion times, $\sum_{j=1}^{n} C_j$. When the sublot completions are of concern, it is reasonable to weigh the completion time of each sublot with the number of units in it. That is, mean completion time of a job is, $\sum_{k=1}^{s_j} C_{[m]jk} L_{[m]jk}$, where $[m]$ is the last machine in the routing of J_j. The objective can be easily revised for item completion time model. Similarly, all other relevant objectives, as well as the other elements of the theory of classical scheduling can be adapted to the lot streaming models.

The problem is to find the sizes (and routings if the shop is an open shop) of the sublots, and their sequence on machines so as to minimize the given objective function, subject to the restrictions mentioned above.

The purpose of this study is to propose solution methods for some of the untouched lot streaming problems. Chapter 2 presents the characteristics of the single job problem along with an extensive review of literature. The main contribution of this chapter is the Section 2.1.2 where we solve the two-machine mean flow time problem under sublot completion time model. The problem of routing and streaming a single job in an open shop is studied in Section 2.2. In Chapter 3, the multi-job lot streaming models are reviewed and studied. Section 3.2 and Section 3.3 present the first studies on streaming multi jobs in open shops and job shops. While different models in the literature are also reviewed, our derivations depend on the following assumptions.

- All units in a job are available at time 0.
- Processing times are known.
- Jobs have zero setups.
- Material handling equipment is not a constraint neither in availability nor in capacity, except that the maximum number of sublots is limited.
- Transfer times are zero.
- Jobs are infinitely divisible, i.e., sublot sizes may not be integer.

Chapter 4 discusses the main results of the thesis and directions for further research.

Chapter 2

Single Job Models

Although their application areas may be limited, single job lot streaming models can be useful in understanding the nature of multi-job problems. They can also be utilized as a subproblem in exact or heuristic procedures to solve the multi-job problems. The research in single job problem concentrates in and is initiated by the flow shop problems with consistent sublots to minimize makespan. In this chapter, since single job models are analyzed, subscript j is omitted in variable definitions.

2.1 Flow Shop Models

2.1.1 Formulations

The basic lot streaming problem was first introduced by Baker [1]. In this problem, the sublot sizes are assumed to be consistent, i.e. $L_{ik} = L_k$ for each machine M_i, so that the integrity of the sublot is preserved throughout the shop. The objective selected is the minimization of makespan. It is a convenient measure to observe the flow time reductions through lot streaming.

For each sublot, we have two types of constraints to be satisfied. Operation of the sublot k cannot start on M_i, before the sublot $(k-1)$ is completed on M_i. The start of this sublot is also restricted by its completion in the previous machine, M_{i-1}. With these constraints, the linear program to minimize makespan can be stated as,

$$\begin{aligned}
&\min\ C_{ms} && && (2.1)\\
\text{subject to}\quad & C_{ik} &\geq\ & C_{i,k-1}+p_iL_k,\ \ i=1,\ldots,m,\ \ k=1,\ldots,s, & (2.2)\\
& C_{ik} &\geq\ & C_{i-1,k}+p_iL_k,\ \ i=1,\ldots,m,\ \ k=1\ldots,s, & (2.3)\\
& \sum_{k=1}^{s} L_k &=\ & U, & (2.4)\\
& C_{ik} &\geq\ & 0,\ \ k=1,\ldots,s,\ \ i=1,\ldots,m, & (2.5)\\
& L_k &\geq\ & 0,\ \ k=1,\ldots,s, & (2.6)\\
& C_{0k} &=\ & 0,\ \ k=1,\ldots,s, & (2.7)\\
& C_{i0} &=\ & 0,\ \ i=1,\ldots,m. & (2.8)
\end{aligned}$$

Rather than minimizing makespan, the average time a unit spends in the shop can also be a measure of performance. The basic assumption that all the units in the job are completed, only when the whole job is completed may be a restrictive assumption. Customer service may be improved if we do not wait until the whole job is processed [21]. Assuming that each sublot is delivered as soon as its processing is completed in the shop ("sublot completion time model"), we have the objective of minimizing sum of sublot completion times, where each sublot is weighed by its size. The resulting model is a quadratic program with the objective function

$$\min \sum_{k=1}^{s} L_k C_{mk} \qquad (2.9)$$

subject to constraints (2.2)-(2.8). This quadratic objective function is first proposed by Kropp and Smunt [16].

Items can also be delivered, as soon as their processing is complete on the last machine ("item completion time model"). Suppose that there are s sublots transferred from M_{m-1} to M_m. Assume as if the last machine M_m processes s sublots. If C_{mk} is the completion time of the processing of sublot k on the last machine, $C_{mk} - p_m L_k$ will be the starting time of sublot k on M_m. Since we assume that the number of units in a sublot can be fractional, the mean completion time of a unit in sublot k will be, $(C_{mk} + C_{mk} - p_m L_k)/2$. Hence, the consistent sublots formulation to minimize the mean flow time under item completion time will be,

$$\min \sum_{k=1}^{s} L_k C_{mk} - \frac{1}{2} L_k^2 p_m \tag{2.10}$$

subject to constraints (2.2)-(2.8).

An extension of this model, under the consistency assumption, can be used to minimize the number of tardy units. Suppose there is a due date, d, for the job, and the problem is to complete as many units as possible by this due date. A unit is tardy, if the sublot to which it belongs is tardy. If the optimal makespan, C^*_{ms}, is less than or equal to d, then we are done, there are no tardy units. Otherwise, i.e., if $C^*_{ms} > d$, then append the constraint

$$C_{m,s-1} = d \tag{2.11}$$

and optimize the objective function,

$$\min L_s,$$

subject to the Constraints (2.2)- (2.8), and (2.11).

When we allow for variable sublot sizes, simple lot streaming models are no longer applicable. The assumption that the number of sublots remains the same through the shop may be unrealistic in many production systems. These considerations lead to a systematically different model proposed by Benli [5]. This model is a periodic review model with variable period lengths, which are decision variables. The total number of transfers is $h = \sum_{i=1}^{m} s_i$, where s_i is the number of transfers allowed from machine M_i to machine M_{i+1}. The periods are denoted by $[T_t, T_{t+1}]$, where $T_1, T_2, \ldots, T_h$ are the times at which transfers may take place. Define,

$X_{i,t}$: Number of units produced on machine i in $[T_{t-1}, T_t]$,
$L_{i,t}$: Number of units transferred to machine $i+1$ at time T_t,
$I_{i,t}$: Number of units in the input buffer of machine i at time T_t,
$O_{i,t}$: Number of units in the output buffer of machine i at time T_t,

$$Y_{i,t} = \begin{cases} 1 & \text{if } L_{i,t} > 0, \\ 0 & \text{if } L_{i,t} = 0. \end{cases}$$

Note that on any machine M_i, production can take place only in periods $i, \ldots, h-m+i$, since the at least the first $i-1$ and last $m-i-1$ periods will be used for the transfer of products from machines $M_1, M_2, \ldots, M_{i-1}$ and $M_{i+1}, M_{i+1}, \ldots M_m$, respectively. Then, we have the following mixed integer linear program to minimize makespan,

$$\min T_h \tag{2.12}$$

subject to

$$\begin{aligned}
I_{i,t-1} + L_{i-1,t-1} &= I_{i,t} + X_{i,t},\ i = 1, \ldots, m,\ t = i, \ldots, h-m+i, && (2.13) \\
O_{i,t-1} + X_{i,t} &= O_{i,t} + L_{i,t},\ i = 1, \ldots, m,\ t = i, \ldots, h-m+i, && (2.14) \\
p_i X_{i,t} &\le T_t - T_{t-1},\ i = 1, \ldots, m,\ t = i, \ldots, h-m+i, && (2.15) \\
L_{i,t} &\le \mu Y_{i,t},\ i = 1, \ldots, m,\ t = i, \ldots, h-m+i, && (2.16) \\
\sum_{t=1}^{h} Y_{i,t} &\le s_i,\ i = 1, \ldots, m, && (2.17) \\
T_t &\ge 0, t = 1, \ldots, h, && (2.18) \\
I_{i,t}, O_{i,t}, L_{i,t}, X_{i,t} &\ge 0,\ i = 1, \ldots, m,\ t = i, \ldots, h-m+i, && (2.19) \\
Y_{i,t} &\in \{0, 1\},\ i = 1, \ldots, m,\ t = i, \ldots, h-m+i, && (2.20)
\end{aligned}$$

where $I_{i,i-1} = I_{i,h-m+i} = O_{i,i-1} = O_{i,h-m+i} = 0$, $L_{0,0} = L_{m,h} = U$ and μ is a very large number or the capacity of the material handling equipment. The Constraints (2.13) and (2.14) are the inventory balance equations for the input and output buffers. Machine capacity constraints are (2.15). Constraints

(2.16) indicate whether a transfer takes place from a machine M_i at time T_t. Constraints (2.17) limit the number of transfers (sublots) at each stage. The formulation is adaptable to other problems like mean flow time minimization. Basic results of the lot streaming problem can also be obtained through the restriction of the general model.

2.1.2 Two-Machine Problem

Minimizing Makespan

When the sublot sizes are consistent, we have the following linear program to solve the minimum makespan problem,

$$\min C_{2s} \tag{2.21}$$

$$\begin{aligned}
\text{subject to} \quad C_{ik} &\geq C_{i,k-1} + p_i L_k, \ i = 1,2, \ k = 1,\ldots,s, && (2.22)\\
C_{2k} &\geq C_{1k} + p_2 L_k, \ k = 1,\ldots,s, && (2.23)\\
\sum_{k=1}^{s} L_k &= U, && (2.24)\\
C_{ik} &\geq 0, \ i = 1,2, \ k = 1,\ldots,s, && (2.25)\\
L_k &\geq 0, \ k = 1,\ldots,s, && (2.26)\\
C_{i0} &= 0, \ i = 1,2, && (2.27)
\end{aligned}$$

This problem was studied by Baker [1] and Potts & Baker [20]. Baker [1] used the LP formulation to derive the solution. Potts & Baker [20] showed that the makespan is equal to the sum of

- The processing time of sublots $1,\ldots,k$ on M_1, and
- The processing time of sublots $k,\ldots,s$ on M_2

for any sublot k and hence each sublot is "*critical*". The solution is given by the "*geometric*" sublot sizes, i.e.,

$$L_1 = U\frac{1-\pi}{1-\pi^s} \tag{2.28}$$

$$L_k = \pi L_{k-1}, \ k = 2, \ldots, s, \tag{2.29}$$

where $\pi \equiv p_2/p_1$.

The discrete version of the problem is studied by Trietsch [29]. He proposed an iterative algorithm of time complexity $O(s)$ to find the optimal integer sublot sizes. Trietsch & Baker [30] analyzed the cases, where the transportation times in between machines are not negligible and the transporters have limited capacity.

VARIABLE SUBLOTS

If we allow for variability in the sublot sizes in each stage, one would expect that any regular measure of performance would improve. This is simply based on the fact that, for these measures of performance, *consistent* sublots are subsets of *variable* sublots. The question is the following: when is it sufficient to consider only the *consistent* sublots in the search of optimal (variable) sublots? When the objective is the minimization of makespan, Trietsch & Baker [30] state that it is not necessary to consider the variable sublot sizes since there is is only one set of transfers. Note that, here the transfer of items from second machine is not considered.

EQUAL SUBLOTS

The optimal solutions for the two-machine flow shop problems result in different sublot sizes. However, it may be more practical to use equal sublot sizes. In this section, we will compare makespan obtained by using equal sublot sizes, $F^E(L)$, with the optimal makespan, $F^*(L)$, using the ratio, $F^E(L)/F^*(L)$. For notational convenience, we shall assume that $U = 1$, $p_1 = 1$ and $p_2 = \pi$.

When equal sublots are used, makespan is.

$$F^E(L) = \max\{1/s + \pi . 1 + \pi/s\}.$$

On the other hand, optimal makespan is,

$$F^*(L) = \frac{(\pi - 1)}{(\pi^s - 1)} + \pi.$$

Potts & Baker [20] have shown that,

$$F^E(L)/F^*(L) < 1.09.$$

Minimizing Mean Flow Time under Sublot Completion Time Model

Suppose, an item leaves the shop when the sublot to which it belongs is completed in the last stage. In a 2−machine flow shop, the flow time of all units in the job will sum up to $\sum_{k=1}^{s} L_{2k}C_{2k}$. This is equivalent to mean flow time, which is the average time a unit spends in the shop, $\frac{1}{U}\sum_{j=1}^{s} L_{2k}C_{2k}$. Thus the problem, with consistent sublots, becomes a quadratic programming problem with the objective function

$$\sum_{k=1}^{s} L_k C_{2k}, \tag{2.30}$$

subject to Constraints (2.22)–(2.27).

An efficient solution procedure, proposed for the two-stage flow shop problem with consistent sublots is given below. In this problem one has to consider two cases: (i) $\pi \leq 1$, and (ii) $\pi > 1$, where $\pi \equiv p_2/p_1$. Çetinkaya & Gupta [8], independently, obtained the same result for the first case, and they conjectured but not proved the result for the second case.

CASE I : $\pi \leq 1$

As discussed in Şen et. al., [27], consider the general case. There are m machines, with the property $p_1 \geq \max_{2\leq i\leq m}\{p_i\}$, and we will show that equal sublot sizes (i.e. $L_k = U/s$ $k = 1,\ldots,s$) are optimal.

We first need the following result showing that there exists an optimal solution with nondecreasing sublot sizes.

Result 1 *If* $p_1 \geq \max_{2\leq i\leq m}\{p_i\}$ *then an optimal solution exists where;*

$$L_k \leq L_{k+1},\ k = 1,\ldots,s. \tag{2.31}$$

To prove this result, Şen et. al. [27] showed that any schedule that does not satisfy (2.31) can be converted to a schedule which satisfies (2.31) without increasing the mean flow time. Suppose we are given the sublot sizes $\overline{L} = (\overline{L}_1, \ldots, \overline{L}_s)$ which are claimed to be optimal and for at least one k , $\overline{L}_k > \overline{L}_{k+1}$. An iterative procedure is designed for achieving a schedule which satisfies (2.31). At each iteration v, maximum sized sublot among the first $s-v$ sublots is replaced at the $(s-v)$ th position in the schedule. In s iterations, the resulting schedule satisfies (2.31). It is also shown that at each iteration, the mean flow time does not increase.

