

8-2015

Integrated Batching and Lot Streaming with Variable Sublots and Sequence-Dependent Setups in a Two-Stage Hybrid Flow Shop

Shasha Wang
Clemson University

Follow this and additional works at: https://tigerprints.clemson.edu/all_theses

Recommended Citation

Wang, Shasha, "Integrated Batching and Lot Streaming with Variable Sublots and Sequence-Dependent Setups in a Two-Stage Hybrid Flow Shop" (2015). *All Theses*. 2497.

https://tigerprints.clemson.edu/all_theses/2497

This Thesis is brought to you for free and open access by the Theses at TigerPrints. It has been accepted for inclusion in All Theses by an authorized administrator of TigerPrints. For more information, please contact kokeefe@clemson.edu.

**INTEGRATED BATCHING AND LOT STREAMING WITH VARIABLE
SUBLOTS AND SEQUENCE-DEPENDENT SETUPS IN A TWO-STAGE
HYBRID FLOW SHOP**

A Thesis
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
Industrial Engineering

by
Shasha Wang
August 2015

Accepted by:
Dr. Scott J. Mason, Committee Chair
Dr. J. Cole Smith
Dr. Mary E. Kurz

ABSTRACT

Consider a paint manufacturing firm whose customers typically place orders for two or more products simultaneously: liquid primer, top coat paint, and/or undercoat paint. Each product belongs to an associated product family that can be batched together during the manufacturing process. Meanwhile, each product can be split into several sublots so that overlapping production is possible in a two-stage hybrid flow shop. Various numbers of identical capacitated machines operate in parallel at each stage. We present a mixed-integer programming (MIP) to analyze this novel integrated batching and lot streaming problem with variable sublots, incompatible job families, and sequence-dependent setup times. The model determines the number of sublots for each product, the size of each subplot, and the production sequencing for each subplot such that the sum of weighted completion time is minimized. Several numerical example problems are presented to validate the proposed formulation and to compare results with similar problems in the literature. Furthermore, an experimental design based on real industrial data is used to evaluate the performance of proposed model. Results indicate that the computational cost of solving the model is high.

ACKNOWLEDGMENTS

First and foremost, I would like to express my gratitude to my advisor, Dr. Scott J. Mason. Without his support and guidance, this work would not have been possible. His encouragement and useful comments inspired me.

I would like to thank my other committee members, Dr. J. Cole Smith and Dr. Mary E. Kurz for their suggestions and comments in completing of my thesis. Special thanks go to my former colleagues Na Wen and Dazhi Zhang, who provided me real industrial data and paint knowledge support.

I also profoundly thank my parents for supporting me and encouraging me with their best wishes. Finally, many thanks to my husband, Tianwei Wang, who was always there for me and stood by me through both good times and bad.

TABLE OF CONTENTS

	Page
TITLE PAGE.....	i
ABSTRACT	ii
ACKNOWLEDGMENTS	iii
TABLE OF CONTENTS	iv
LIST OF TABLES.....	vi
LIST OF FIGURES	vii
INTRODUCTION AND RESEARCH MOTIVATION.....	1
1.1 Introduction.....	1
1.2 Research Motivation.....	4
1.3 Research Contribution	6
1.4 Thesis Overview	7
LITERATURE REVIEW	8
2.1 Batch Scheduling	8
2.2 Lot Streaming.....	9
2.2.1 <i>Equal Sublots</i>	10
2.2.2 <i>Consistent Sublots</i>	10
2.2.3 <i>Variable Sublots</i>	11
PROBLEM STATEMENT.....	13
MATHEMATICAL MODEL.....	16
4.1 MIP Formulation.....	16
4.2 Establishing Values for Big M Parameters	22
NUMERICAL EXPERIMENTS	24

Table of Contents (Continued)

	Page
5.1 Model Validation	24
5.2 Model Demonstration	27
5.3 Experimental Study.....	31
CONCLUSIONS AND FUTURE RESEARCH	35
REFERENCES	37

LIST OF TABLES

Table		Page
5.1	Data for Example Problems	27
5.2	Processing Times for Example Problems	27
5.3	Setup Times for Each Product Family	28
5.4	Experimental Design for Model Evaluation	32
5.5	Number of Orders and Number of Products in each Order	32
5.6	Processing Times for Job Families	32
5.7	Setup Times for Job Families	33
5.8	Optimality Gaps for Instances	34

LIST OF FIGURES

Figure		Page
1.1	An Example of Sublot Type: (a) Equal Sublots (b) Consistent Sublots (c) Variable Sublots.....	4
1.2	Paint Production Process.....	6
3.1	Example of Proposed Problem.....	15
5.1	Optimal Solution of Example 2 (a) Integrated Batching with Lot Streaming (b) Lot Streaming.....	29
5.2	Optimal Solution of Example 3	30

CHAPTER ONE

INTRODUCTION AND RESEARCH MOTIVATION

1.1 Introduction

As manufacturing enterprises continue to endure market pressures, reducing costs and improving customer satisfaction remain key factors for successful businesses. Effective supply chain scheduling is one methodology companies have turned to for increasing their manufacturing productivity. A supply chain system is composed of procurement, production, and distribution processes. Raw materials are purchased from suppliers, and then goods are produced at one or more manufacturing plants, distributed to distribution centers or warehouses for storage, and finally delivered to customers or retailers.

Scheduling is a crucial decision-making process in any system that is performed at a variety of temporal levels. Medium-term supply chain scheduling (planning) considers allocating jobs to sequencing and timeframe decisions for completing customer orders to minimize cost-related objectives, while short-term supply chain scheduling considers allocation decisions to a specific resource (e.g., machine or people) over a shorter time horizon (e.g., a shift or days).

Batching and lot streaming are two concepts and methods dealing with problems involving treatment of lots in scheduling theory (Burtseva *et al.*, 2012). Batching is usually used to help reduce setup time and costs in real world industry settings. The primary advantage of lot streaming lies in its reduction of makespan (Sarin and Jaiprakash, 2007).

A product family can be defined as products that have the same properties or manufacturing attributes—such as shape, size, or color—or that require the same raw materials. If some products at a manufacturing plant belong to the same product family, they can be sorted out to form a batch. The products within a batch are processed on the same machine simultaneously. Usually, there is no setup between each product within a batch thus saving setup time and cost. Batch scheduling focuses on finding capacity-feasible schedules that optimize given objective function(s) while meeting all required constraints (Cakici *et al.*, 2013). Batch scheduling research typically assumes that batches cannot be split during the manufacturing process (Kopanos *et al.*, 2011; Tai *et al.*, 2011; Amorim *et al.*, 2011; Mason *et al.*, 2010; Fumero *et al.*, 2014).

If a batch is allowed to be split into several sublots, the problem is usually called lot streaming, which was first introduced by Reiter (1966). Lot streaming problem focuses more on when and how to split a batch since the batching decision is already made. In a flow shop or job shop, lot streaming allows sublots to be processed in overlapping fashion on successive stages or machines in order to optimize some performance criteria. Besides reducing makespan, other advantages of applying lot streaming include reductions in cycle time, average work-in-process inventory, required storage space, and material handling equipment requirements (Cheng *et al.*, 2013). While lot streaming problems focus on improving performance by dividing product lots into several sublots, three key decisions must be made: 1) the number of sublots to create, 2) the size of each subplot, and 3) the processing sequence of the sublots.

Feldmann and Biskup (2008) categorize lot streaming problems according to machine configuration, product type, subplot type, and other criteria. Equal sublots refer to the case wherein the size of all sublots is fixed and equal for all products. Problems with consistent sublots allow for each product to have its own, potentially unique, subplot size that remains constant for all stages/processes. Finally, variable sublots cases contain no restrictions on subplot sizes across machines. Consider the example case in Figure 1.1 containing a batch of 84 items to be processed on three machines. The processing time for machines 1, 2, and 3 are 2, 1, and 3 minutes per unit, respectively. In Figure 1.1a, the job is split into two equal sublots of 42 items each, resulting in a makespan of 378 minutes. Next, consistent sublots of 36 and 48 items are shown in Figure 1.1b—the resulting makespan is reduced to 360 minutes. Finally, the variable subplot case in Figure 1.1c depicts 56 of 84 items being processed and transferred as the first subplot on machine 1, with the remaining 28 items comprising the second subplot. Alternately, the first 21 items are sent as the first subplot on machine 2 to machine 3 once they are completed. The remaining 63 items then are processed and transferred as the second subplot to machine 3, resulting in a makespan of 385 minutes.