Çetinkaya & Gupta [8] proved the same result using the following Lemma by Miyazaki & Nishiyama [19],

Lemma 1 *For the ordinary flow shop problem (without lot streaming) to minimize weighted flow time ($\sum w_j C_j$), job h precedes job ℓ in the optimal schedule if,*
i) $w_h \leq w_\ell$
ii) $w_h \sum_{r=i}^{m} p_{r,h} \leq w_\ell \sum_{r=i}^{m} p_{r,\ell}$, $i = 1, \ldots, m$
where, w_j *is the weight of job* j.

Consider our problem as a weighted flow time problem, with sublots considered as jobs. The processing time of job k on machine i is $p_i L_k$ and weight of job k, $w_k = L_k$. Note that,

$$L_k \leq L_\ell \Rightarrow L_k \sum_{r=i}^{m} p_r L_k \leq L_\ell \sum_{r=i}^{m} p_r L_\ell.$$

It is easy to see that sublot k precedes sublot ℓ if $L_k \leq L_\ell$. Then, the result follows.

With this property, the following formulation with a convex function and fewer constraints can be obtained. Assuming $U = 1$,

$$\min \sum_{k=1}^{s} L_k C_{mk}$$

$$\text{subject to } \; C_{mk} = p_1 \sum_{\ell=1}^{k} L_\ell + L_k \sum_{v=2}^{m} p_v, \quad k = 1, ..., s,$$
$$\sum_{k=1}^{s} L_k = 1,$$

equivalently,

$$\min \sum_{k=1}^{s} L_k (p_1 \sum_{\ell=1}^{k-1} L_\ell + L_k \sum_{v=1}^{m} p_v)$$
$$\text{subject to} \quad \sum_{k=1}^{s} L_k = 1.$$

Result 2 *An optimal solution to the above problem is*

$$L_k = \frac{1}{s}, \qquad k = 1, ..., s.$$

Proof: Let the Lagrangian function be,

$$\mathcal{L}(L_1, ..., L_s, \delta) = \sum_{k=1}^{s} L_k (p_1 \sum_{\ell=1}^{k-1} L_\ell + L_k \sum_{v=1}^{m} p_v) + \delta(\sum_{k=1}^{s} L_k - 1),$$

$$\text{then } \frac{\partial \mathcal{L}}{\partial L_k} = p_1 \sum_{\ell=1}^{s} L_\ell + 2L_k \sum_{v=1}^{m} p_v - p_1 L_k + \delta = 0, \text{ and}$$
$$\frac{\partial \mathcal{L}}{\partial \delta} = \sum_{k=1}^{s} L_k - 1 = 0$$

Since,

$$L_k = (-\delta - p_1 \sum_{\ell=1}^{s} L_\ell)/(2 \sum_{v=1}^{m} p_v - p_1), \quad k = 1, \ldots, s$$

$\sum_{k=1}^{s} L_k = 1$ implies that $L_k = \frac{1}{s}$ is the candidate optimal solution. However, to prove that it is the desired solution, we have to show that the objective function is convex. The Hessian matrix of the objective function is,

$$H = \begin{bmatrix} a & b & b & b & b & .. \\ b & a & b & b & b & .. \\ b & b & a & b & b & .. \\ b & b & b & a & b & .. \\ b & b & b & b & a & .. \\ . & & & & & . \\ . & & & & & . \end{bmatrix}$$

where $a = 2\sum_{i=1}^{m} p_i$, and $b = p_1$.

In order for the objective function to be convex, the Hessian matrix should be positive definite. In a positive definite matrix, every upper left sub-matrix should have positive determinant. Let $H_1, \ldots, H_r, \ldots, H_s = H$ be the upper left sub-matrices of H. The determinant of H_r can be found to be,

$$\det H_r = b^r(\frac{a}{b} - 1)^{r-1}(\frac{a}{b} + r - 1) \tag{2.32}$$

since $a > b$. It is clear that, $\det H_r > 0$, $r = 1, \ldots, s$.

CASE II : $\pi > 1$

Again assume, without loss of generality, that $U = 1$ and the processing time of the job is 1 on the first machine and $\pi (= p_2/p_1)$ on the second machine. Since $p_2 > p_1$, we have $\pi > 1$.

Result 3 *When $\pi > 1$, $\pi L_k \geq L_{k+1}$, $k = 1, \ldots s-1$, in an optimal schedule.*

Proof : Suppose the contrary, i.e., there exists an optimal solution $\overline{L} = (\overline{L}_1, \ldots, \overline{L}_s)$ such that, at least for one k, $\pi \overline{L}_k < \overline{L}_{k+1}$. Let $v = \min_{1 \leq k \leq s-1}\{k \mid \pi \overline{L}_k < \overline{L}_{k+1}\}$. A new solution can be constructed for some $\epsilon > 0$, as

$$\begin{array}{lcll} \hat{L}_k & = & \overline{L}_k, & k = 1, \ldots, v-1 \\ \hat{L}_v & = & \overline{L}_v + \epsilon, & \\ \hat{L}_{v+1} & = & \overline{L}_{v+1} - \epsilon, & \\ \hat{L}_k & = & \overline{L}_k, & k = v+2, \ldots, s. \end{array}$$

It is sufficient to show that the new solution, $\hat{L}$ is feasible and $F(\hat{L}) < F(\overline{L})$.

Since $\sum_{k=1}^{v+1} \hat{L}_k = \sum_{k=1}^{v+1} \overline{L}_k$, and $\hat{C}_{1,v+1} = \overline{C}_{1,v+1} = \sum_{k=1}^{v+1} L_k$, for feasibility, it is enough to show

$$\hat{C}_{2,v+1} \leq \overline{C}_{2,v+1}. \tag{2.33}$$

We will now show that (2.33) holds and $F(\hat{L}) < F(\overline{L})$ for the following two possible cases.

Case 1 : $\overline{C}_{1,v+1} > \overline{C}_{2,v}$ (See Figure 2.1 and Figure 2.2).

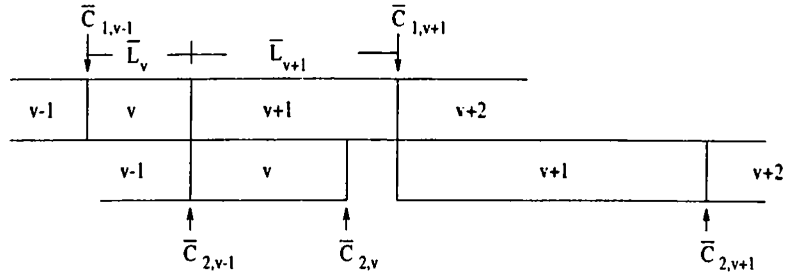

Figure 2.1: Sublot completion, Case 1: $\overline{L} = (\overline{L}_1, \ldots, \overline{L}_v, \overline{L}_{v+1}, \ldots, \overline{L}_s)$

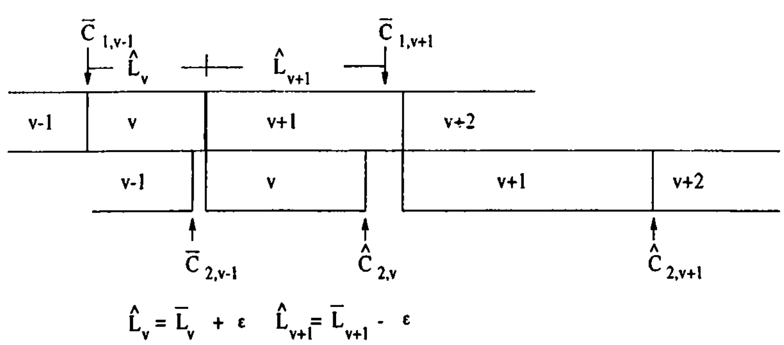

Figure 2.2: Sublot completion, Case 1: $\hat{L} = (\overline{L}_1, \ldots, \overline{L}_v + \epsilon, \overline{L}_{v+1} - \epsilon, \ldots, \overline{L}_s)$

For small $\epsilon > 0$, we also have $\hat{C}_{1,v+1} > \hat{C}_{2,v}$.

$$\begin{aligned}
\hat{C}_{2,v+1} &= \pi \hat{L}_{v+1} + \hat{C}_{1,v+1} \\
&= \pi(\overline{L}_{v+1} - \epsilon) + \overline{C}_{1,v+1} \\
&= \overline{C}_{2,v+1} - \pi\epsilon
\end{aligned}$$

Hence, the condition (2.33) is satisfied. To show $F(\hat{L}) < F(\overline{L})$, define $F_{v,v+1}(L)$ to be the contribution of sublots v and $v+1$ to the objective function and $U_{v,v+1} \equiv \overline{L}_v + \overline{L}_{v+1} = \hat{L}_v + \hat{L}_{v+1}$.

$$\begin{aligned}
F_{v,v+1}(\overline{L}) &= [\overline{C}_{1,v-1} + \overline{L}_v(1+\pi)]\overline{L}_v + [(\overline{C}_{1,v-1} + \overline{L}_v + \overline{L}_{v+1}) + \pi\overline{L}_{v+1}]\overline{L}_{v+1} \\
&= (1+\pi)\overline{L}_v^2 + (U_{v,v+1} + \pi\overline{L}_{v+1})\overline{L}_{v+1} + \overline{C}_{1,v-1}U_{v,v+1} \\
&= (1+\pi)\overline{L}_v^2 + U_{v,v+1}\overline{L}_{v+1} + \pi\overline{L}_{v+1}^2 + \overline{C}_{1,v-1}U_{v,v+1}
\end{aligned}$$

Similarly,

$$\begin{aligned}
F_{v,v+1}(\hat{L}) &= (1+\pi)\hat{L}_v^2 + U_{v,v+1}\hat{L}_{v+1} + \pi\hat{L}_{v+1}^2 + \overline{C}_{1,v-1}U_{v,v+1} \\
&= (1+\pi)(\overline{L}_v + \epsilon)^2 + U_{v,v+1}(\overline{L}_{v+1} - \epsilon) + \pi(\overline{L}_{v+1} - \epsilon)^2 + \overline{C}_{1,v-1}U_{v,v+1}
\end{aligned}$$

then,

$$\begin{aligned} F_{v,v+1}(\overline{L}) - F_{v,v+1}(\hat{L}) &= -(2\pi+1)\epsilon^2 + (2\pi+1)(U_{v,v+1} - 2\overline{L}_v)\epsilon \\ &= -(2\pi+1)\epsilon^2 + (2\pi+1)(\overline{L}_{v+1} - \overline{L}_v)\epsilon \end{aligned}$$

But, we know that $\pi\overline{L}_v < \overline{L}_{v+1}$, thus $\overline{L}_v < \overline{L}_{v+1}$, therefore it is clear that $F_{v,v+1}(\overline{L}) - F_{v,v+1}(\hat{L})$ is positive for some $\epsilon > 0$. Hence, $F(\overline{L}) > F(\hat{L})$, for some $\epsilon > 0$.

Case 2 : $\overline{C}_{1,v+1} \leq \overline{C}_{2,v}$ (See Figure 2.3 and Figure 2.4).

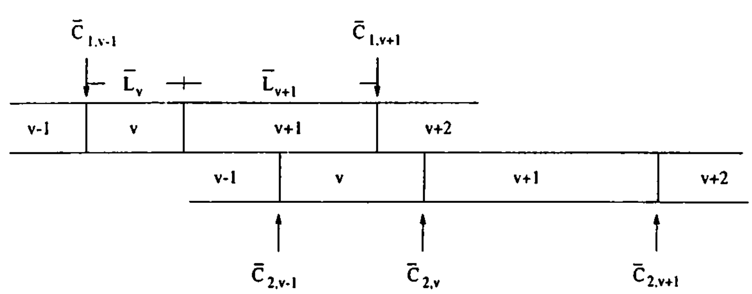

Figure 2.3: Sublot completion, Case 2: $\overline{L} = (\overline{L}_1, \ldots, \overline{L}_v, \overline{L}_{v+1}, \ldots, \overline{L}_s)$

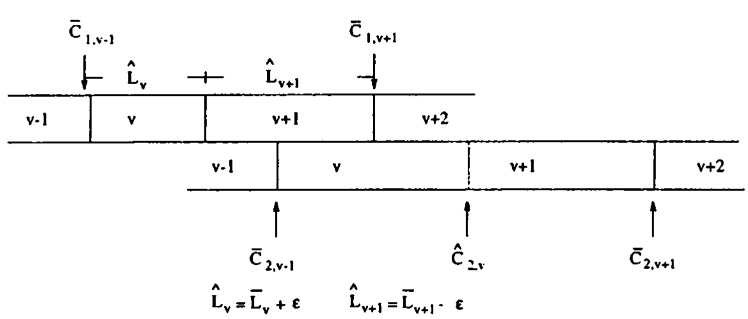

Figure 2.4: Sublot completion, Case 2: $\hat{L} = (\overline{L}_1, \ldots, \overline{L}_v + \epsilon, \overline{L}_{v+1} - \epsilon, \ldots, \overline{L}_s)$