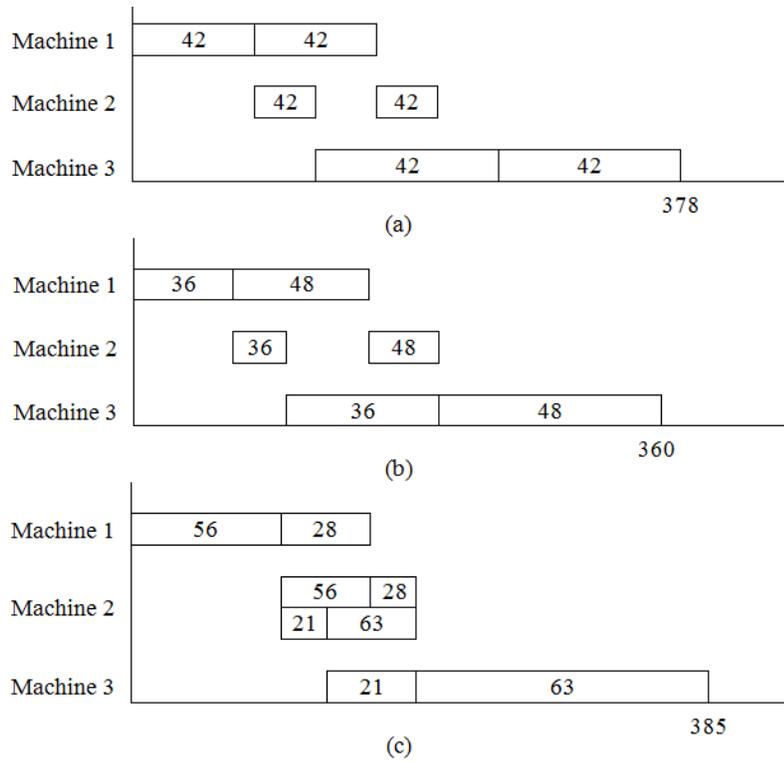


Figure 1.1 An example of subplot type: (a) Equal Sublots (b) Consistent Sublots (c) Variable Sublots (Trietsch and Baker, 1993; Morakotkarn, 1995)

1.2 Research Motivation

The practical motivation of this proposed research is the author's work experience at a coating company. Customers place orders for one of two available product groups according to their requirements, such as painting cargo containers, painting ship hulls, or painting other large structures. One product group consists of primer and top coat paint; the other is composed of primer, top coat paint, and undercoat paint. A product in any order (i.e., primer, top coat paint, or undercoat paint) can be divided in to hundreds of subcategories according to its formulation and color (e.g., gray epoxy zinc-rich primer,

red epoxy micaceous iron oxide undercoat paint, and blue acrylic top coat paint). Each subcategory is associated with an incompatible product family such that two products cannot be processed in the same batch if they belong to different product families. Sequence-dependent setup times are unavoidable when production switches from one product family to another product family (Schultmann *et al.* 2006).

Figure 1.2 overviews the basic production steps required to manufacture each coating system component. Raw materials such as resin, pigment, and solvent are pre-mixed in a container, and then are milled into fine particles. In order to produce each specific customer-requested item, additional materials such as resins, hardeners, additives, and/or solvents are added to the milled “base” mixture and blended to produce the required viscosity, fineness, brightness, and color properties. The blended final product is then packed into barrels. The manufacturing environment resembles a two-stage hybrid flow shop. A hybrid flow shop in this study refers to a flow shop with multiple stages where, in at least one stage, multiple identical machines are operated in parallel (Kurz and Askin, 2004). Each batch needs to be processed by only one machine at each stage. Six identical containers with a specific capacity operate in parallel at stage 1; pre-mixing and milling processes are completed in the same stage 1 container. There are 18 capacitated vessels working in parallel in stage 2’s blending operations. After blending, the completed coating system component (paint) will be packed into barrels—we assume unlimited packing resources are available.

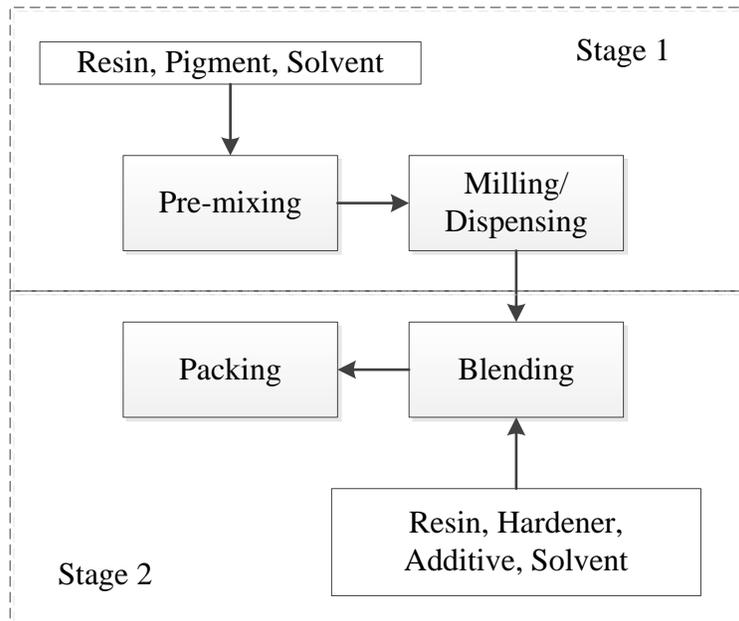


Figure 1.2 Paint Production Process

If the demand for each product is so large that it exceeds machine capacity, then each product has to be divided into several sublots to be processed. Each subplot is considered as a batch. If one subplot size is smaller than machine capacity, sublots of other products (belong to the same product family) can be manufactured in this batch to achieve the machine capacity. Therefore, batch scheduling and lot streaming decisions have to be made simultaneously in one model. However, few studies consider the integration of them in one model.

1.3 Research Contribution

In the past 30 years, batch scheduling and lot streaming are well studied in isolation. In batch scheduling problems, researchers focus on how to group products to

form a batch and how to sequence them on machines. A batch cannot be split during the manufacturing process. In lot streaming problems, batches are already given so the batching process is not considered. The efforts are only made to when and how to split batches. This study introduces a mathematical model that incorporates batching and lot streaming to determine the subplot sizes and sequences for multiple products in a two-stage hybrid flow shop environment to minimize the sum of total weighted completion times for product sublots while satisfying customer demand.

1.4 Thesis Overview

The remainder of the thesis is organized as follows. Chapter 2 is a literature review on batch scheduling and lot streaming. The proposed research problem is described in Chapter 3. Chapter 4 presents a detailed mathematical model to formulate the problem objective function and constraints. A series of numerical cases and tests were implemented and discussed in Chapter 5. The conclusion and future plan is given in Chapter 6.

CHAPTER TWO

LITERATURE REVIEW

2.1 Batch Scheduling

Batch scheduling integrates scheduling and batching decisions. Batching occurs when jobs share the same setup on a machine (family scheduling) or when a machine can process several jobs simultaneously (batching machine). Potts and Kovalyov (2000) provide a review of batch scheduling on the above two types of models. Erramilli and Mason (2006) investigate the multiple orders per job batch scheduling problem with compatible job families wherein jobs that belong to any family may be grouped to form a production batch. A mixed-integer programming (MIP) formulation is presented to minimize the total weighted tardiness in a single machine environment. In order to find near-optimal solutions in a reasonable amount of computation time, a simulated annealing-based heuristic is presented. Erramilli and Mason (2008) consider the same problem with incompatible job families in which only jobs from the same family can be batched together. Cakici *et al.* (2013) consider batch scheduling with dynamic job arrivals and incompatible job families in a parallel machine environment. Both a mathematical model and a heuristic algorithm are proposed to minimize the total weighted completion time.

Lin and Liao (2013) study a scheduling problem in a two-stage assembly shop to minimize weighted sum of makespan, total completion time, and total tardiness. The proposed model combines a job dividing strategy and batch processing in which jobs are divided into several sub-jobs and processed separately, but simultaneously by workers in

stage 1. Three heuristics are developed for solving medium- and large-sized instances. Huang and Lin (2013) study batch scheduling in a differential flow shop where the stage 1 machines process jobs in batches to minimize makespan. A dynamic programming algorithm is developed to solve a special case that in turn derives a lower bound for general cases. Fu *et al.* (2012) consider a differential flow shop scheduling problem with limited buffers and incompatible job families to minimize mean completion time. Behnamian *et al.* (2012) examine a three-machine flow shop where a stage 2 batch-processing machine is located between two discrete machines in stages 1 and 3. Both a MIP model and a heuristic algorithm are proposed to minimize makespan. Batch scheduling with sequence-independent setup times and with sequence-dependent setup times are studied by Pranzo (2004) and Logendran *et al.* (2006).

2.2 Lot Streaming

Cheng *et al.* (2013) review lot streaming problems for two categories: time-based objective functions and cost-based objective functions. Machine environments such as flow shops, parallel machines, hybrid flow shops, job shops, open shops, and two-stage assembly systems are discussed. An earlier review can be found in Chang and Chiu (2005). Trietsch and Baker (1993) provide basic models and algorithms for the lot streaming problem and present complexity classifications for some lot streaming problems.