For $\epsilon > 0$ we also have $\hat{C}_{1,v+1} \leq \hat{C}_{2,v}$,

$$\begin{aligned} \hat{C}_{2,v+1} &= \pi\hat{L}_{v+1} + \hat{C}_{2v} \\ &= \pi\hat{L}_{v+1} + \overline{C}_{2,v-1} + \pi\hat{L}_v \\ &= \pi\overline{L}_{v+1} + \overline{C}_{2,v-1} + \pi\overline{L}_v \\ &= \overline{C}_{2,v+1} \end{aligned}$$

since $\overline{L}_v + \overline{L}_{v+1} = \hat{L}_v + \hat{L}_{v+1}$. Hence, (2.33) is satisfied. To show $F(\hat{L}) < F(\overline{L})$,

$$\begin{aligned} F_{v,v+1}(\overline{L}) &= (\overline{C}_{2,v-1} + \pi\overline{L}_v)\overline{L}_v + (\overline{C}_{2,v-1} + \pi U_{v,v+1})\overline{L}_{v+1} \\ &= \pi\overline{L}_v^2 + \pi U_{v,v+1}(U_{v,v+1} - \overline{L}_v) + U_{v,v+1}\overline{C}_{2,v-1} \end{aligned}$$

Similarly,

$$\begin{aligned} F_{v,v+1}(\hat{L}) &= \pi\hat{L}_v^2 + \pi U_{v,v+1}(U_{v,v+1} - \hat{L}_v) + U_{v,v+1}\overline{C}_{2,v-1} \\ &= \pi(\overline{L}_v + \epsilon)^2 + \pi U_{v,v+1}(U_{v,v+1} - \overline{L}_v - \epsilon) + U_{v,v+1}\overline{C}_{2,v-1} \end{aligned}$$

then,

$$\begin{aligned} F_{v,v+1}(\overline{L}) - F_{v,v+1}(\hat{L}) &= -\pi\epsilon^2 + (U_{v,v+1} - 2\overline{L}_v)\pi\epsilon \\ &= -\pi\epsilon^2 + (\overline{L}_{v+1} - \overline{L}_v)\pi\epsilon. \end{aligned}$$

Since $(\overline{L}_{v+1} - \overline{L}_v)\pi$ is positive, $F_{v,v+1}(\overline{L}) - F_{v,v+1}(\hat{L})$ is positive for some $\epsilon > 0$. Thus, $F(\overline{L}) > F(\hat{L})$ for some $\epsilon > 0$. Thus in any optimal schedule, $\pi L_k \geq L_{k+1} \quad k = 1, \ldots s-1$. □

Having observed that $\pi L_k \geq L_{k+1} \quad k = 1, \ldots s-1$ for any optimal schedule, we can write the completion time of each sublot on the second machine as (Figure 2.5),

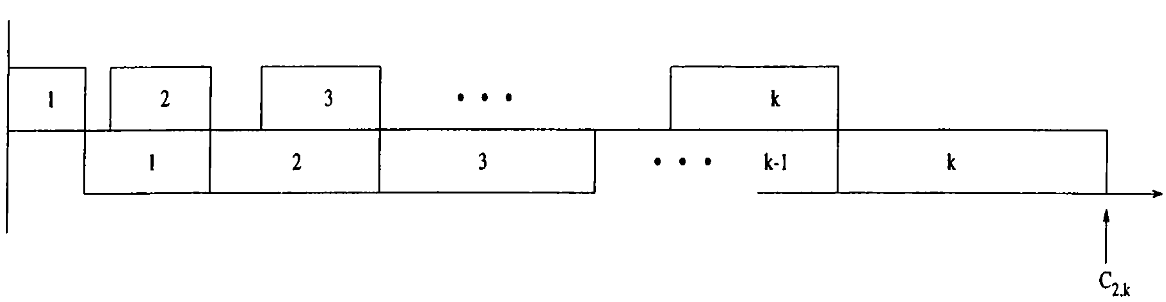

Figure 2.5: Sublot completion, $\pi L_k \geq L_{k+1}$, $k = 1, \ldots, s$

$$C_{2k} = L_1 + \pi \sum_{\ell=1}^{k} L_\ell \quad k = 1, \ldots, s.$$

The mean flow time is;

$$\begin{aligned} F(L) &= \textstyle\sum_{k=1}^{s} C_{2k} L_k \\ &= \textstyle\sum_{k=1}^{s} (L_1 + \pi \sum_{\ell=1}^{k} L_\ell) L_k \\ &= L_1 + \pi \textstyle\sum_{k=1}^{s} \sum_{\ell=1}^{k} L_\ell L_k \end{aligned}$$

Then, an equivalent reformulation of the problem is,

$$\begin{aligned} \min F(L) &= L_1 + \pi \sum_{k=1}^{s} \sum_{\ell=1}^{k} L_\ell L_k && (2.34) \\ \text{subject to} \quad \sum_{k=1}^{s} L_k &= 1, && (2.35) \\ L_{k+1} - \pi L_k &\leq 0, \quad k = 1, \ldots, s-1, && (2.36) \\ L_k &\geq 0, \quad k = 1, \ldots, s. && (2.37) \end{aligned}$$

Result 4 *The following sublot sizes are optimal for (2.34)-(2.37),*

$$\begin{aligned} \overline{L}_1 &= \frac{\frac{\pi^v - 1}{\pi - 1}\pi - (s-v)}{\frac{\pi^{2v}-1}{\pi^2-1}\pi(s-v) + (\frac{\pi^v-1}{\pi-1})^2 \pi}, && (2.38) \\ \overline{L}_k &= \pi^{k-1}\overline{L}_1, \quad k = 1, \ldots, v, && (2.39) \\ \overline{L}_k &= \frac{1 - \overline{L}_1 \sum_{\ell=1}^{v} \pi^{\ell-1}}{(s-v)}, \quad k = v+1, \ldots, s && (2.40) \end{aligned}$$

if $\pi \overline{L}_v \geq \overline{L}_{v+1} \geq \overline{L}_v$ *and* $v < s$.

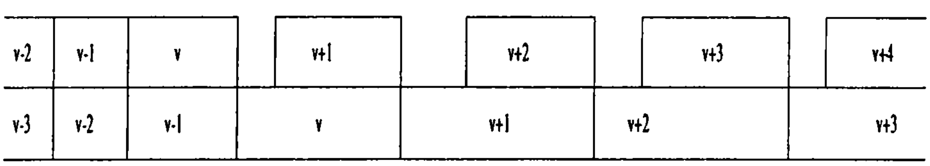

Figure 2.6: Sublot completion, optimal sublots

Proof : The Gantt chart for an instance of the above sublot sizes will be as shown in Figure 2.6. Since the objective function can be shown to be convex, it will be sufficient to show that the above solution is a Karush-Kuhn-Tucker point. The Hessian matrix for the objective function is,

$$H = \begin{bmatrix} 2 & 1 & 1 & 1 & 1 & .. \\ 1 & 2 & 1 & 1 & 1 & .. \\ 1 & 1 & 2 & 1 & 1 & .. \\ 1 & 1 & 1 & 2 & 1 & .. \\ 1 & 1 & 1 & 1 & 2 & .. \\ . & & & & & . \\ . & & & & & . \end{bmatrix}$$

The determinant of each upper left sub-matrix H_r of H is positive since, from (2.32) we have,

$$\det H_r = r + 1, \; r = 1, \ldots, s.$$

Hence, the Hessian matrix is positive definite and the objective function is convex.

Assign, Lagrange Multipliers δ for (2.35), and λ_k for (2.36). As seen in Figure 2.6, only the first $v-1$ of the type (2.36) constraints are binding. So Karush-Kuhn-Tucker conditions for the solution are,

For L_1

$$1 + \pi + \pi L_1 + \delta - \pi\lambda_1 = 0. \tag{2.41}$$

For $L_k \quad k = 2, \ldots, v-1$

$$\pi + \pi L_k + \delta + \lambda_{k-1} - \pi\lambda_k = 0. \tag{2.42}$$

For L_v

$$\pi + \pi L_v + \delta + \lambda_{v-1} = 0. \tag{2.43}$$

For $L_k \quad k = v + 1 \ldots, s$

$$\pi + \pi \mathrm{L}_k + \delta = 0. \tag{2.44}$$

We have the following solution to the system (2.41)-(2.44). Using the values $\overline{L} = (\overline{L}_1, \ldots, \overline{L}_s)$, and noting that $\overline{L}_k = \overline{L}_s \;\; k = v+1, \ldots, s$, we get from (2.44),

$$\delta = -\pi - \pi\overline{L}_s, \tag{2.45}$$

We also get from (2.43) and (2.45),

$$\lambda_{v-1} = \pi(\overline{L}_s - \overline{L}_v), \tag{2.46}$$

which is nonnegative.

From (2.42) we obtain,

$$\lambda_k = \pi\lambda_{k+1} - \delta - \pi - \pi\overline{L}_{k+1} \quad k = 1, \ldots, v-2,$$

$$\lambda_k = \pi\lambda_{k+1} + \pi(\overline{L}_s - \overline{L}_{k+1}) \quad k = 1, \ldots, v-2. \tag{2.47}$$

which together with (2.46), proves the non-negativity of λ_k $k = 1, \ldots, v-2$. We have from (2.46) and (2.47),

$$\lambda_1 = (\frac{\pi^v - \pi}{\pi - 1})\overline{L}_s - (\frac{\pi^{2v} - \pi^2}{\pi^2 - 1})\overline{L}_1. \tag{2.48}$$

On the other hand, (2.41) and (2.45) give,

$$\lambda_1 = \frac{1 + \pi\overline{L}_1 - \pi\overline{L}_s}{\pi}. \tag{2.49}$$

We also need to show that the sublot sizes result in a consistent solution of Lagrange multipliers,

$$\lambda_1 = \frac{1 + \pi\overline{L}_1 - \pi\overline{L}_s}{\pi} = (\frac{\pi^v - \pi}{\pi - 1})\overline{L}_s - (\frac{\pi^{2v} - \pi^2}{\pi^2 - 1})\overline{L}_1$$

$$(\frac{\pi^v - \pi}{\pi - 1})\overline{L}_s - (\frac{\pi^{2v} - \pi^2}{\pi^2 - 1})\overline{L}_1 - \overline{L}_1 + \overline{L}_s = \frac{1}{\pi}$$

$$(\frac{\pi^v - 1}{\pi - 1})\overline{L}_s - (\frac{\pi^{2v} - 1}{\pi^2 - 1})\overline{L}_1 = \frac{1}{\pi}$$

Using (2.40),

$$(\frac{\pi^v - 1}{\pi - 1})\frac{1 - \overline{L}_1(\frac{\pi^v - 1}{\pi - 1})}{(s - v)} - (\frac{\pi^{2v} - 1}{\pi^2 - 1})\overline{L}_1 = \frac{1}{\pi}$$

$$(\frac{\pi^v - 1}{\pi - 1})\frac{1}{(s - v)} - \frac{1}{\pi} = (\frac{\pi^v - 1}{\pi - 1})^2\frac{\overline{L}_1}{(s - v)} + (\frac{\pi^{2v} - 1}{\pi^2 - 1})\overline{L}_1$$

which results in,

$$\overline{L}_1 = \frac{\frac{\pi^v - 1}{\pi - 1}\pi - (s - v)}{\frac{\pi^{2v} - 1}{\pi^2 - 1}\pi(s - v) + (\frac{\pi^v - 1}{\pi - 1})^2\pi} \qquad \square$$

For $v = s$ (i.e. all the sublot sizes are geometric), we have the system of equations (2.41), (2.42) and (2.43). The system has a consistent solution, hence it is enough only to show the non-negativity of the Lagrange multipliers, λ_k, $k = 1, \ldots, s-1$. We have,

$$\lambda_{s-1} = \frac{-\pi^{2s} + 2\pi^{s+1} + 2\pi^s - 2\pi - 1}{(\pi^s - 1)(\pi + 1)\sum_{\ell=0}^{s-1}\pi^\ell}$$

$$\lambda_k = \lambda_{s-1} \sum_{\ell=0}^{s-k-1} \pi^\ell + \sum_{\ell=k+1}^{s-1} (L_s - L_\ell)\pi^{\ell-k}$$

Since $\lambda_k > \lambda_{s-1}$ $k = 1, \ldots s-2$, it is sufficient to check the non-negativity of λ_{s-1}.

Hence, all the sublot sizes are geometric ($v = s$), only if the polynomial in the numerator of λ_{s-1} is positive for a given π, since the denominator is always positive.

Combining these results, the following algorithm solves the problem:

Algorithm I

$v \leftarrow 0$, optimal$\leftarrow$FALSE

If $f(\pi) = -\pi^{2s} + 2\pi^{s+1} + 2\pi^s - 2\pi - 1 > 0$
  optimal $\leftarrow$ TRUE, geometric sublots are optimal

While not optimal
  $v \leftarrow v + 1$
  $\overline{L}_1 \leftarrow [\frac{\pi^v-1}{\pi-1}\pi - (s-v)]/[\frac{\pi^{2v}-1}{\pi^2-1}\pi(s-v) + (\frac{\pi^v-1}{\pi-1})^2\pi]$
  $\overline{L}_k \leftarrow \pi^{k-1}\overline{L}_1,\ k = 2, \ldots, v$
  $\overline{L}_k \leftarrow [1 - \overline{L}_1 \sum_{\ell=1}^{v} \pi^{\ell-1}]/[(s-v)],\ k = v+1, \ldots, s$
  if $\pi\ \overline{L}_v \geq \overline{L}_{v+1} \geq \overline{L}_v$,
    optimal $\leftarrow$ TRUE, $\overline{L}$=$(\overline{L}_1, .., \overline{L}_s)$ is optimal

---

For certain values of s, closed form solutions can be obtained. These solutions can be obtained by determining the intervals of π in which $\pi\overline{L}_v - \overline{L}_{v+1}$ and $\overline{L}_{v+1} - \overline{L}_v$ is positive for $v < s$ and $f(\pi)$ is positive for s. Sample solutions for $s = 2$ and $s = 3$ are given below.

Solution For $s = 2$

$$(L_1^*, L_2^*) = \begin{cases} (\frac{1}{\pi+1}, \frac{\pi}{\pi+1}) & \text{if } 1 \leq \pi \leq 1+\sqrt{2} \\ (\frac{\pi-1}{2\pi}, \frac{\pi+1}{2\pi}) & \text{if } 1+\sqrt{2} \leq \pi \end{cases}$$

In the first interval of π, the sublot sizes are geometric ($v = s = 2$), while in the second $v = 1$.