2.2.1 *Equal Sublots*

Yalaoui and Chu (2006) develop a branch-and-bound algorithm to minimize total completion time in a parallel machine environment with lot streaming. Tseng and Liao (2008) propose a discrete particle swarm approach to find good solutions for n job, m machine lot streaming flow shops that minimize total weighted earliness and tardiness. Pan and Ruiz (2012) consider the same problem with sequence-dependent setup times to minimize makespan. Kalir and Sarin (2001) study the lot streaming problem of sequencing a set of batches in a flow shop to minimize makespan. A heuristic is proposed to minimize bottleneck.

2.2.2 *Consistent Sublots*

Mortezaei and Zulkifli (2014) propose a MIP for multi-product lot sizing and lot streaming in a flow shop. The objective is to minimize production costs, holding costs, and makespan costs. Interestingly, two cases are considered: all machines are available and all machines need preventive maintenance. Gasquet *et al.* (2012) present a MIP model for the m stage flow shop lot streaming problem with sequence dependent setup times to minimize makespan. Zhang *et al.* (2005) study multi-job lot streaming in a two-stage hybrid flow shop with m identical machines at stage 1 and a single machine at the second stage. A MIP formulation is used to calculate a lower bound and then two heuristic methods are proposed to solve this problem to minimize mean completion time.

Feldmann and Biskup (2008) study lot streaming with multiple products in a multi-stage permutation flow shop. Sublots with and without intermingling are

investigated. Increasing either the number of sublots or the number of stages reduces makespan when lot streaming is applied in multi-stage setting. Furthermore, intermingling is beneficial to the lot streaming as compared with non-intermingling.

Defersha and Chen (2012a) consider lot streaming in a hybrid flexible flow shop with sequence-dependent setup times, release time for machines, and machine eligibility constraints. A MIP formulation is presented to minimize the makespan. To deal with model tractability issues, a parallel genetic algorithm is proposed. Martin (2009) develops a hybrid genetic approach for a m machine flow shop with lot streaming of multiple products using consistent sublots and intermingling; a similar study with variable sublots is conducted by the same authors (Defersha and Chen, 2010).

Ghasemi (2008) investigates lot streaming multiple products with consistent sublots in hybrid flow shops. However, the parallel machines in each stage are non-identical. Both attached and sequence-independent setups are considered in a MIP model to minimize the makespan. The author modifies the proposed model to accommodate lot streaming with variable sublots of a single product in a multiple stage hybrid flow shop. In contrast to our study, this paper does not consider batching multiple products or sequence-dependent setups.

2.2.3 *Variable Sublots*

The vast majority of the available lot streaming literature analyzes variable subplot problems using heuristic algorithms. Pan *et al.* (2011) develop a discrete artificial bee colony algorithm for the lot streaming flow shop scheduling problem to minimize total

weighted earliness and tardiness. Additional research efforts include those of Sen and Benli (1998) and Liu (2003), as well as Goyal and Szendrovits (1986).

Biskup and Feldmann (2006) present the first MIP model for lot streaming with variable sublots and subplot availability constraints. The authors also demonstrate that the use of variable sublots can lead to large improvements in makespan. Defersha and Chen (2010) extend this model to the multiple product case and develop a hybrid genetic algorithm to improve computational efficiency. Chiu *et al.* (2004) investigate a lot streaming problem with a limited number of capacitated transporters in a multi-stage batch production system. Both attached and detached setups are considered while minimizing makespan and transportation cost. A mathematical model and two heuristic methods are proposed. Defersha and Chen (2012b) study the lot streaming problem in a job shop with routing flexibility, sequence-dependent setups, machine release dates, and lag time constraints. An island-model parallel genetic algorithm is presented.

CHAPTER THREE

PROBLEM STATEMENT

Although batch scheduling and lot streaming have been well studied in isolation, the problem of integrating batching and lot streaming has not been addressed in the literature. Consider a two-stage hybrid flow shop: m_1 identical parallel, capacitated machines operate in stage 1, while m_2 ($m_2 \geq m_1$) identical parallel, capacitated machines comprise stage 2. A set of customer orders of varying weights (priorities) is released at the beginning of the time horizon of interest. All products within a customer order have the same weight (priority). Each product can be divided into several sublots that may vary in size. Two or more sublots, which are possibly from different products, can be processed simultaneously on the same machine as a batch if 1) they belong to the same product family and 2) their total size does not exceed the machine capacity. A sequence-dependent setup time is required for changeovers at each machine. We seek to determine the number of sublots for each product, the size of each subplot, and the corresponding sequences for each subplot such that the sum of total weighted completion times for product sublots is minimized.

The proposed problem can be considered as an integration of two problems: the multiple orders per job batch scheduling problem with incompatible jobs and sequence-dependent setups in a two-stage hybrid flow shop, and lot streaming problem with variable sublots and subplot availability in a two-stage hybrid flow shop. Based on classification scheme of lot streaming provided by Cheng (2013), this problem can be denoted as $Fm1 + m2/n/V/CV/S(a)/\Sigma wC$. Alternately, using the scheduling notation

scheme of Graham *et al.* (1979), this problem can be denoted $\mathfrak{a} HF2|lot, incompatible, p$ – batch, split, $s_{ij}|\sum wC$.

Gupta (1988) prove that two-stage hybrid flow shop problem is NP-complete in case of $\max (M^{(1)}, M^{(2)}) > 1$. Biskup and Feldmann (2006) argue that the multi-stage, variable sublots, subplot availability (MVS) lot streaming problem is probably NP-hard, although the complexity status is still open. In our proposed problem, we relax three assumptions considered in their study: 1) there is only one machine operating at each stage, 2) sublots are not allowed to be batched, and 3) setup times are ignored. After relaxing these assumptions, the MVS lot streaming problem reduces to our problem. It follows that given these two statements, in combination with the existence of sequence-dependent setups, cause our problem under study to be NP-hard.

Figure 3.1 shows an example instance of the research problem under study as motivated by paint production. Order 1 contains customer demand for two products (i.e., a primer and a top coat), while order 2 consists of three products including primer, undercoat paint, and a top coat. The five products in the two orders belong to four product families. The top coat requirements in order 1 and order 2 belong to the same product family, so they can be processed simultaneously. The proposed model will evaluate this decision such that they could be batched together on stage 1's second machine, for example. Both top coats in order 1 (T1) and order 2 (T2) are split into two sublots at stage 1.

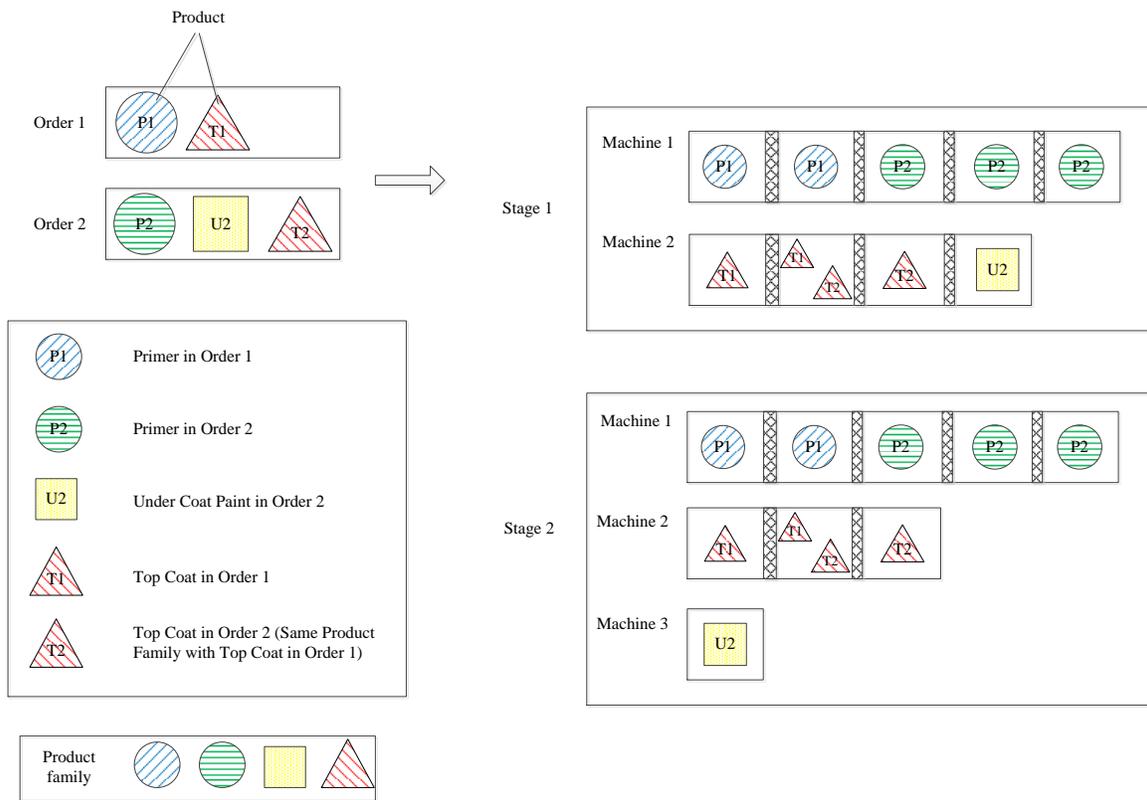


Figure 3.1 Example of Proposed Problem

CHAPTER FOUR
MATHEMATICAL MODEL

4.1 MIP Formulation

We formulate the integrated batching and lot streaming problem in a two-stage flow shop as a MIP. We model two inherent goals of the problem in a monolithic model: determining the size of individual sublots and sequencing the sublots. The notation used in the mathematical model is defined as follows:

Sets

P	Set of products; indexed by $p = 1, 2, \dots, P $
S	Set of flow shop stages; indexed by $s = 1, 2$
W_s	Set of machines in stage s ; indexed by $k, m = 1, 2, \dots, W_s $
B	Set of batch positions; indexed by $j, b = 0, 1, 2, \dots, B $
N	Set of sublots; indexed by $\alpha = 1, 2, \dots, N $
F	Set of product families; indexed by $f, g = 0, 1, 2, \dots, F $

Initially, a maximum number of sublots $|N|$ is given to any product p by a decision maker. Not all these sublots are necessary to be used. $|B|$ is the maximum number of batches that any machine can process. Batch position 0 is a dummy batch position that only dummy product family 0 can be assigned to it.

Parameters

K_s	Capacity of each identical machine in stage s
D_p	Demand for product p
M_1, M_2, M_3	Large positive numbers

t_{fs}	Processing time of product family f in stage s
w_p	Weight of product p
ρ_{pf}	=1 if product p belongs to product family f , 0 otherwise
τ_{fg}	Setup time between product family f and g

Variables

$n_{p\alpha smb}$	Size of the α^{th} subplot of product p in the b^{th} batch position on machine m in stage s
A_{fsmb}	Starting time of the b^{th} batch position (belongs to product family f) on machine m in stage s
δ_{fsmb}	Completion time of the b^{th} batch position (belongs to product family f) on machine m in stage s
C_{pasmb}	Completion time of the α^{th} subplot of product p assigned to the b^{th} batch position on machine m in stage s
u_{pa}	Binary variable equals to 1 if subplot α of product p is produced
$x_{p\alpha smb}$	Binary variable equals to 1 if the α^{th} subplot of product p is assigned to the b^{th} batch position on machine m in stage s , 0 otherwise
y_{fsmb}	Binary variable equals to 1 if the b^{th} batch position (processes product family f) on machine m in stage s is used, 0 otherwise
z_{pakjmb}	Binary variable equals to 1 if the α^{th} subplot of product p is successively assigned to the j^{th} batch position on machine k in

stage 1 and the b^{th} batch position on machine m in stage 2, 0 otherwise

$\beta_{fbg,b+1}^{sm}$ Binary variable equals to 1 if product family g in the $(b+1)^{th}$ batch position is processed immediately after product family f in the b^{th} batch position on machine m stage s , 0 otherwise

Using the above notation, the objective function and constraints of the proposed MIP model for integrated batching and lot streaming with variable sublots in a two-stage hybrid flow shop is as follows:

Minimize:

$$Z = \sum_{p \in P} \sum_{\alpha \in N} \sum_{m \in W_2} \sum_{b \in B_m} w_p C_{p\alpha 2mb} \quad (1)$$

Subject to:

$$n_{p\alpha smb} \leq M_1 x_{p\alpha smb}, \forall p \in P, \forall \alpha \in N, \forall s \in S, \forall m \in W_s, \forall b \in B \quad (2)$$

$$\sum_{p \in P} \sum_{\alpha \in N} n_{p\alpha smb} \geq 0.001 \sum_{f \in F} y_{fsm}, \forall s \in S, \forall m \in W_s, \forall b \in B \quad (3)$$

$$\sum_{p \in P} \sum_{\alpha \in N} n_{p\alpha smb} \leq K_s \sum_{f \in F} y_{fsm}, \forall s \in S, \forall m \in W_s, \forall b \in B \quad (4)$$

$$\sum_{\alpha \in N} \sum_{m \in W_1} \sum_{b \in B} n_{p\alpha 1mb} = D_p, \forall p \in P \quad (5)$$

$$y_{0sm0} = 1, \forall s \in S, \forall m \in W_s \quad (6)$$

$$y_{0sm0} - \sum_{g \in F} y_{gsm1} \geq 0, \forall s \in S, \forall m \in W_s \quad (7)$$

$$\sum_{f \in F} y_{fsm} - \sum_{g \in F} y_{gsm,b+1} \geq 0, \forall s \in S, \forall m \in W_s, \forall b \in B \setminus \{|B|\} \quad (8)$$

$$\begin{aligned} \rho_{pf} x_{p\alpha smb} - y_{fsmb} &\leq 0, \forall p \in P, \forall \alpha \in N, \forall f \in F, \forall s \in S, \forall m \in W_s, \\ \forall b \in B \end{aligned} \quad (9)$$

$$\begin{aligned} y_{fsmb} + y_{gsm,b+1} - \beta_{fbg,b+1}^{sm} &\leq 1, \forall f \in F \cup \{0\}, \forall g \in F, \forall s \in S, \forall m \in W_s, \\ \forall b \in (B \cup \{0\}) \setminus \{|B|\} \end{aligned} \quad (10)$$

$$\begin{aligned} x_{p\alpha 1kj} + x_{p\alpha 2mb} - z_{p\alpha kjmb} &\leq 1, \forall p \in P, \forall \alpha \in N, \forall k \in W_1, \forall m \in W_2, \\ \forall j, b \in B \end{aligned} \quad (11)$$

$$\sum_{k \in W_1} \sum_{j \in B} x_{p\alpha 1kj} = u_{p\alpha}, \forall p \in P, \forall \alpha \in N \quad (12)$$

$$\sum_{m \in W_2} \sum_{b \in B} x_{p\alpha 2mb} \geq \sum_{k \in W_1} \sum_{j \in B} x_{p\alpha 1kj} - M_2(1 - u_{p\alpha}), \forall p \in P, \forall \alpha \in N \quad (13)$$

$$\sum_{m \in W_2} \sum_{b \in B} x_{p\alpha 2mb} \leq \sum_{k \in W_1} \sum_{j \in B} x_{p\alpha 1kj} + M_2 u_{p\alpha}, \forall p \in P, \forall \alpha \in N \quad (14)$$

$$\begin{aligned} n_{p\alpha 1kj} &\geq \sum_{m \in W_2} \sum_{b \in B} n_{p\alpha 2mb} + M_1 \left(\sum_{m \in W_2} \sum_{b \in B} z_{p\alpha kjmb} - \sum_{m \in W_2} \sum_{b \in B} x_{p\alpha 2mb} \right), \forall p \in P, \\ \forall \alpha \in N, \forall k \in W_1, \forall j \in B \end{aligned} \quad (15)$$

$$n_{p\alpha 1kj} \leq \sum_{m \in W_2} \sum_{b \in B} n_{p\alpha 2mb}, \forall p \in P, \forall \alpha \in N, \forall k \in W_1, \forall j \in B \quad (16)$$

$$n_{p\alpha 1kj} \leq M_1 \sum_{m \in W_2} \sum_{b \in B} z_{p\alpha kjmb}, \forall p \in P, \forall \alpha \in N, \forall k \in W_1, \forall j \in B \quad (17)$$

$$\begin{aligned} C_{p\alpha smb} &\geq \delta_{fsmb} + M_3(x_{p\alpha smb} - 1), \forall p \in P, \forall s \in S, \forall f \in F, \forall m \in W_s, \\ \forall b \in B \end{aligned} \quad (18)$$

$$\begin{aligned} \delta_{fsmb} &\geq A_{fsmb} + t_{fs} + M_3(y_{fsmb} - 1), \forall f \in F \cup \{0\}, \forall s \in S, \forall m \in W_s, \\ \forall b \in B \cup \{0\} \end{aligned} \quad (19)$$

$$\begin{aligned} A_{gsm,b+1} &\geq \delta_{fsmb} + \beta_{fbg,b+1}^{sm} \tau_{fg} - M_3(1 - \beta_{fbg,b+1}^{sm}), \forall f \in F \cup \{0\}, \forall g \in F, \\ \forall s \in S, \forall m \in W_s, \forall b \in (B \cup \{0\}) \setminus \{|B|\} \end{aligned} \quad (20)$$

$$\begin{aligned}
A_{g2m,b+1} &\geq C_{p\alpha1kj} + \beta_{fbg,b+1}^{2m} \tau_{fg} - M_3(2 - \beta_{fbg,b+1}^{2m} - z_{p\alpha kjmb}), \forall p \in P, \\
\forall \alpha \in N, \forall k \in W_1, \forall m \in W_2, \forall f \in F \cup \{0\}, \forall g \in F, \forall j \in B, \\
\forall b \in (B \cup \{0\}) \setminus \{B\}
\end{aligned} \tag{21}$$

$$\begin{aligned}
x_{p\alpha smb}, y_{fsmb}, z_{pkjmb}, u_{p\alpha}, \beta_{fbg,b+1}^{sm} &\in \{0,1\}, \forall p \in P, \forall \alpha \in N, \\
\forall f, g \in F \cup \{0\}, \forall s \in S, \forall k, m \in W_s, \forall j, b \in B \cup \{0\}
\end{aligned} \tag{22}$$

$$\begin{aligned}
n_{p\alpha smb}, \delta_{fsmb}, A_{fsmb}, C_{p\alpha smb} &\geq 0, \forall p \in P, \forall \alpha \in N, \forall f \in F \cup \{0\}, \\
\forall s \in S, \forall m \in W_s, \forall b \in B
\end{aligned} \tag{23}$$