Solution For $s = 3$

$$(L_1^*, L_2^*, L_3^*) = \begin{cases} (\frac{1}{\pi^2+\pi+1}, \frac{\pi}{\pi^2+\pi+1}, \frac{\pi^2}{\pi^2+\pi+1}) & \text{if } 1 \leq \pi \leq (1+\sqrt{5})/2 \\ (\frac{1}{2}\frac{\pi^2+\pi-1}{\pi^3+\pi^2+\pi}, \frac{1}{2}\frac{\pi^3+\pi^2-\pi}{\pi^3+\pi^2+\pi}, \frac{1}{2}\frac{\pi^3+2\pi+1}{\pi^3+\pi^2+\pi}) & \text{if } (1+\sqrt{5})/2 \leq \pi \leq (3+\sqrt{13})/2 \\ (\frac{\pi-2}{3\pi}, \frac{\pi+1}{3\pi}, \frac{\pi+1}{3\pi}) & \text{if } (3+\sqrt{13})/2 \leq \pi \end{cases}$$

In the first interval of π, the sublot sizes are geometric ($v = s = 3$), in the second $v = 2$, and in the last interval $v = 1$.

VARIABLE SUBLOTS

Although the consistent sublots are optimal in the job completion case, they are not necessarily optimal for the sublot completion case. The following example shows that when the objective is the minimization of sum of sublot completion times, consistent sublots do not result in global optimality. Note that in this case there is a second set of transfers from the second machine.

Consider the following example : 60 units will be processed on a two-stage flow shop and $p_1 = 1$ and $p_2 = 3$. There are two sublots available. As shown in the sample solution above and since $\pi = 3$ the optimal consistent sublot sizes are 20 and 40. These sublot sizes result in a mean flow time of $\frac{1}{60}(20 \times 80 + 40 \times 200) = 160$ (Figure 2.7).

But, we can achieve mean flow time of $\frac{1}{60}(30 \times 105 + 30 \times 195) = 150$ by using sublot sizes (15, 45) on the first machine and (30, 30) on the second machine (Figure 2.7) .

For the optimal variable sublot sizes in a two-stage flow shop, we propose

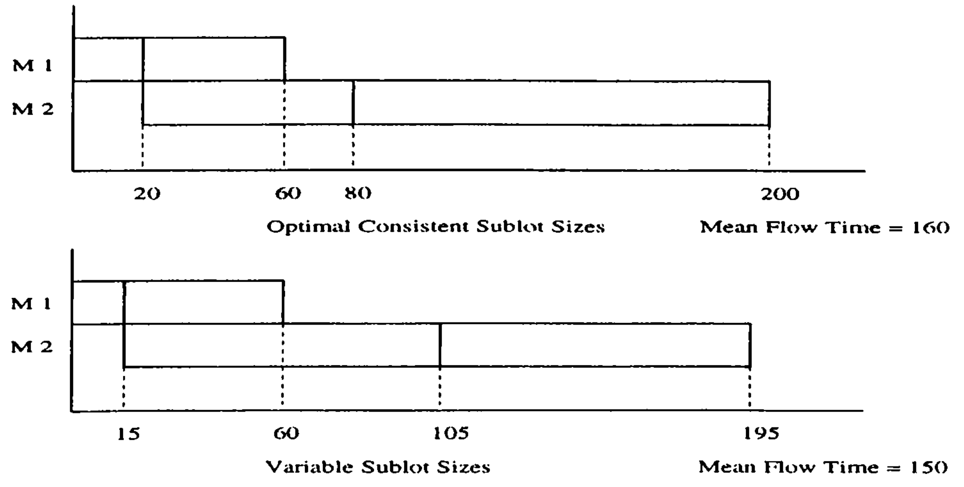

Figure 2.7: Sublot completion, non-optimality of consistent sublots

the following conjectures, without proofs.

Conjecture 1 *Mean flow time is minimized by equal sublots on each stage if* $p_1 \geq p_2$.

Conjecture 2 *Mean flow time is minimized by geometric sublots on first stage, and equal sublots on second stage if* $p_1 < p_2$.

These conjectures depend on the continuous production on dominant machines. For $p_1 \geq p_2$, the first machine is dominant, and determines the sublot sizes. For $p_1 < p_2$, the dominant machine is the second one and geometric sublots on the first machine provide the minimum idle time for second machine which allows continuous production. Thus, operations on the second machine start as early as possible and since geometric sublots on the first machine provide required input, equal sublots are obvious on the second machine. Note that, the Conjecture 2 in addition gives the alternate optimal solution to the minimum makespan problem.

EQUAL SUBLOTS

The optimal sublot sizes are derived for the consistent case and conjectured for the variable case in previous sections. Recall that, for $\pi \leq 1$, equal sublots are optimal. For the case $\pi > 1$, mean flow time with equal sublot sizes is (See Figure 2.8),

$$\begin{aligned} F^E(L) &= (\tfrac{1}{s} + \pi\tfrac{1}{s})\tfrac{1}{s} + (\tfrac{1}{s} + \pi\tfrac{2}{s})\tfrac{1}{s} + \ldots + (\tfrac{1}{s} + \pi\tfrac{s}{s})\tfrac{1}{s} \\ &= \tfrac{1}{s} + \pi\tfrac{1}{s^2}\textstyle\sum_{k=1}^{s} k = \tfrac{1}{s} + \pi\tfrac{(s+1)}{2s}. \end{aligned}$$

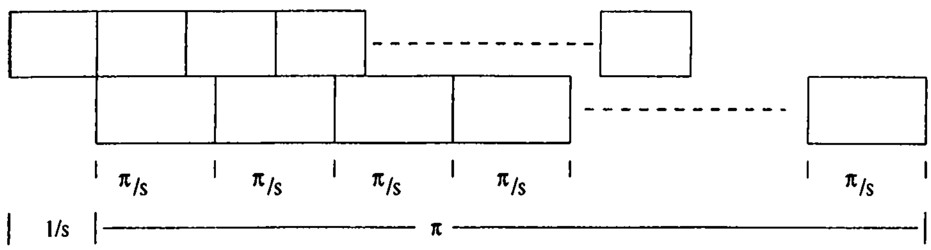

Figure 2.8: Sublot completion, equal sublots

Since it is not possible for general s to derive explicit expression for the optimal mean flow time that can be achieved by the consistent sublots, $F^C(L)$, we shall use a lower bound for its value. We know (from Result 3) that,

$$\pi L_k \geq L_{k+1}, \quad k = 1, \ldots, s-1,$$

is a necessary condition for optimality.

Consider the following linear program:

$$z = \min L_1$$

$$\begin{aligned} \text{subject to} \quad \pi L_k &\geq L_{k+1}, \; k = 1, \ldots, s-1, \\ \sum_{k=1}^{s} L_k &= 1, \\ L_k &\geq 0, \; k = 1, \ldots, s-1. \end{aligned}$$

It is not difficult to show that $z = \frac{\pi-1}{\pi^s-1}$. Thus, the smallest possible size of the first sublot on M_1 is z. Since $p_1 = 1$, z is the earliest time M_2 can start processing. Once M_2 starts processing, it will continue uninterrupted because $\pi > 1$. Thus a lower bound for the optimal flow time, $F^c(L)$, is given by the minimal value of the following quadratic program,

$$F^{LB}(L) = \{\min[z+\pi L_1]L_1+[z+\pi(L_1+L_2)]L_2+\ldots+[z+\pi(L_1+L_2+\ldots+L_s)]L_s\}$$

$$\text{subject to} \quad \sum_{k=1}^{s} L_k = 1,$$
$$L_k \geq 0, \quad k = 1, \ldots, s.$$

which has the solution $L_k = 1/s$ and $F^{LB}(L) = \frac{(\pi-1)}{(\pi^s-1)} + \pi\frac{(s+1)}{2s}$. (See Figure 2.9)

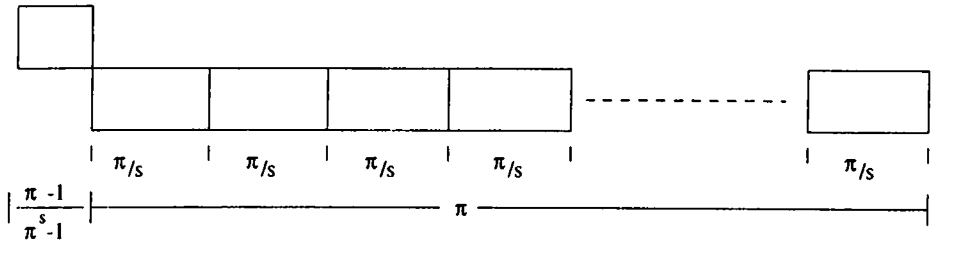

Figure 2.9: Sublot completion, lower bound on consistent sublot sizes

We have $F^{LB}(L) \leq F^C(L)$. Thus, $F^E(L)/F^C(L) \leq F^E(L)/F^{LB}(L)$ where,

$$F^E(L)/F^{LB}(L) = \frac{\frac{1}{s} + \pi\frac{(s+1)}{2s}}{\frac{(\pi-1)}{(\pi^s-1)} + \pi\frac{(s+1)}{2s}}.$$

Result 5 $F^E(L)/F^C(L) \leq F^E(L)/F^{LB}(L) < 1.14$.

Suppose s can take any real value, then, $f(\pi, s) = F^E(L)/F^{LB}(L)$ is a continuous function of s and π for $s \geq 2$ and $\pi > 1$. Then, we set the partial derivatives with respect to s and π equal to zero. These two non-linear equations are solved numerically by Maple V ©, giving a single solution $(\pi^*, s^*) = (1.938, 4.267)$. The solution $(\pi^*. s^*)$ gives $f(\pi^*, s^*) = 1.14$. This single solution is a maximum point, since

$$[\frac{\partial^2 f(\pi^*, s^*)}{\partial\pi\partial s}]^2 - \frac{\partial^2 f(\pi^*, s^*)}{\partial\pi^2}\frac{\partial^2 f(\pi^*, s^*)}{\partial s^2} = -0.00184.$$

It is obvious that, for discrete values of s the function's maximum is less then the one we have found. Consider the solution $(\hat{\pi}, \hat{s}) = (1.992, 4)$. These values result in the ratio 1.139, which is very close the ratio found using (π^*, s^*).

The construction of the lower bound for the consistent sublot sizes gives the optimal variable sublot sizes that we have conjectured, that is geometric sublots on the first machine, equal sublots on the second machine. Since $F^*(L)$ is the mean flow time achievable by variable sublots, we claim the above result holds for the variable sublots ($F^E(L)/F^*(L) < 1.14$).

Minimizing Mean Flow Time under Item Completion Time Model

In this case, an item is assumed to be completed as soon as it completes processing in the last machine. When continuous sublot sizes are allowed, this is equivalent to assuming infinite number of transfers in the last stage. In the case of two-machine flow shop with consistent sublot sizes, the objective function is

$$\min \sum_{k=1}^{s} \left[C_{2k} - (p_2/2)L_k\right] L_k$$

subject to Constraints (2.22)–(2.27).

Again we have two cases to consider: (i) $\pi \leq 1$, and (ii) $\pi > 1$, where $\pi \equiv p_2/p_1$. Çetinkaya & Gupta [8] have shown that, if $\pi \leq 1$, then equal size sublots are optimal, otherwise it is optimal to use the geometric sublot sizes as given in equations (2.28) and (2.29).

EQUAL SUBLOTS

Note that, equal sublots are also optimal, when $\pi \leq 1$. Therefore, we again consider the case $\pi > 1$, in which geometric sublot sizes are optimal. Equal sublots give the following mean flow time (See Figure 2.10 and Figure 2.11)

$$F^E(L) = 1/s + \pi/2,$$

since all the items can be assumed to be delivered at time $1/s + \pi/2$. For the optimal sublot sizes we have a similar form:

$$F^*(L) = \frac{(\pi - 1)}{(\pi^s - 1)} + \pi/2.$$

Result 6 $F^E(L)/F^*(L) < 1.18$.

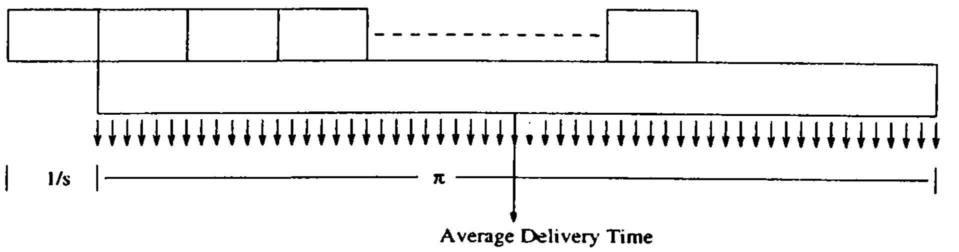

Figure 2.10: Item completion, equal sublot sizes

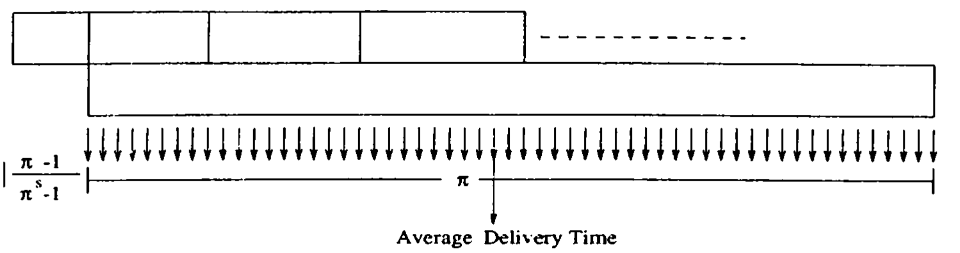

Figure 2.11: Item completion, optimal sublot sizes

This result is obtained by a similar approach to the one used in Result 5. For $s = 4$ and $\pi = 2.021$, the ratio turns out to be 1.172.