The objective function (1) seeks to minimize the sum of total weighted completion times for product sublots. Constraint set (2) ensures that a batch position can only start to produce a subplot after the subplot is assigned to that same batch position (i.e., the production size of any batch position is equal to 0 if the batch position is not used). In addition, batched quantities must be larger than 0 (3) and smaller than machine capacity in any stage (4). Constraint set (5) ensures that all customer demands are assigned to the first stage. Constraint set (6) forces that dummy product family 0 only can be assigned to dummy batch position 0 in any stage. Constraint sets (7) and (8) are valid inequalities that forces batch positions to be used in sequence.

Next, constraint set (9) ensures that any product subplot with product family f cannot be assigned to a batch position if the product family is not assigned to the same batch position. Constraint set (10) assigns product family sequences between two sequential batch positions. Constraint set (11) is used for assigning values to $z_{p\alpha kjmb}$.

Constraint set (12) indicates which sublots of product p are produced in stage 1. Constraint sets (13) and (14) ensure that if a subplot is produced in stage 1 then it must be assigned to a batch position on some machine in stage 2. Constraint sets (15)-(17)

collectively require that the size of a subplot of product p that is processed in stage 1 and stage 2 should be consistent (i.e., the size of a subplot of product p processed in stage 2 should be equal to the size of a subplot of product p processed in stage 1). However, two sublots of a product can be processed in one batch position so that the actual subplot size of product p on machine m at stage s is determined by $\sum_{\alpha \in N} n_{p\alpha smb}$. Furthermore, constraint sets (15)-(17) guarantee that products produced in stage 2 also satisfy customer demand.

Constraint set (18) restricts the completion time of product subplot α processed by the b^{th} batch position on machine m in stage s to be equal to the completion time of the batch position which is used for manufacturing product family f . Constraint set (19) requires that the completion time of a batch position on any machine at stage s is equal to its starting time plus the associated product family's processing time at stage s . Constraint sets (20) and (21) ensure the setup for a subplot on a machine cannot be started until the subplot arrives at that machine. Constraint set (20) ensures that the overlapping of processing sublots on the same machine is prevented. Sublots processed in batch position $b+1$ on machine m in stage s are allowed to start only after sublots assigned to batch position b on machine m in stage s have been completed. Constraint set (21) prevents overlapping sublots in consecutive stages. Sublots can only start to be manufactured in stage 2 after their completion in stage 1. Finally, constraint sets (22) and (23) are integrality and non-negativity constraints, respectively.

4.2 Establishing Values for Big M Parameters

Three positive large numbers are used in the disjunctive scheduling constraints: M_1 , M_2 , and M_3 . Appropriately establishing “tight” values for these parameters can help to improve model tractability. The value of M_1 (shown in equations (24) - (26)) is limited by constraint sets (2), (15), and (17). Equation (24) determines the value of M_1 that is greater or equal to the subplot size. The maximum subplot size is bounded by $\min(d_p, K_s)$. Therefore, the value of M_1 is $\min(d_p, K_s)$.

$$M_1 \geq \frac{n_{p\alpha smb}}{x_{p\alpha smb}} \quad (24)$$

$$M_1 \geq \frac{n_{p\alpha 1mb} - \sum_{m \in W_2} \sum_{b \in B} n_{p\alpha 2mb}}{\sum_{m \in W_2} \sum_{b \in B} x_{p\alpha 2mb} - \sum_{m \in W_2} \sum_{b \in B} z_{p\alpha kjmb}} \quad (25)$$

$$M_1 \geq \frac{n_{p\alpha 1kj}}{\sum_{m \in W_2} \sum_{b \in B} z_{p\alpha kjmb}} \quad (26)$$

Next, parameter M_2 is used in sublots assignment constraint sets (13) and (14) which are binding for $u_{p\alpha}$ is equal to 1. Then when $u_{p\alpha}$ is equal to 1, the value of M_2 can be written as:

$$M_2 \geq \sum_{m \in W_2} \sum_{b \in B} x_{p\alpha 2mb} - \sum_{k \in W_1} \sum_{j \in B} x_{p\alpha 1kj} \quad (27)$$

Furthermore, equation (27) can be relaxed as:

$$M_2 \geq \sum_{m \in W_2} \sum_{b \in B} x_{p\alpha 2mb} \quad (28)$$

The term $\sum_{m \in W_2} \sum_{b \in B} x_{p\alpha 2mb}$ is the summation of all the batches on all machines at stage 2.

Therefore, the value of M_2 is greater or equal than $|W_2| \times |B|$.

Finally, as it can be transformed from time related constraint sets (18) - (19), M_3 can be expressed as equations (29) - (32). To ensure M_3 is effective, it should be greater or equal than the upper bound of makespan $\delta_{f 2mb}$ (or $C_{p\alpha 2mb}$).

$$M_3 \geq \frac{\delta_{fsmb} - C_{p\alpha smb}}{1 - x_{p\alpha smb}} \quad (29)$$

$$M_3 \geq \frac{A_{fsmb} + t_{fs} - \delta_{fsmb}}{1 - y_{fsmb}} \quad (30)$$

$$M_3 \geq \frac{\delta_{fsmb} + \beta_{fbgb+1}^{sm} \tau_{fg} - A_{gsm,b+1}}{1 - \beta_{fbgb+1}^{sm}} \quad (31)$$

$$M_3 \geq \frac{C_{p\alpha 1kj} + \beta_{fbgb+1}^{2m} \tau_{fg} - A_{2mb+1}}{2 - \beta_{fbgb+1}^{2m} - z_{p\alpha kjmb}} \quad (32)$$

CHAPTER FIVE

NUMERICAL EXPERIMENTS

5.1 Model Validation

In the first example, we consider an instance taken from Biskup and Feldmann (2006): a lot streaming problem with variable sublots and no setups in a flow shop. A single product with a demand of 30 units is to be scheduled by five machines in a flow shop. The product is forced to be split into three sublots. To obtain an appropriate solution, some constraints and variables in our model need to be varied to accommodate the objective function and some problem assumptions. Since the objective is to minimize the makespan, a new variable C_{max} is introduced for makespan. The processing time of a product in the instance is defined as the processing time per unit of product on machine m . Therefore, parameter t_{fs} in our model is changed to t_s .

In the modified model, stage S is a set of machines in a flow shop such that index $s = 1, 2, \dots, |S|$. Machine index m and product family index f are removed from the variables in all the constraints. For example, y_{sb} is a binary variable equal to 1 if the b^{th} batch position (processes product family f) on machine s is used, 0 otherwise. The dummy batch 0 is unnecessary for the new model so that now, the batch position index starts at 1. The machine capacity is set to a large positive number in order to remove its effect as the reference instance is uncapacitated.

The objective function in constraint set (1) is changed to minimize the makespan:

$$Z = C_{max} \quad (33)$$

Constraint sets (2) - (5) hold for the instance. Constraint set (6) for assigning the dummy family 0 to dummy batch 0 is removed. Constraint set (7) is discarded and constraint set (8) is kept in constraint sets (7) and (8) to require batch positions to be used in sequence:

$$y_{sb} - y_{s,b+1} \geq 0, \forall s \in S, \forall b \in B \setminus \{|B|\}. \quad (34)$$

Further, constraint sets (9) and (10) are removed. Since binary variable $z_{pasj,s+1,b}$ indicates a subplot inheritance relationship between batches in all successive stages, constraint sets (11) is modified as follows:

$$x_{pasj} + x_{pa,s+1,b} - z_{pasj,s+1,b} \leq 1, \forall p \in P, \forall \alpha \in N, \forall s \in S \setminus \{|S|\}, \forall j, b \in B \quad (35)$$

Constraint set (12) holds for indicating which sublots of product p are produced on machine 1. Constraint sets (13) to (14) are modified to ensure that if a subplot is produced on machine s then it must be assigned to a batch position on successive machine $s+1$:

$$\sum_{j \in B} x_{pa,s+1,j} \geq \sum_{b \in B} x_{pasb} - M_2(1 - u_{pa}), \forall p \in P, \forall \alpha \in N, \forall s \in S \setminus \{|S|\} \quad (36)$$

$$\sum_{j \in B} x_{pa,s+1,j} \leq \sum_{b \in B} x_{pasb} + M_2 u_{pa}, \forall p \in P, \forall \alpha \in N, \forall s \in S \setminus \{|S|\} \quad (37)$$