In this section we analyzed lot streaming of a single job in a two-stage flow shop. Makespan minimization problem can be viewed as a mean flow minimization problem under the job completion time model. Where applicable, consistent and variable sublots are separately treated. The implications of equal sublots, which are widely used in practice are also presented. Table 2.1 summarizes the results of minimization of mean flow time in a two-stage flow shop.

Except in Sublot Completion Time Model with $p_1 < p_2$, consistent sublots are optimal in other cases even if variable sublot sizes are allowed. As seen from the last column, equal sublots are quite effective. Thus, the practical use of equal sublots may be justified.

There may be other streaming policies applicable to the two-machine flow shop. Some instances may allow infinite number of transfers (unit transfers) between machine 1 and machine 2. In this case, equal sized deliveries are

Table 2.1: Two-Machine Mean Flow Time Problems

		Sublot Sizes		Bound
		Consistent	Variable	Equal/Opt
Job Completion	$p_1 \geq p_2$	Geometric	Geometric	1.09
	$p_1 < p_2$	Geometric	Geometric	1.09
Sublot Completion	$p_1 \geq p_2$	Equal	Equal	†
	$p_1 < p_2$	Algorithm I	Machine I: Geometric* Machine II :Equal*	1.14
Item Completion	$p_1 \geq p_2$	Equal	†	†
	$p_1 < p_2$	Geometric	†	1.18

* conjectured † not applicable

obvious. Some other models may allow different number of sublots at each stage. While specific instances should be studied for analytical results for this problem, the general model of Benli [5] presented in section 2.1.1 provides a mixed integer linear (or quadratic) programming formulation.

2.1.3 Three or More Machines

When there are three or more machines and the consistent sublots are used, linear programming and quadratic programming formulations are available for minimum makespan and minimum mean flow time problems.

Potts & Baker [20] observed that the flow shop problem with processing times $p_1, \ldots, p_i, \ldots, p_m$, and the *inverse* problem with processing times $p_m, \ldots, p_{m-i+1}, \ldots, p_1$ are equivalent.

Baker [1] studied the three-machine problem with two sublots and obtained results similar to that of the two-machine problem. Glass et. al. [12] used the network representation of the lot streaming problem to provide solutions for the three machine problem with s sublots. A vertex (i, k) is defined for each machine i and for each sublot k, with weight $p_i L_k$. Directed edges from vertex (i, k) to vertex $(i+1, k)$ for $i = 1, \ldots, m-1$ and $k = 1, \ldots, s$ ensure that sublot

k can start processing on machine $i+1$ only after it is completed on machine i. Directed edges from (i,k) to $(i,k+1)$ for $i=1,\dots,m$ and $k=1,\dots,s-1$ ensure that machine i can start processing sublot $k+1$, only after it completes the processing of sublot k. The *length* of a path is defined as the total weight of vertices that are on it. The longest path from vertex $(1,1)$ to vertex (m,s), referred to as *critical path*, gives the makespan. A 3-machine 4-sublot problem is depicted in Figure 2.12.

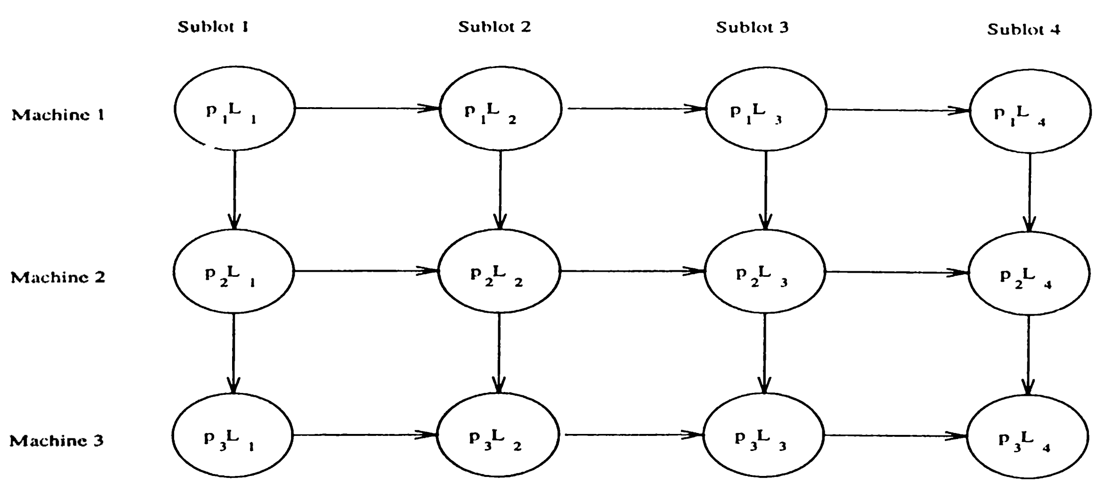

Figure 2.12: Network representation of a lot streaming problem

Glass et. al. [12] showed that, in an optimal solution, all sublots are positive. This intuitive result states that all the possible transfers will be utilized to accelerate the production. Using network representation of the problem, Glass et. al. [12] derived the optimal consistent sublot sizes for the three-machine minimum makespan problem.

Result 7 *In a three-stage flow shop, if $p_2^2 \le p_1p_3$ optimal sublot sizes are*

$$L_1 = \begin{cases} (q-1)/(q^s-1), & \text{if } p_1 \neq p_3, \\ 1/s, & \text{if } p_1 = p_3, \end{cases}$$

$$L_k = q^{v-k}, \quad k = 2,\dots,s,$$

where, $q = (p_2+p_3)/(p_1+p_2)$.

Result 8 *In a three-stage flow shop, if $p_2^2 > p_1 p_3$ optimal sublot sizes are*

$$
\begin{aligned}
L_k &= q_1 L_v, \quad k = 1, \ldots, v-1, \\
L_v &= \begin{cases} 1/[(q_1^v - 1)/(q_1 - 1) + (q_3^{s-v+1} - 1)/(q_3 - 1) - 1], & \text{if } p_1 \neq p_2, p_2 \neq p_3, \\ 1/[v - 1 + (v_3^{s-v+1} - 1)/(q_3 - 1)], & \text{if } p_1 = p_2, p_2 \neq p_3, \\ 1/[(q_1^v - 1)/(q_1 - 1) + s - v], & \text{if } p_1 \neq p_2, p_2 = p_3 \end{cases} \\
L_k &= q_3^{k-v} L_k, \quad k = v+1, \ldots, s,
\end{aligned}
$$

where v can be easily found by bi-section search in $\{1, \ldots, s\}$ and $q_1 = p_1/p_2$, $q_3 = p_3/p_2$.

When the sublot sizes are not restricted to be consistent, the three-machine problem can be solved by a procedure proposed by Trietsch & Baker [30].

When intermittent idling is not allowed, the two-machine solution can be applied independently to the consecutive machines to find the variable sublot sizes which minimize makespan.

When the objective is minimization of the mean flow times, note that the results presented in Section 2.1.2 for the case $p_1 \geq p_2$ is applicable to m-machine problem for the case $p_1 \geq \max_{2 \leq i \leq m}\{p_i\}$. Thus, equal sized sublots are optimal for minimizing mean flow time under sublot and item completion time models, when the processing time on the first machine is greater than processing times any of the other machines.

The two-sublot problem received a greater attention, since marginal returns diminishes as the number of sublots increases. Baker & Jia [3] reported that two or three sublots are sufficient to obtain most of the benefit that can be achieved by lot streaming. Furthermore, two-sublot solutions can be used in developing heuristic methods in s-sublot problems.

Baker & Pyke [4] and Williams & Tüfekçi [36] studied the two-sublot makespan minimization problem. They derived algorithms of complexity $\mathcal{O}(m^2)$ to calculate the optimal sizes of the consistent sublots and used it in heuristics to compute the sizes of multiple sublots.

Topaloğlu et. al. [28] and Çetinkaya & Gupta [8] proposed $\mathcal{O}(m^2)$ time algorithms to find sublot sizes that minimize mean flow time under sublot and item completion time models.

2.2 Open Shop Models

In open shop problems, since one is able to choose any routing for the jobs, it is possible to obtain shorter makespan than the flow shop problems. However, this flexibility also adds complexity to both formulation and solution of open shop problems. Therefore, the current research is limited to minimum makespan problems. Before analyzing the lot streaming problem, the basic properties and results in open shops will be summarized.

An open shop schedule must satisfy the following two sets of constraints,

- No two jobs can be processed simultaneously on a machine. That is, for each M_i and for each pair of jobs (J_j, J_k),

$$\text{either } C_{ij} \geq P_{ij} + C_{ik} \text{ or } C_{ik} \geq P_{ik} + C_{ij}. \tag{2.50}$$

- No two machines can process a job simultaneously. That is, for each J_j and for each pair of machines (M_i, M_ℓ),

$$\text{either } C_{ij} \geq P_{ij} + C_{\ell j} \text{ or } C_{\ell j} \geq P_{\ell j} + C_{ij} \tag{2.51}$$

Gonzales & Sahni [13] proposed a linear time algorithm, to minimize makespan in a two-machine non-preemptive open shop when there is no lot streaming. We briefly outline the algorithm below,

Denote $a_j = P_{1j}$, $b_j = P_{2j}$

Algorithm II

Step 1: Define $A = \{J_j | a_j \geq b_j\}$, $B = \{J_j | a_j < b_j\}$

Step 2: Choose J_r and J_ℓ to be any two distinct jobs whether in A or B such that

$$a_r \geq \max_{J_j \in A} b_j \quad b_\ell \geq \max_{J_j \in B} a_j$$

and let $A' = A - \{J_\ell, J_r\}$ $B' = B - \{J_\ell, J_r\}$

Step 3: If $\sum_{j=1}^{n} a_j - a_\ell > \sum_{j=1}^{n} b_j - b_r$,

Construct the schedule (J_ℓ, B', A', J_r) on M_1, (J_r, J_ℓ, B', A') on M_2, with job J_r having the routing (M_2, M_1), and other jobs (M_1, M_2)

otherwise,

Construct the schedule (B', A', J_r, J_ℓ) on M_1, (J_ℓ, B', A', J_r) on M_2, with job J_ℓ having the routing (M_2, M_1), and other jobs (M_1, M_2)

Note that the jobs in A' and B' can be ordered arbitrarily.

It can be shown [13] that the algorithm finds a schedule with a makespan,

$$C_{max} = \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j, \max_j(a_j + b_j)\} \tag{2.52}$$

Since this is a lower bound for the length of any schedule, the algorithm is optimal. However, Gonzales & Sahni [13], also have shown that the problem is *NP-Hard* for $m \geq 3$.

It has been customary to analyze the scheduling problems from the machines' point of view. Alternatively, one may consider the problems from the viewpoint of jobs. For example, the Gantt charts can be constructed for the jobs rather than the machines. In Figure 2.13, the first Gantt chart represents a 2-machine 3-job schedule, while the second one represents the same schedule from the jobs point of view. This *"duality"* is useful in open shop problems.

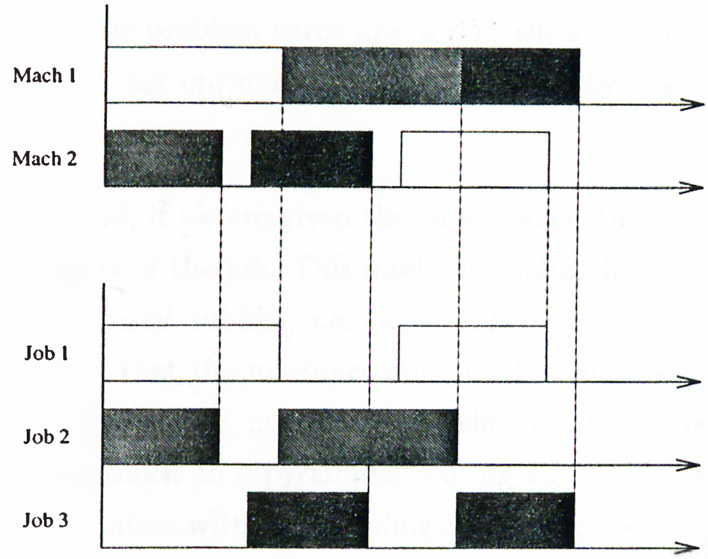

Figure 2.13: Gantt charts for machines and jobs

Since there is no machine order in open shops, the two types of representation are equivalent in studying a minimum makespan problem. Therefore, if we consider jobs as machines and machines as jobs, the schedules (makespans) will not be affected. Hence an m-machine n-job open shop minimum makespan problem (in which processing time of job j on machine i is P_{ij}) is equivalent to an n-machine m-job open shop minimum makespan problem (in which processing time of job i on machine j is P_{ij}).

In the lot streaming problem, there are two cases to consider. In the first case, all the sublots of the single job may be restricted to follow the same routing, which will be called *single routing* models. In this case, the routing for the job and sizes of the sublots should be optimized. However, the open shop may have further flexibility to allow for different routings for each sublot of the single job, i.e. a *multiple routing* model. In this case, we expect to have shorter makespans by optimizing the routing and size of each sublot.

2.2.1 Single Routing Model

Assume that the sublots are consistent. There are two decisions to be made: the routing of the job and the sizes of the sublots. Clearly, if we are given the routing of the job, the problem turns into a flow shop lot streaming problem, for which we can obtain optimal solutions efficiently by linear programming formulations [1].

On the other hand, if we are given the sublot sizes, the problem is only to determine the routing of the job. This model is studied by Steiner & Truscott [24] with the equal sized sublots, i.e. $L_k = 1/s$, $k = 1, \ldots, s$. With the additional restriction that the machines must work continuously ("*continuous work*"), they have shown that in an optimal schedule the job must follow any of the *pyramidal* routings. In a pyramidal routing $R_p = (M_{[1]}, M_{[2]}, \ldots, M_{[m]})$, the job visits the machines with an ascending order of processing times followed by machines with a descending order of processing times, i.e., there is no i such that $p_{[i-1]} > p_{[i]} < p_{[i+1]}$.

Here, we relax the assumption that the sublots must be of equal size and the machines must work continuously. However, we will show that the result we will obtain, will also imply the above mentioned result.

Now, consider the m-machine open shop lot streaming problem. Suppose that sublots are known *a-priori* and be $\overline{L} = (\overline{L}_1, \overline{L}_2, \ldots, \overline{L}_s)$. Hence the problem is a classical open shop problem with m machines and s jobs, with processing times,

$$p_{ik} = p_i \overline{L}_k, \quad i = 1, \ldots, m, \quad k = 1, \ldots, s.$$

But in this specific problem, we also have job $(k+1)$ follows job k. Therefore, the dual s-machine m- job open shop problem is in fact a flow shop problem with processing times,

$$p_{ik} = p_k \overline{L}_i, \quad i = 1, \ldots, s, \quad k = 1, \ldots, m. \tag{2.53}$$

Observing this relation, we can now use the basic results of the flow shop problem.

Note that, when there are two sublots, the corresponding flow shop problem is a 2-machine one. There are two cases to consider, $\overline{L}_1 \geq \overline{L}_2$ and $\overline{L}_2 > \overline{L}_1$. When $\overline{L}_1 \geq \overline{L}_2$, in the corresponding flow shop, the processing time on the first machine is always greater than the processing time on the second machine for each job, (2.53). An optimal solution to this problem is LPT sequence for the processing times on machine 2 (see Section 3.1.1). Thus, the routing in the original open shop, which corresponds to the LPT sequence in the corresponding flow shop, is the routing in which the job visits the machines with a descending order of processing times, i.e. the routing $R_d = (M_{[1]}, M_{[2]}, \ldots, M_{[m]})$ is such that $p_{[i]} \geq p_{[i+1]}$ for $i = 1, \ldots, m-1$. Similar arguments are valid for the case $\overline{L}_2 > \overline{L}_1$, in which the job visits the machines with an ascending order of processing times, i.e. the routing $R_a = (M_{[1]}, M_{[2]}, \ldots, M_{[m]})$ is such that $p_{[i]} \leq p_{[i+1]}$ for $i = 1, \ldots, m-1$. Moreover, because of the reversibility of the flow shop lot streaming problem, the two routings give the same makespan. Thus, it is enough to consider only one of these routings. Once the routing is known, the problem is a single job two-sublot flow shop lot streaming problem, which can be solved by an LP formulation or by the algorithms due to Baker & Pyke [4] and Williams & Tüfekçi [36].

When there are more than two sublots ($s > 2$), we observe the following characteristic of the corresponding s-machine flow shop.

$$p_{ik} > p_{i\ell} \Rightarrow p_{hk} > p_{h\ell},$$

$$\text{since} \quad p_k\overline{L}_i > p_\ell\overline{L}_i \Rightarrow p_k\overline{L}_h > p_\ell\overline{L}_h, \text{ and}$$

$$p_{ik} > p_{hk} \Rightarrow p_{i\ell} > p_{h\ell},$$

$$\text{since} \quad p_k\overline{L}_i > p_k\overline{L}_h \Rightarrow p_\ell\overline{L}_i > p_\ell\overline{L}_h \;.$$

$$\text{for } i, h \in \{1, \ldots, s\} \; k, \ell \in \{1, \ldots, m\}.$$

These characteristics are nothing but the properties of an ordered flow shop. Smith et. al. [23] have shown that the best permutation schedule for this problem is one of the *pyramidal* schedules, i.e. the sequence on any machine $S_p = (J_{[1]}, J_{[2]}, \ldots, J_{[m]})$ is such that there is no k, $1 \leq k \leq m$ such that $p_{[k-1]} > p_{[k]} < p_{[k+1]}$. An immediate result of pyramidal schedules in the corresponding flow shop is the pyramidal routings for the original open shop. Hence, we need to consider one of the 2^{m-1} pyramidal routings.

When the sublots are of equal size, it is easy to see that all the pyramidal routings result in the same makespan. Moreover, it is always possible to ensure continuous work on each machine without increasing the makespan. Hence, this is an alternative proof for the result given in [24].

2.2.2 Multiple Routing Model

In this case, each sublot of the job may have a different routing resulting in shorter makespans. This problem is studied by Glass et. al. [12] and the following results are derived.

When the number of sublots is more than the number of machines, i.e. $s \geq m$, optimal sublots are consistent and,

$$L_k = \begin{cases} 1/m, & \text{for } k = 1, \dots, m, \\ 0, & \text{for } k = m+1, \dots, s, \end{cases}$$

with sublot k having the routing $(M_k, \dots, M_m, M_1, \dots, M_{k-1})$ and achieving a makespan $C_{max} = \max\{p_1, \dots, p_m\}$. Note that, in each of the m equal length intervals in the interval $(0, C_{max})$, each machine processes exactly one of the m sublots and hence there is no overlapping.

When there are two sublots and m machines, optimal sublots are consistent and $L_1 = L_2 = 1/2$. The routings of two sublots are found by applying Gonzales & Sahni's algorithm [13] to the corresponding 2–machine m–job problem, generating a makespan, $C_{max} = \max\{1/2 \sum_{i=1}^{m} p_i, \max\{p_1, \dots, p_m\}\}$

2.3 Job Shop Models

In a single job problem, job shop problem is different than flow shop problem, only when the job requires the same machine at different stages of its production.

Consider the two-machine job shop, in which the job requires machine 1 at the first and third stages and machine 2 at the second stage and sublot sizes are consistent. Glass et. al. [12] showed that this problem can easily be solved using the 3-machine flow shop results, discussed in Section 2.1.3. Let the processing time of the job be P_1 at the first stage on machine 1, P_2 at the second stage on machine 2, and P_3 at the third stage on machine 1. Relaxing the assumption that the job requires same machine at stages 1 and 3 and solving the problem as a 3-machine flow shop problem (using Results 7 and 8), we obtain a schedule. Let C^*_{max} be the makespan of this schedule. Note that, the makespan does not increase, if we ensure no intermittent idling on first and third stages. Now if $C^*_{max} \geq P_1 + P_3$, we are done, there is no overlapping of operations at first and third stages. If $C^*_{max} < P_1 + P_3$, increase the start time of all the sublots on third stage by length $P_1 + P_3 - C_{max}$. The resulting schedule has length $\max\{C^*_{max}, P_1 + P_3\}$ and therefore optimal.

Chapter 3

Multiple Job Models

Lot streaming problems are harder when the number of jobs is more than one. This is due to the fact that the sublot sizing, routing and sequencing decisions must be made simultaneously. Therefore, some researchers assumed that the sublot sizes are known *a-priori* (i.e., equal or unit sized transfers) and tried to implement rules for sequencing sublots. It is obvious that, even these assumptions will not help to derive exact and efficient solutions, because of the already $NP - Hard$ nature of the scheduling problems without lot streaming. In this chapter, we will discuss 2-machine problems with the objective of minimizing makespan.

In addition to the variety of the problems discussed for the single job, we have to also consider the preemptive and non-preemptive models. While we do not allow for interruption of individual sublots, makespan may improve when one processes sublot(s) of a job in between any two consecutive sublots of some other job on a machine.

In case of multi jobs, we must also take into account the setup times. The setup of a job may be *attached* to the first sublot of the job, i.e. setup may require the presence of a physical unit. The *detached* setups can be made, whenever the machine is idle. Note that, if the inventory costs are not extremely high, attached setups can be converted to detached setups, by simply holding one physical unit of each job, at each machine.

3.1 Flow Shop Models

3.1.1 Non-Preemptive Models

The fundamental result in two-machine flow shop is the Johnson's algorithm to minimize makespan [15]. Denote $a_j = P_{1j}$, $b_j = P_{2j}$. Let $A = \{J_j | a_j \leq b_j\}$ and $B = \{J_j | a_j > b_j\}$. An optimal sequence of jobs, which is the same on both machines, is the Shortest Processing Time (SPT) ordering of jobs in A, according to a_j, followed by a Longest Processing Time (LPT) ordering of jobs in B, according to b_j.

Mitten [18] extended the Johnson's algorithm to allow for overlapping of the operations at both machines. As presented in [22], define ℓ_j to be the *start lag* of job j, i.e., job j may start processing on M_2 ℓ_j time units after it is started on M_1. Alternatively, ℓ'_j is the *stop lag* of job j, i.e., job j cannot be completed on M_2 before ℓ'_j time units elapsed after it is completed on M_1. It is shown that, Johnson's algorithm can be applied to this time lag problem, with modified processing times ℓ_j on M_1 and ℓ'_j on M_2 for each job j.

Vickson & Alfredsson [34] studied the lot streaming problem with unit sized sublots. Identifying each unit as a distinct job, they observed that the shop can be also scheduled using Johnson's algorithm. They have shown that there exists an optimal schedule where there is no preemption and the optimal job sequence does not change if each unit is transferred in $t > 0$ time units from M_1 to M_2. They have also extended these results to three machines. The non-preemptive schedules may not be optimal, when the objective is minimizing sum of sublot completion times, even in a two-machine flow shop.

In the existence of detached setups, Çetinkaya & Kayalıgil [9] derived an algorithm similar to Johnson's to find optimal sequence of jobs, which have unit sized sublots.

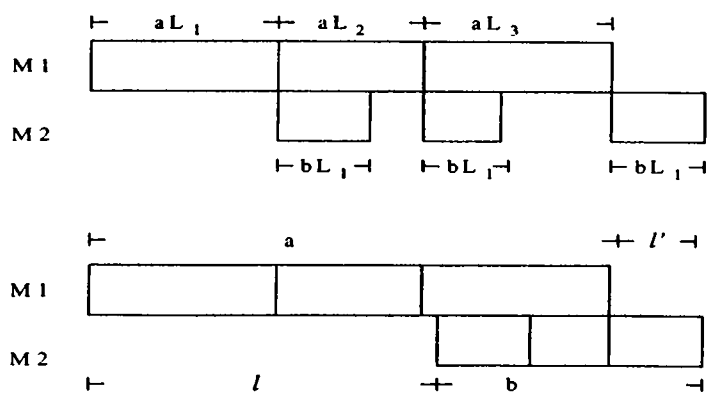

Figure 3.1: Time lags for lot streaming

However, a time lag model for the lot streaming problem is more appropriate and insightful. First, consider the simplest case, in which there are no setups. Let the sublot sizes be given for each job j, $L_j = L_{j1}, L_{j2}, \ldots, L_{js_j}$. Consider each job separately. The sublots can be shifted to the left on M_1 and to the right on M_2, resulting in a no intermittent idling case, without increasing the flow time of the job (See Figure 3.1). Then the start lag ℓ is the difference between the starting times of the job on M_1 and M_2. Alternatively, stop lag ℓ' is the difference between completion times of the job on M_1 and M_2. The sequence of jobs can be easily found by applying the Johnson's algorithm using the modified processing times ℓ_j on M_1 and ℓ'_j on M_2 for each job j.

Baker [2] used the time lag model to sequence the two-machine flow shop with equal sized sublots in which jobs have detached or attached setups. Vickson [33] and Çetinkaya [7] independently showed that, sublot sizing and sequencing decisions can be made separately. Çetinkaya studied the problem with detached setup times and found optimal sublot sizes similar to the geometric sublots described in Section 2.1.2 for the makespan problem. Vickson considered both the detached and attached setups and found similar results. The two authors also considered the case when the number of units in the sublots are restricted to integers. To see that the optimal sublot sizing of a job is independent of other jobs, assume the contrary, i.e. there are jobs that are not streamed according to the optimal rule if they were to be streamed separately. Applying the optimal rule to each job will obviously decrease the completion time of each job and thus makespan.

If there is no setup, the optimal rule for each job is the geometric rule given by the equations 2.28 and 2.29. Then, it is easy to see that sublots $(2, \ldots, s_j)$ on M_1 will overlap with the sublots $(1, \ldots, s_j - 1)$ on M_2. Therefore, time lags will be $\ell = a_j L_{j1}$ and $\ell' = b_j L_{js_j}$. Modifying the processing times with these lags and applying Johnson's algorithm will give the optimal schedule.

3.1.2 Preemptive Models

Potts & Baker [20] showed that even in a simple problem with two machines and two sublots, non-preemptive schedules may not be optimal. Moreover, even with the equal sized sublots, preemptive schedules may be optimal in a three-stage flow shop [34]. Therefore, especially when the setups are negligible, we have to consider also the preemptive schedules.

To our knowledge, there is no study of analytical models in the literature on streaming multi jobs in a flow shop. However, Dauzere-Peres & Laserre [10] give an iterative procedure to solve the preemptive open shop, job shop and flow shop problems. The procedure starts with a sequence of sublots on each machine. Given the sequences, optimal sublot sizes are computed. The optimal sublot sizes are then input to a classical scheduling problem in which each sublot are assumed to be distinct jobs. The iterative procedure stops when there are no more improvements.

3.2 Open Shop Models

In this section, we study the 2–machine open shop problem. In the first part, we discuss the non-preemptive case where each sublot of a job has the same routing, i.e. "single routing". In the second part, we study the preemptive case where each sublot of a job may have different routings, i.e. "multiple routing". Again, we denote $a_j = P_{1j}$, $b_j = P_{2j}$.

When there are only two machines, Gonzales & Sahni's [13] (see Section 2.2) linear time algorithm finds the optimal schedule with a makespan,

$$C_{max} = \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j, \max_j(a_j + b_j)\}.$$

Clearly, if $\max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\} \geq \max_j a_j + b_j$, the makespan cannot be improved by lot streaming. Hence, lot streaming is efficient only if $\max_j(a_j + b_j) > \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\}$.

For the results of the following sections, we need the following lemma.