Constraint sets (15) to (17) need to be extended to guarantee that products produced at each stage satisfy customer demands:

$$n_{pasj} \geq \sum_{b \in B} n_{pa,s+1,b} + M_1 \left(\sum_{b \in B} z_{pasj,s+1,2b} - \sum_{b \in B} x_{pa,s+1,b} \right), \forall p \in P, \forall \alpha \in N, \forall s \in S \setminus \{|S|\}, \quad (38)$$

$$\forall j, b \in B$$

$$n_{pasj} \leq \sum_{b \in B} n_{pa,s+1,b}, \forall p \in P, \forall \alpha \in N, \forall s \in S \setminus \{|S|\}, \forall j, b \in B \quad (39)$$

$$n_{p\alpha sj} \leq M_1 \sum_{b \in B} z_{p\alpha sj, s+1, b}, \forall p \in P, \forall \alpha \in N, \forall s \in S \setminus \{|S|\}, \forall j, b \in B \quad (40)$$

Constraint sets (18) hold for representing the non-linear equation $C_{pasb} = \delta_{sb}(x_{pasb})$ such that the completion time of a product subplot α processed by the b^{th} batch position on machine s equals the completion time of that batch position. Constraint sets (19) substitute for:

$$\delta_{sb} \geq A_{sb} + t_s \sum_{p \in P} \sum_{\alpha \in N} n_{p\alpha sb} + M(y_{sb} - 1), \forall p \in P, \forall \alpha \in N, \forall s \in S, \forall b \in B_m \cup \{0\} \quad (41)$$

The completion time of a batch position on any machine s is equal to its starting time plus the processing time of all sublots in that batch. The processing time is directly proportional to batch size. Therefore, constraint set (20) and (21) are simplified as follows:

$$A_{sb} \geq \delta_{s, b-1}, \forall s \in S, \forall b \in B \setminus \{1\} \quad (42)$$

$$A_{s, b} \geq C_{p\alpha, s-1, j} - M_3(1 - z_{p\alpha, s-1, j, s, b}), \forall p \in P, \forall \alpha \in N, \forall s \in S \setminus \{1\}, \forall j, b \in B \quad (43)$$

To prevent overlapping sublots in successive batch positions on consecutive machines, batch position b on machine s should be started after completion of the preceding batch position $b-1$ on the same machine s as well as after the completion of batch position j , which is used for processing the same subplot on the preceding machine $s-1$. Finally, new constraint set (44) is added to define the makespan as the completion time of the last batch on the last machine:

$$C_{\max} \geq C_{p\alpha|s|b}, p \in P, \alpha \in N, s \in S, b \in B \quad (44)$$

The model is coded in AMPL and solved using Gurobi 6.0 on a Core i7, 3.40 GHz CPU with 8 GB of RAM. The optimal objective value (269.8 minutes) of this sample instance is successfully found by our modified model.

5.2 Model Demonstration

The proposed MIP model is analyzed under various problem settings based on representative case study input data. In all three example problems that follow, there are three orders wherein each order contains one product. The maximum number of sublots $|N|$ and the maximum number of batch positions $|B|$ in all examples are 3 and 4, respectively. The data for machine configuration and product information are specified in Table 5.1. Table 5.2 shows the processing time for all stages, while sequence-dependent setup times for all examples are given in Table 5.3.

Table 5.1 Data for Example Problems

Example	(Stage, No. of Machines, Capacity)	(Product, Product Family, Demand, Weight)
2	(1, 1, 4)	(1, 1, 2, 1)
	(2, 2, 2)	(2, 2, 3, 2)
		(3, 1, 5, 3)
3	(1, 2, 4)	(1, 1, 2, 1)
	(2, 2, 2)	(2, 2, 3, 2)
		(3, 1, 5, 3)

Table 5.2 Processing Times for Example Problems

Product Family	Processing Time in Stage 1	Processing Time in Stage 2
1	2	1
2	2	1
3	2	1

Table 5.3 Setup Times for Each Product Family

Product Family	Product Family	
	F1	F2
F1	0	3
F2	1	0

Example 2, which consists two problems, is designed to show the effect of applying lot streaming with batching. Example 2(a) allows batching and lot streaming simultaneously, while only lot streaming is considered in Example 2(b). The parameter settings in the two problems are the same. The resulting Gantt charts produced by the MIP model for Example 2(a) and (b) are shown in Figure 5.1. An analysis of Example 2(a) reveals that the sum of total weighted completion times for all products is 84 (Figure 5.1(a)). Product 3 (product 2) is split into 2 (1) sublots in stage 1, and 3 (2) sublots in stage 2. Since products 1 and 3 belong to the same product family, one subplot of product 1 and one subplot of product 3 are processed in the same batch position (2^{th}) on stage 1's single machine simultaneously. Sublot sizes vary across the two stages, given the relationship between product 3's demand of 5 and the capacity per machine in each stage. The objective value of Example 2(b) is 87. It is clear that scheduling using integrated batching and lot streaming is better than only using lot streaming alone.

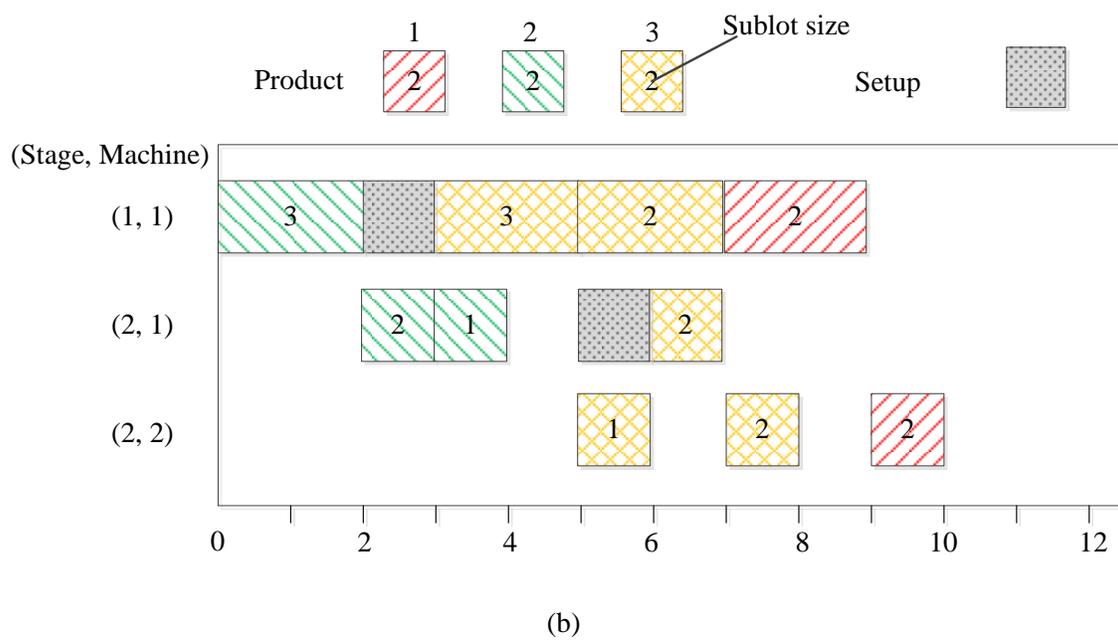
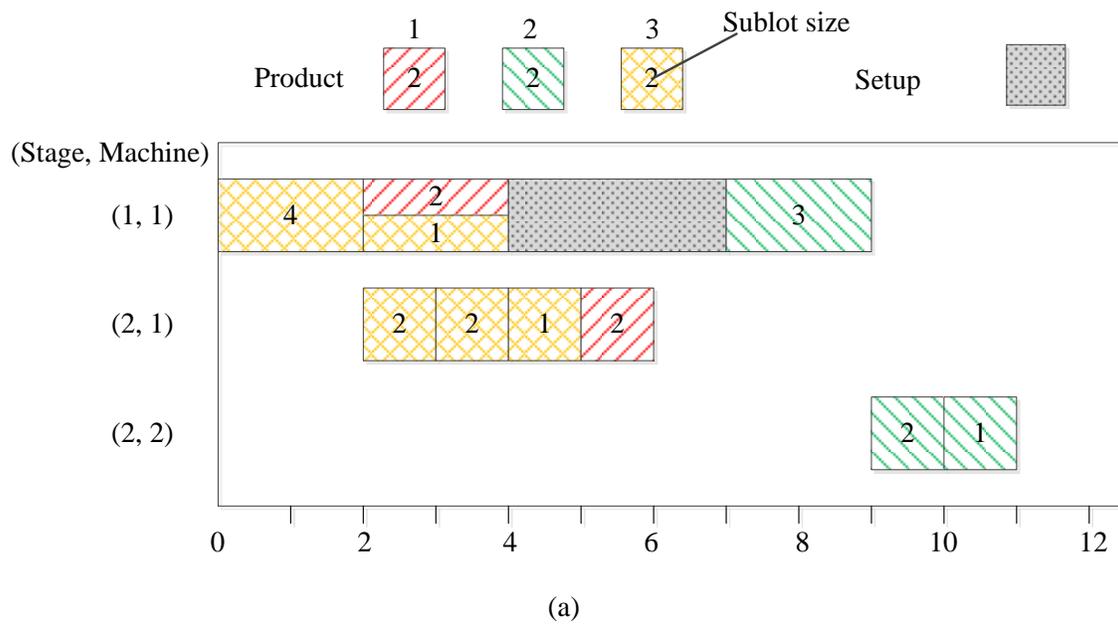