Lemma 2 *There can be at most one job v such that*

$$a_v + b_v > \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\}. \tag{3.1}$$

Proof: Suppose that there are two jobs v and ℓ that satisfy,

$$a_v + b_v > \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\}, \tag{3.2}$$

$$a_\ell + b_\ell > \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\}, \tag{3.3}$$

adding both sides,

$$a_v + b_v + a_\ell + b_\ell > 2\max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\}. \tag{3.4}$$

On the other hand,

$$\sum_{j=1}^{n} a_j + \sum_{j=1}^{n} b_j \geq a_v + b_v + a_\ell + b_\ell, \tag{3.5}$$

$$2\max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\} \geq \sum_{j=1}^{n} a_j + \sum_{j=1}^{n} b_j. \tag{3.6}$$

From (3.4),(3.5) and (3.6) we have,

$$\max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\} > \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\}, \tag{3.7}$$

which contradicts with the existence of jobs v and ℓ. □

3.2.1 Non-preemptive Single Routing Model

Consider the case, in which there is a job v that satisfies (3.1), for otherwise lot streaming will not improve makespan. Without loss of generality, assume that $U_v = 1$. If we consider only the job v for streaming, assigning an arbitrary routing (M_1, M_2) or (M_2, M_1) and ignoring the other jobs, the optimal sizes of the s_v sublots are simply the *geometric* sublots given in [20]. If we take the routing as (M_1, M_2) these sizes are,

$$L_{v1} = \frac{1-\pi}{1-\pi^{s_v}}, \tag{3.8}$$

$$L_{vk} = \pi L_{k-1},\ k = 2, \ldots, s_v, \tag{3.9}$$

where $\pi \equiv b_v/a_v$. The optimal completion time of the job v is,

$$C_v = a_v L_{v1} + b_v = a_v \frac{1-\pi}{1-\pi^{s_v}} + b_v \tag{3.10}$$

(See Figure 3.2.a).

The next step in constructing an optimal schedule is sequencing other jobs to the right of job v on machine M_1 and to the left of job v on machine M_2 (See Figure 3.2.b). The resulting schedule is optimal if one of the following conditions hold,

$$a_v L_{v1} \geq \sum_{j=1, j\neq v}^{n} b_j \text{ and } b_v L_{vs_v} \geq \sum_{j=1, j\neq v}^{n} a_j \tag{3.11}$$

which results in a makespan $C_{max} = C_v$ (Figure 3.2.b),

$$a_v L_{v1} \geq \sum_{j=1, j\neq v}^{n} b_j \text{ and } b_v L_{vs_v} < \sum_{j=1, j\neq v}^{n} a_j \tag{3.12}$$

which results in a makespan $C_{max} = \sum_{j=1}^{n} a_j$ (Figure 3.2.c),

$$a_v L_{v1} < \sum_{j=1, j\neq v}^{n} b_j \text{ and } b_v L_{vs_v} \geq \sum_{j=1, j\neq v}^{n} a_j \tag{3.13}$$

which results in a makespan $C_{max} = \sum_{j=1}^{n} b_j$ (Figure 3.2.d).

However, if

$$a_v L_{v1} < \sum_{j=1, j\neq v}^{n} b_j \text{ and } b_v L_{vs_v} < \sum_{j=1, j\neq v}^{n} a_j \tag{3.14}$$

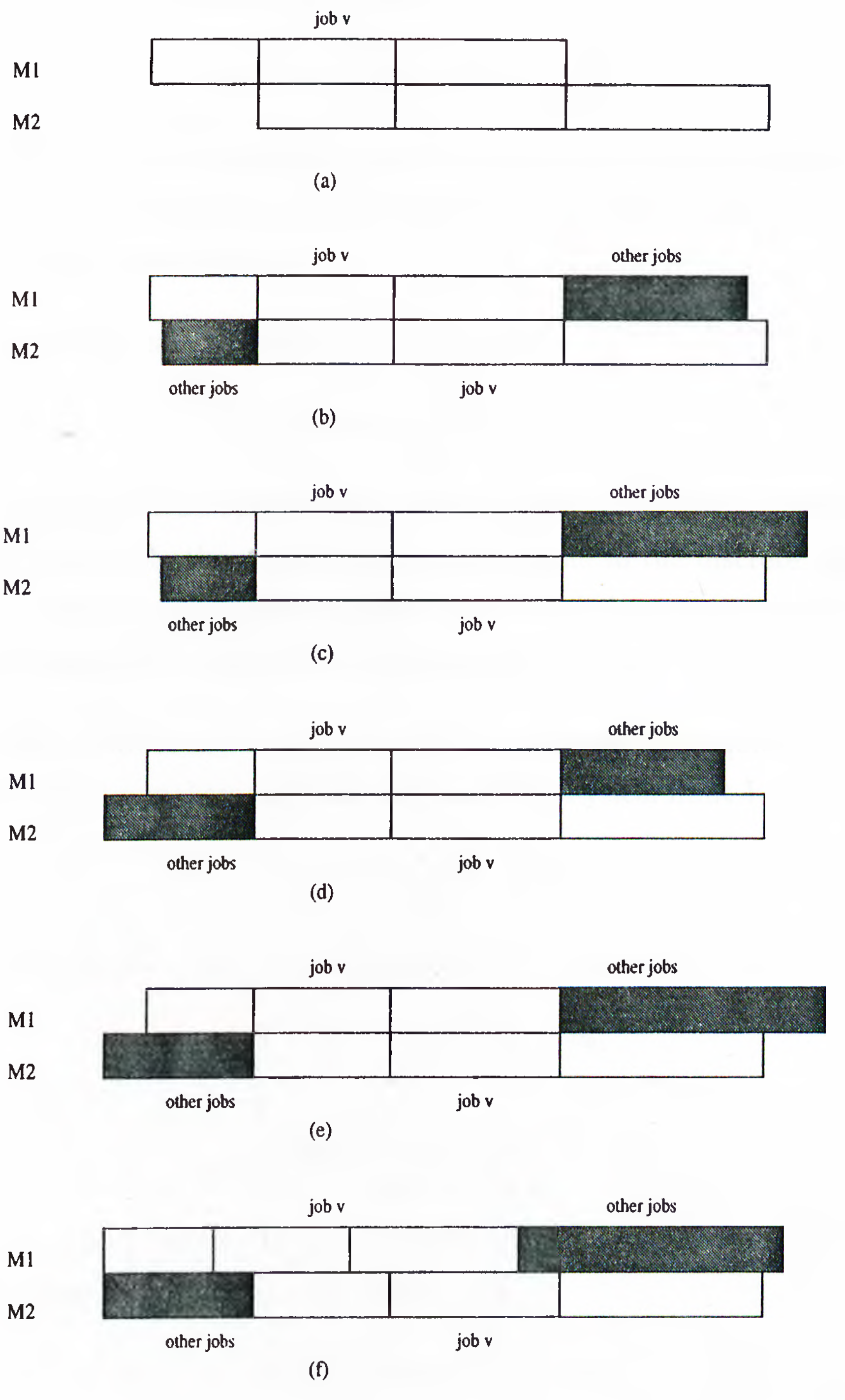

Figure 3.2: Two-machine open shop, constructing the optimal schedule

(Figure 3.2.e), a left shift of all jobs and sublots on machine M_1 will be required to achieve a makespan $C_{max} = \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\}$ (Figure 3.2.f). Note that this left shift is always possible. Since,

$$a_v + b_v > \sum_{j=1}^{n} b_j \Rightarrow a_v > \sum_{j=1, j\neq v}^{n} b_j \quad (3.15)$$

means that the processing time of job v on machine M_1 is longer than the total processing of all other jobs on machine M_2, a left shift on machine M_1 does not create any overlapping.

Since in each case we achieve the makespan,

$$C_{max} = \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j, C_v\} \quad (3.16)$$

this construction is an optimal one. Although this procedure assumes infinite divisibility of a job, the results are also applicable to the discrete sublot case by using Trietsch's [29] iterative algorithm to find the optimal discrete sublot sizes of a single job streamed in a 2–machine flow shop.

For the continuous case, we can also determine the required number of sublots to have a makespan which achieves the physical limit Y, which is,

$$Y = \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\} \quad (3.17)$$

for any two-machine shop. Equating this limit to C_v,

$$Y = C_v = a_v \frac{1-\pi}{1-\pi^{s_v}} + b_v, \quad (3.18)$$

we get

$$s_v = \left\lceil \frac{\log(Y - a_v) - \log(Y - b_v)}{\log b_v - \log a_v} \right\rceil \quad (3.19)$$

after some manipulation. It can be shown that s_v is a positive integer for all values of a_v and b_v when $a_v \neq b_v$. For $a_v = b_v$,

$$C_v = \frac{a_v}{s_v} + a_v = \frac{(s_v + 1)}{s_v} a_v \quad (3.20)$$

and the optimal value of s_v is

$$s_v = \left\lceil \frac{a_v}{Y - a_v} \right\rceil . \quad (3.21)$$

3.2.2 Preemptive Multiple Routing Model

In this case, each sublot is taken as a separate job. Again we will consider the case when there is a job v that satisfies (3.1), for otherwise lot streaming will not improve makespan. We start with the following lemma.

Lemma 3 *For each job ℓ,*

$$\frac{1}{2}(a_\ell + b_\ell) \;\le\; \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j.\} \tag{3.22}$$

Proof :

$$a_\ell + b_\ell \;\le\; \sum_{j=1}^{n} a_j + \sum_{j=1}^{n} b_j \;\le\; 2\max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\} \qquad \square \tag{3.23}$$

We will now show that two sublots of equal size for job v will be sufficient to reduce makespan to its physical limit, (3.17). Take these sublots as distinct jobs v_1 and v_2 with processing times $a_{v_1} = a_{v_2} = \frac{1}{2}a_v$ and $b_{v_1} = b_{v_2} = \frac{1}{2}b_v$. Then apply the Algorithm of Gonzales & Sahni given in Section 2.2 to the $n+1$ jobs. The optimal makespan will be,

$$C_{max} = \max\{\sum_{j=1}^{n+1} a_j, \sum_{j=1}^{n+1} b_j, \max_j(a_j + b_j).\} \tag{3.24}$$

Obviously $\sum_{j=1}^{n+1} a_j = \sum_{j=1}^{n} a_j$ and $\sum_{j=1}^{n+1} b_j = \sum_{j=1}^{n} b_j$. Then we have,

$$a_\ell + b_\ell < \max\{\sum_{j=1}^{n+1} a_j, \sum_{j=1}^{n+1} b_j\}, \text{ for } \ell \neq v_1 \text{ and } \ell \neq v_2 \tag{3.25}$$

from Lemma 1, and,

$$a_\ell + b_\ell < \max\{\sum_{j=1}^{n+1} a_j, \sum_{j=1}^{n+1} b_j,\} \text{ for } \ell = v_1 \text{ or } \ell = v_2, \tag{3.26}$$

from Lemma 2. Hence our actual makespan is,

$$C_{max} = \max\{\sum_{j=1}^{n} a_j, \sum_{j=1}^{n} b_j\} \tag{3.27}$$

which is the physical limit.

3.3 Job Shop Models

In this section, we analyze the problem of streaming multi jobs on two-machine job shops to minimize makespan, when the number of operations for each job is at most 2. We have 4 sets of jobs.

$$\begin{aligned}
A &= \{j \mid J_j \text{ is processed first on } M_1, \text{ next on } M_2\} \\
B &= \{j \mid J_j \text{ is processed first on } M_2, \text{ next on } M_1\} \\
C &= \{j \mid J_j \text{ is processed only on } M_1\} \\
D &= \{j \mid J_j \text{ is processed only on } M_2\}
\end{aligned}$$

Jackson [14] proposed the following algorithm to find the optimal schedule when there is no lot streaming.

Algorithm III

Step 1: Sequence jobs in C arbitrarily to give sequence S_C.

Step 2: Sequence jobs in D arbitrarily to give sequence S_D.

Step 3: Sequence jobs in A according to Johnson's rule to give sequence S_A

Step 4: Sequence jobs in B according to Johnson's rule to give sequence S_B

Step 5: An optimal schedule is (S_A, S_C, S_B) on M_1, (S_B, S_D, S_A) on M_2.

In the presence of lot streaming, the construction will be similar to the one above. We have to revise the Step 3 and Step 4 of the algorithm. This can be done by applying the time lag model to the jobs in A and B, rather then applying the Johnson's rule. Since there are no setups, the optimal sublot sizes of each job are given by the *geometric* pattern given by Equations 2.28 and 2.29.

To justify the argument, consider the jobs in $A \cup C$. Consider optimal schedule, for M_1 there will be no job in $B \cup D$ scheduled before any of the jobs in $A \cup C$, since otherwise it is possible to achieve the same or a shorter makespan by simple interchanges. Hence, jobs in $A \cup C$ will be scheduled first on M_1. A symmetric argument is valid for jobs in $B \cup D$ on machine M_2.

In order to sequence jobs in $A \cup C$, create a dummy a job J_d, such that

$$a_d = \ell_d = 0 \text{ and } b_d = \ell'_d = \sum_{j \in B \cup D} b_j$$

As mentioned in Section 3.1.1 geometric sublot sizes will create lags $\ell_j = a_j L_{j1}$ and $\ell_j = b_j L_{js_j}$ for each job in A. The jobs in D will clearly have $\ell_j = a_j$ and $\ell'_j = 0$. The dummy job will be scheduled first on M_1 by Johnson's rule, leaving enough time for jobs in $B \cup D$ to be processed on M_2. The jobs in A will be scheduled next on M_1, before the jobs in C which are sequenced arbitrarily.

Analogous construction can be made for jobs in $B \cup D$.

Chapter 4

Conclusions

The purpose of this study is to propose solution procedures for a number of lot streaming problems. The basic assumptions are similar to that of Baker [1], Potts & Baker [20] and Glass et. al. [12]. Namely, there are no setup times and the sublot sizes are decision variables. In addition to the detailed analysis mean flow time objective functions, problems in open shops and job shops are investigated.