Figure 5.1 Optimal Solution of Example 2
 (a) Integrated Batching with Lot Streaming (b) Lot Streaming

In another example instance, the machine configuration is changed from the previous examples: there are two machines at each flow shop stage. The optimal solution depicted in Figure 5.2 confirms that a subplot of product 3 and a subplot of product 1 are manufactured in the 2nd batch position on machine 1 in stage 1, as expected, thereby again validating the model's functionality for a two-stage hybrid flow shop with multiple machines at each stage. As an additional machine is added at stage 1, the sum of total weighted completion times of Example 4 is 56, 28 units shorter than the corresponding result in Example 1.

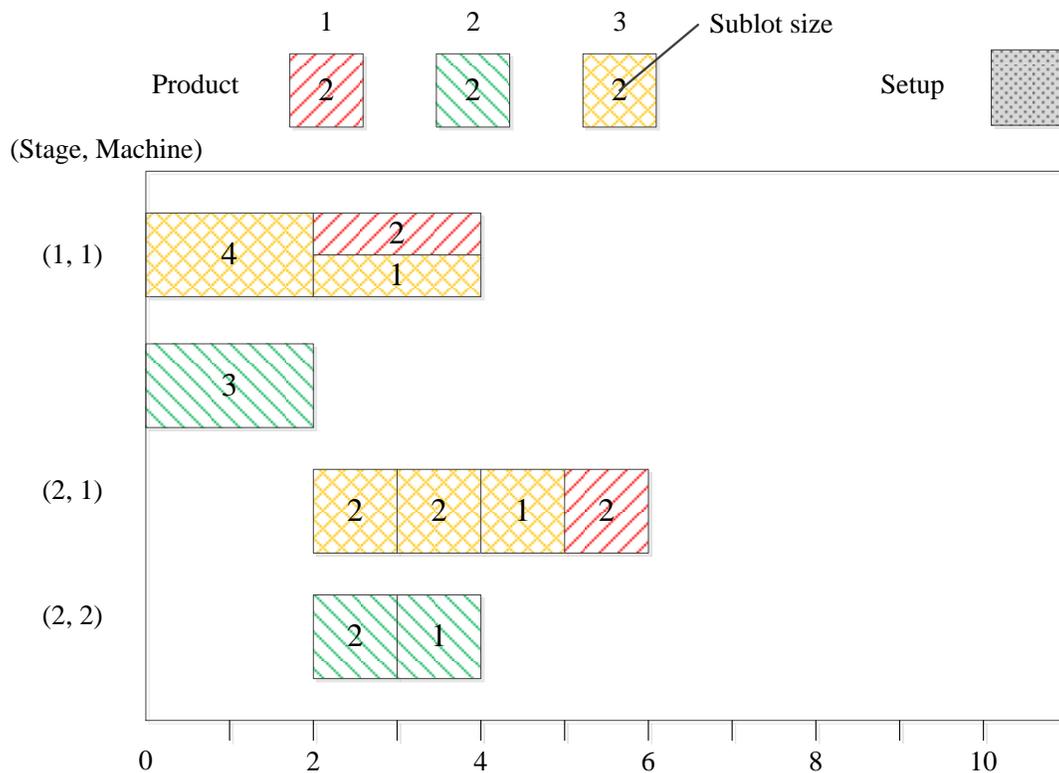


Figure 5.2 Optimal Solution of Example 3

5.3 Experimental Study

In order to evaluate the performance of the proposed MIP model, 60 experimental problem instances are analyzed using the experimental design in Table 5.4. The machine environment setting for all 60 instances is a two-stage flow shop: one capacitated machine operated at stage 1 having a capacity of 7.2 units; three identical parallel vessels each with capacity of 4 comprise stage 2. A product is allowed to be split into at most eight sublots. The maximum value for the number of batches is set to 15.

The weights (priorities) of the products are the same if they are from the same order: random integers between 1 and 3 (i.e., $w_p \sim DU [1, 3]$). Table 5.5 provides the number of orders and the number of products in each order for all 60 instances. In instances 1 – 20, one order is considered in each instance, and in instances 21 – 60, two orders were analyzed in each instance. As shown in Table 5.5, two or three products are studied in each order. Each order is for painting cargo containers, ship hulls, or industrial structures with probability 0.05, 0.25, and 0.7, respectively. The demand for primer in cargo container orders is randomly generated using the uniform distribution $U [12, 25]$. Similarly, demand for primer in a ship hull order and an industrial structure order are created according to uniform distribution $U [12, 45]$ and $DU [7, 35]$, respectively. The demands for top coat paint and undercoat paint are 50% of and 20% of the corresponding primer quantity in the order, respectively. We assume that primer, top coat paint, and undercoat paint belong to three different, incompatible job families. The processing time of each product family at each stage (Table 5.6) and the setup time between each product family (Table 5.7) are fixed in all 60 instances.

Table 5.4 Experimental Design for Model Evaluation

Parameter	Value Description
Order Type	Cargo container with probability of 0.05 Ship hull with probability of 0.25 Industrial structure with probability of 0.7
Weight (Priority)	$DU [1, 2]$
Primer Demand	Cargo container: $U [12, 25]$ Ship hull: $U [12, 45]$ Industrial structure: $U [7, 35]$

Table 5.5 Number of Orders and Number of Products in each Order

Instances	Number of Orders	Number of Products in Each Order
1-10	1	2
11-20	1	3
21-30	2	(Order 1, 2), (Order 2, 2)
31-40	2	(Order 1, 2), (Order 2, 3)
41-50	2	(Order 1, 3), (Order 2, 2)
51-60	2	(Order 1, 3), (Order 2, 3)

Table 5.6 Processing Times for Product Families

Product Family	Processing Time in Stage 1	Processing Time in Stage 2
1	2.14	3.36
2	2.73	1.52
3	1.67	2.44

Table 5.7 Setup Times for Product Families

Product Family	Product Family		
	F1	F2	F3
F1	0	0.5	0.7
F2	0.3	0	1
F3	0.8	0.8	0

The problem instances are coded in AMPL and solved in Gurobi 6.0 on a Core i7 3.40 GHz CPU with 8 GB of memory. Each problem is allowed to run for a maximum of 7,200 seconds of CPU time (two hours). All 60 problems were stopped due to the time limit being reached before finding an optimal solution. Table 5.8 summarizes the information from the Gurobi solutions. As the experiment results revealed, the optimality gap increases as the number of products increases. The average optimality gap of instances 1 – 10 is 87.2% wherein when two products are considered in each order. Unfortunately, if more than three total products are involved (instances 21 – 60), the optimality gap never reduces below 100%, even after a fairly lengthy amount of computation time.

This high computation cost is additional evidence that our problem’s complexity is most probably NP hard. The problems in the experimental study are small in comparison to actual problems address in practice. This resulting computational performance suggests the need (as expected) for the development of efficient heuristics to analyze both this experimental study set and large, more practical-sized industrial problems.

Table 5.8 Optimality Gaps for 60 Instances

Problem	Optimality Gap (%)	Problem	Optimality Gap (%)	Problem	Optimality Gap (%)
1	92.8	21	100	41	100
2	92.9	22	100	42	100
3	92.4	23	100	43	100
4	87.6	24	100	44	100
5	84.5	25	100	45	100
6	77.6	26	100	46	100
7	85.1	27	100	47	100
8	91.6	28	100	48	100
9	84.7	29	100	49	100
10	82.8	30	100	50	100
11	100	31	100	51	100
12	100	32	100	52	100
13	100	33	100	53	100
14	100	34	100	54	100
15	100	35	100	55	100
16	100	36	100	56	100
17	100	37	100	57	100
18	100	38	100	58	100
19	100	39	100	59	100
20	100	40	100	60	100

CHAPTER SIX

CONCLUSIONS AND FUTURE RESEARCH

Both batching and lot streaming are well studied in the past 20 years of literature. However, few studies investigate integrated batching and lot streaming simultaneously. This research investigates the integrated batching and lot streaming problem with variable sublots, incompatible job families, and sequence-dependent setup in a two-stage hybrid flow shop. This research is motivated by the author's work experience at a coating company in China. A MIP model is presented for this problem wherein the number of sublots for each product, the size of each subplot, and the production sequence for each subplot are determined simultaneously to minimize the sum of total weighted completion times.