In Chapter 2, two-machine single job flow shop lot streaming problems are studied in detail. Optimal consistent sublot sizes which minimize mean flow time are derived for the sublot completion time model. It is also shown within the chapter that consistent sublots do not always give the optimal mean flow time, contrary to the comments in [8]. For the general problem, optimal sublot sizes are conjectured. Further research is needed to prove these conjectures. The single job two-machine solutions may be useful in developing solution methods for single job m–machine problems as well as for the multi-job two-machine problems. Worst case performance bound of equal size sublots with mean flow time minimization is also given. The worst case performance of 18% shows that the use of equal sized transfers may be justified at least in two-machine flow shops.

The routing and streaming problem of a single job in an open shop to minimize makespan was also an area of research. The $m!$ possible number of routings is reduced to 2^{m-1} by showing that the optimal routing should be one of the *pyramidal* routings given any arbitrary sublot sizes. The results here may be also important in designing a flow shop.

Chapter 3 deals with the multi-job lot streaming problems. The preemptive two-machine flow shop multi-job problem with lot streaming remains still open. Streaming policy, that minimizes makespan in a two-machine open shop, is derived for two models. It is observed that, in most cases, the flexibility of open shops already allows for makespans which are very close to the physical limit (maximum of the total processing times on each machine). However, it is shown that lot streaming can be used to achieve this limit even in the presence of a job, whose total processing time determines the makespan. Finally, Jackson's algorithm [14] is revised to minimize makespan in multi-job, two-machine job shops.

Major drawback of lot streaming models is that the maximum number of transfers between machines is a parameter, rather than a decision variable. It is assumed that the material handling equipment is always available and transfer times are negligible. We suggest the following single transporter model, in which the transfers can take place whenever the transporter is available. The transfer times are positive and depend on the the two machines, between which the transfer takes place.

The problem is the following. There is a single job composed of U identical units to be processed in an m-stage flow shop. These U units are ready in the input buffer of M_1 at time 0. There is a single transporter, which starts its service always from a central location. q_i is the travel time from M_i to M_{i+1}, which also includes the loading and unloading times (q_m denotes the transfer time from M_m to the finished product inventory). r_i is the travel time from M_i to the central location (r_{m+1} is the travel time from finished product inventory to the central location) and vice versa. The objective is to complete all the units as soon as possible, minimize "makespan".

The model proposed to solve this problem is similar to the periodic review model of Benli [5] presented in Section 2.1.1. The period lengths are decision variables. Let h be an estimate of the maximum total number of transfers that can take place. Transfers can start at times $T_1, T_2, \ldots, T_h$. Define,

$$\begin{aligned} L_{i,t} &: \text{Number of units transferred to machine } i+1 \text{ at time } T_t, \\ Y_{i,t} &= \begin{cases} 1, & \text{if } L_{i,t} > 0, \\ 0, & \text{if } L_{i,t} = 0, \end{cases} \\ Z_t &: \text{the time at which } t\text{th transfer is completed.} \end{aligned}$$

Clearly, we have,

$$\sum_{i=1}^{m} Y_{i,t} \leq 1, \quad t = 1, \ldots, h. \tag{4.1}$$

The value of Z_t can be found by the following set of inequalities.

$$Z_t \geq T_t + q_i Y_{i,t}, \quad i = 1, \ldots, m, \quad t = 1, \ldots, h \tag{4.2}$$

with $T_0 = Z_0 = 0$. The $(t+1)$th transfer can start only after the transporter becomes available.

$$T_{t+1} \geq T_t + q_i Y_{i,t} + r_{i+1} Y_{i,t} + r_\ell Y_{\ell,t+1}, \quad i, \ell = 1, \ldots, m, \quad t = 1, \ldots, h-1 \tag{4.3}$$

with $T_1 = r_1$. If there is a transfer at time T_t from machine i, the output buffer of machine i is decreased by $L_{i,t}$ units at time T_t. Also, at time Z_t the input buffer of M_{i+1} is increased by $L_{i,t}$ units. To write the inventory balance equations, define,

$$\begin{aligned} X_{i,t} &: \text{Number of units produced on machine } i \text{ in } [T_{t-1}, Z_{t-1}], \\ \overline{X}_{i,t} &: \text{Number of units produced on machine } i \text{ in } [Z_{t-1}, T_t], \\ I_{i,t} &: \text{Number of units in the input buffer of machine } i \text{ at time } T_t, \\ \overline{I}_{i,t} &: \text{Number of units in the input buffer of machine } i \text{ at time } Z_t, \\ O_{i,t} &: \text{Number of units in the output buffer of machine } i \text{ at time } T_t, \\ \overline{O}_{i,t} &: \text{Number of units in the output buffer of machine } i \text{ at time } Z_t. \end{aligned}$$

The inventories must be in balance at time T_t

$$\begin{aligned} \overline{I}_{i,t-1} &= I_{i,t} + X_{i,t}, \quad i = 1, \ldots, m, \quad t = 1, \ldots, h, & (4.4) \\ \overline{O}_{i,t-1} + \overline{X}_{i,t} &= O_{i,t} + L_{i,t}, \quad i = 1, \ldots, m, \quad t = 1, \ldots, h, & (4.5) \end{aligned}$$

and at time Z_t,

$$I_{i,t} + L_{i-1,t} = \overline{I}_{i,t} + X_{i,t+1}, \quad i = 1,\dots,m, \quad t = 1,\dots,h, \tag{4.6}$$

$$O_{i,t} + X_{i,t+1} = \overline{O}_{i,t}, \quad i = 1,\dots,m, \quad t = 1,\dots,h, \tag{4.7}$$

with

$$O_{i,0} = \overline{O}_{i,0} = 0, \quad i = 1,\dots,m, \tag{4.8}$$

$$L_{0,t} = 0, \quad t = 1,\dots,h, \tag{4.9}$$

$$I_{i,0} = \overline{I}_{i,0} = 0, \quad i = 2,\dots,m, \tag{4.10}$$

$$I_{1,0} = \overline{I}_{1,0} = U, \tag{4.11}$$

$$\sum_{t=1}^{h} L_{m,t} = U. \tag{4.12}$$

The transfers take place only if they are indicated,

$$L_{i,t} \leq \mu Y_{i,t}, \quad i = 1,\dots,m, \quad t = 1,\dots,h. \tag{4.13}$$

where μ is a very large number or the capacity of the transporter. There are production capacity constraints,

$$p_i X_{i,t} \leq T_{t-1} - Z_{t-1}, \quad i = 1,\dots,m, \; t = 1,\dots,h, \tag{4.14}$$

$$p_i \overline{X}_{i,t} \leq T_t - Z_{t-1}, \quad i = 1,\dots,m, \; t = 1,\dots,h. \tag{4.15}$$

Finally, there are non-negativity and integrality constraints,

$$T_t, Z_t \geq 0, \quad t = 1,\dots,h \tag{4.16}$$

$$X_{i,t}, \overline{X}_{i,t}, L_{i,t} \geq 0, \quad i = 1,\dots,m, \quad t = 1,\dots,h, \tag{4.17}$$

$$I_{i,t}, \overline{I}_{i,t}, O_{i,t}, \overline{O}_{i,t} \geq 0, \quad i = 1,\dots,m, \quad t = 1,\dots,h, \tag{4.18}$$

$$Y_{i,t} \in \{0,1\}, \quad i = 1,\dots,m, \quad t = 1,\dots,h. \tag{4.19}$$

Then our mixed integer linear program will be,

$$\min \; Z_h \tag{4.20}$$

subject to the Constraints (4.1)-(4.19). Note that Z_h will be the time at which all units are completed and transferred to the end product inventory. It may be the case that there will be less than h transfers, then we will have for some $v < h$

$$T_t = Z_t = Z_{v-1}, \quad t = v, \ldots, h-1,$$

or

$$Y_{i,t} = 0 \quad t = v, \ldots, h-1 \quad i = 1, \ldots, m.$$

For further research, it will be appropriate to analyze this model to make it computationally feasible. Some of the periods may be defined as *active periods* as in the model of Benli [5]. Computational experience can be helpful in observing some special structures of the problem like periodicity of transfers. Extension of this would be construction of the models and heuristic procedures for the problems with more transporters.

Bibliography

[1] K. R. Baker. Lot streaming to reduce cycle time in a flow shop. Working Paper 203, Amos Tuck School, Dartmouth College, 1988. Presented at the Third Annual Conference on Current Issues in Computer Science and Operations Research, Bilkent University, June 1988.

[2] K. R. Baker. Lot streaming in the two machine flow shop with setup times. Working Paper 297, The Amos Tuck School, Dartmouth College, 1993.

[3] K. R. Baker and D. Jia. A comparative study of lot streaming procedures. *OMEGA Int. J. of Mgmt. Sci.*, 21:561–566, 1993.

[4] K. R. Baker and D. F. Pyke. Solution procedures for the lot-streaming problem. *Decision Sciences*, 21:475–491, 1990.

[5] Ömer Benli. Streaming a single job in a flow shop. Research Report IEOR-9409, Department of Industrial Engineering, Bilkent University, 1994.

[6] Jimmie Browne, John Harhen, and James Shivnan. *Production Management Systems: A CIM Perspective.* Addison-Wesley, 1988.

[7] Ferda C. Çetinkaya. Lot streaming in a two-stage flow shop with set-up, processing and removal times separated. *J. Opl Res. Soc.*, 45, 1994. forthcoming.

[8] Ferda C. Çetinkaya and Jatinder N. D. Gupta. Flowshop lot streaming to minimize total weighted flow time. Research Memorandum 94-24, Industrial Engineering, Purdue University, 1994.

[9] Ferda C. Çetinkaya and M. Sinan Kayalıgil. Unit sized transfer batch scheduling with setup times. *Computers and Industrial Engineering*, 22:177–183, 1992.

[10] Stephane Dauzere-Peres and Jean-Bernard Lasserre. An iterative procedure for lot streaming in job shop scheduling. *Computers and Industrial Engineering*, 25:231–234, 1993.

[11] D. W. Fogarty, J. H. Blackstone, and T. R. Hoffman. *Production and Inventory Management.* South-Western Pub. Co., Cincinnati, Ohio, 1991.

[12] C. A. Glass, J. N. D. Gupta, and C. N. Potts. Lot streaming in three-stage production processes. *European J. Operations Research*, 75:378–394, 1994.

[13] T. Gonzales and S. Sahni. Open shop scheduling to minimize finish time. *J. Assoc. Comput. Mach.*, 23:665–679, 1976.

[14] J. R. Jackson. An extension of Johnson's results on job-lot scheduling. *Naval Research Logistics Quarterly*, 3:201–224, 1956.

[15] S. M. Johnson. Optimal two and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1:61–68, 1954.

[16] D. H. Kropp and T. L. Smunt. Optimal and heuristic models for lot splitting in a flow shop. *Decision Sciences*, 21:691–709, 1990.

[17] R. Lundrigan. What is this thing called OPT? *Prod. and Inv. Mgmt.*, 27:2–12, 1986.

[18] L. G. Mitten. Sequencing n jobs on two machines with arbitrary time lags. *Mgmt Sci.*, 5:293–298, 1959.

[19] S. Miyazaki and N. Nishiyama. Analysis for minimizing weighted flow-time in flow-shop scheduling. *J. Opl Res. Soc. Japan*, 23:118–131, 1980.

[20] C. N. Potts and K. R. Baker. Flow shop scheduling with lot streaming. *Opns. Res. Lett.*, 8:297–303, 1989.

[21] C. N. Potts and L. van Wassenhove. Integrating scheduling with batching and lot-sizing: a review of algorithms and complexity. *J. Opl Res. Soc.*, 48:395-406, 1992.

[22] A. H. G. Rinnooy Kan. *Machine Scheduling Problems: Classification, Complexity, and Computations*. Martinus Nijhoff, The Hague, 1976.

[23] M.L. Smith, S.S. Panwalkar, and R.A. Dudek. Flowshop sequencing problem with ordered processing time matrices: A general case. *Mgmt Sci.*, 21:544–549, 1975.

[24] G. Steiner and W. G. Truscott. Batch scheduling to minimize cycle time, flow time and processing cost. *IIE Transactions*, 25:90–97, 1993.

[25] D. Swann. Using MRP for optimized schedules (emulating OPT). *Prod. and Inv. Mgmt.*, 27:30–37, 1986.

[26] A. Z. Szendrovits. Manufacturing cycle time determination for a multi-stage economic production quantity model. *Mgmt Sci.*, 22:298–308, 1975.

[27] Alper Şen, Engin Topaloğlu, and Ömer Benli. Optimal streaming of a single job in a two stage flow shop. Research Report IEOR-9413, Department of Industrial Engineering, Bilkent University, 1994.

[28] Engin Topaloğlu, Alper Şen, and Ömer Benli. Optimal streaming of a single job in a flow shop with two sublots. Research Report IEOR-9412, Department of Industrial Engineering, Bilkent University, 1994.

[29] D. Trietsch. Optimal transfer lots for batch manufacturing: A basic case and extensions. Technical Report NPS-54-87-010, Naval Postgraduate School, Monterey, CA, 1987.

[30] D. Trietsch and K. R. Baker. Basic techniques for lot streaming. *Ops Res.*, 41:1065–1076, 1993.

[31] W. Truscott. Scheduling production activities in multistage manufacturing systems. *Int. J. Prodn Res.*, 23:315–328, 1985.

[32] M. Michael Umble and M. L. Srikanth. *Synchronous Manufacturing: Principles For World Class Excellence.* APICS South-Western series in production and operations management. South-Western Publishing, Cincinnati, Ohio, 1990.

[33] R. G. Vickson. Optimal lot streaming for multiple products in a two-machine flow shop. *European J. Operations Research*, 1994. forthcoming.

[34] R. G. Vickson and B. E. Alfredsson. Two- and three-machine flow shop scheduling problems with equal sized transfer batches. *Int. J. Prodn Res.*, 30:1551–1574, 1992.

[35] T. Vollmann. OPT as an enhancement to MRPII. *Prod. and Inv. Mgmt.*, 27:38–46, 1986.

[36] E. F. Williams and S. Tufekci. Polynomial time algorithms for the $m \times 2$ lot streaming problem. Research Report 92-10, Department of Industrial and Systems Engineering, University of Florida, 1992.