Three example problems are tested to validate the proposed model. One set of examples illustrate that applying integrated batching and lot streaming can lead to improvements in the sum of total weighted completion times for product sublots as compared to considering lot streaming alone. The model is implemented in a two-stage hybrid flow shop with multiple machines at each stage. In addition, the experimental test results show that the optimality gap changes in the same direction as the number of the products varies. Besides, when considering more than three products, the optimality gap reaches up to 100%. However, the computation cost for solving this optimization model is too high for practical implementation.

The high computation cost may be explained as the proposed problem is most probably an NP-hard problem. Hence, the proof of our problem's complexity is an

interesting topic for further study. Furthermore, as the MIP model takes a large amount of computation time, the development of heuristic approaches for solving large-size problems is a necessary extension to this work. In addition, the model could be extended to deal with the same problem in a multi-stage hybrid flow shop in the future. For a future journal article submission of this research, some of these research extensions clearly must be undertaken.

REFERENCES

- Amorim, P., Antunes, C. H., & Almada-Lobo, B. (2011). Multi-objective lot-sizing and scheduling dealing with perishability issues. *Industrial & Engineering Chemistry Research*, 50(6), 3371-3381.
- Behnamian, J., Ghomi, S. F., Jolai, F., & Amirtaheri, O. (2012). Minimizing makespan on a three-machine flowshop batch scheduling problem with transportation using genetic algorithm. *Applied Soft Computing*, 12(2), 768-777.
- Biskup, D., & Feldmann, M. (2006). Lot streaming with variable sublots: an integer programming formulation. *Journal of the Operational Research Society*, 57(3), 296-303.
- Burtseva, L., González-Navarro, F. F., Perez, P. F., Romero, R., Ramirez, S., & Yaurima, V. (2012). *Lot processing in Hybrid Flow Shop scheduling problem*. INTECH Open Access Publisher.
- Cakici, E., Mason, S. J., Fowler, J. W., & Geismar, H. N. (2013). Batch scheduling on parallel machines with dynamic job arrivals and incompatible job families. *International Journal of Production Research*, 51(8), 2462-2477.
- Chang, J. H., & Chiu, H. N. (2005). A comprehensive review of lot streaming. *International Journal of Production Research*, 43(8), 1515-1536.
- Cheng, M., Mukherjee, N. J., & Sarin, S. C. (2013). A review of lot streaming. *International Journal of Production Research*, 51(23-24), 7023-7046.
- Chiu, H. N., Chang, J. H., & Lee, C. H. (2004). Lot streaming models with a limited number of capacitated transporters in multistage batch production systems. *Computers & Operations Research*, 31(12), 2003-2020.
- Defersha, F. M., & Chen, M. (2010). A hybrid genetic algorithm for flowshop lot streaming with setups and variable sublots. *International Journal of Production Research*, 48(6), 1705-1726.
- Defersha, F. M., & Chen, M. (2012a). Mathematical model and parallel genetic algorithm for hybrid flexible flowshop lot streaming problem. *The International Journal of Advanced Manufacturing Technology*, 62(1-4), 249-265.
- Defersha, F. M., & Chen, M. (2012b). Jobshop lot streaming with routing flexibility, sequence-dependent setups, machine release dates and lag time. *International Journal of Production Research*, 50(8), 2331-2352.

- Erramilli, V., & Mason, S. J. (2006). Multiple orders per job compatible batch scheduling. *Electronics Packaging Manufacturing, IEEE Transactions on*, 29(4), 285-296.
- Erramilli, V., & Mason, S. J. (2008). Multiple orders per job batch scheduling with incompatible jobs. *Annals of Operations Research*, 159(1), 245-260.
- Feldmann, M., & Biskup, D. (2008). Lot streaming in a multiple product permutation flow shop with intermingling. *International Journal of Production Research*, 46(1), 197-216.
- Fu, Q., Sivakumar, A. I., & Li, K. (2012). Optimisation of flow-shop scheduling with batch processor and limited buffer. *International Journal of Production Research*, 50(8), 2267-2285.
- Fumero, Y., Corsano, G., & Montagna, J. M. (2014). Simultaneous batching and scheduling of batch plants that operate in a campaign-mode, considering nonidentical parallel units and sequence-dependent changeovers. *Industrial & Engineering Chemistry Research*, 53(44), 17059-17074.
- Graham, R. L., Lawler, E. L., Lenstra, J. K., & Kan, A. R. (1979). Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of discrete mathematics*, 5, 287-326.
- Ghasemi, M. (2008). Lot streaming in hybrid flow shop scheduling, MS Thesis, Concordia University.
- Goyal, S. K., & Szendrovits, A. Z. (1986). A constant lot size model with equal and unequal sized batch shipments between production stages. *Engineering Costs and Production Economics*, 10(3), 203-210.
- Gupta, J. N. (1988). Two-stage, hybrid flowshop scheduling problem. *Journal of the Operational Research Society*, 359-364.
- Huang, T. C., & Lin, B. M. (2013). Batch scheduling in differentiation flow shops for makespan minimisation. *International Journal of Production Research*, 51(17), 5073-5082.
- Kalir, A. A., & Sarin, S. C. (2001). A near-optimal heuristic for the sequencing problem in multiple-batch flow-shops with small equal sublots. *Omega*, 29(6), 577-584.
- Kopanos, G. M., Puigjaner, L., & Georgiadis, M. C. (2011). Production scheduling in multiproduct multistage semicontinuous food processes. *Industrial & Engineering Chemistry Research*, 50(10), 6316-6324.

- Kurz, M. E., & Askin, R. G. (2004). Scheduling flexible flow lines with sequence-dependent setup times. *European Journal of Operational Research*, 159(1), 66-82.
- Liu, S. C. (2003). A heuristic method for discrete lot streaming with variable sublots in a flow shop. *The International Journal of Advanced Manufacturing Technology*, 22(9-10), 662-668.
- Logendran, R., & Barnard, F. (2006). Sequence-dependent group scheduling problems in flexible flow shops. *International Journal of Production Economics*, 102(1), 66-86.
- Martin, C. H. (2009). A hybrid genetic algorithm/mathematical programming approach to the multi-family flowshop scheduling problem with lot streaming. *Omega*, 37(1), 126-137.
- Morakotkarn, W. (1995). Two-stage scheduling with lot-streaming, MS Thesis, Texas Tech University.
- Mortezaei, N., & Zulkifli, N. (2014). Mixed-integer Formulation for Integration of Lot Sizing and Lot Streaming Problem with Scheduled Preventive Maintenance. *Research Journal of Applied Sciences, Engineering and Technology*, 7(12), 2563-2568.
- Pan, Q. K., & Ruiz, R. (2012). An estimation of distribution algorithm for lot-streaming flow shop problems with setup times. *Omega*, 40(2), 166-180.
- Pan, Q. K., Tasgetiren, M. F., Suganthan, P. N., & Chua, T. J. (2011). A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. *Information sciences*, 181(12), 2455-2468.
- Potts, C. N., & Baker, K. R. (1989). Flow shop scheduling with lot streaming. *Operations research letters*, 8(6), 297-303.
- Potts, C. N., & Kovalyov, M. Y. (2000). Scheduling with batching: a review. *European journal of operational research*, 120(2), 228-249.
- Pranzo, M. (2004). Batch scheduling in a two-machine flow shop with limited buffer and sequence independent setup times and removal times. *European Journal of Operational Research*, 153(3), 581-592.
- Reiter, S. (1966). A system for managing job-shop production. *Journal of Business*, 39(3), 371-393.

- Sarin, S. C., & Jaiprakash, P. (2007). *Flow shop lot streaming*. New York: Springer Science & Business Media.
- Schultmann, F., Fröhling, M., & Rentz, O. (2006). Fuzzy approach for production planning and detailed scheduling in paints manufacturing. *International journal of production research*, 44(8), 1589-1612.
- Tai, Y. T., & Lai, C. M. (2011). The liquid crystal injection scheduling problem (LCISP). *International Journal of Production Research*, 49(2), 467-484.
- Trietsch, D., & Baker, K. R. (1993). Basic techniques for lot streaming. *Operations Research*, 41(6), 1065-1076.
- Tseng, C. T., & Liao, C. J. (2008). A discrete particle swarm optimization for lot-streaming flowshop scheduling problem. *European Journal of Operational Research*, 191(2), 360-373.
- Yalaoui, F., & Chu, C. (2006). New exact method to solve the $Pm/r_j/\sum C_j$ schedule problem. *International Journal of Production Economics*, 100(1), 168-179.
- Zhang, W., Yin, C., Liu, J., & Linn, R. J. (2005). Multi-job lot streaming to minimize the mean completion time in m-1 hybrid flowshops. *International Journal of Production Economics*, 96(2), 189-200.