Production and Transportation Integration for a Make-to-Order Manufacturing Company with a Commit-to-Delivery Business Mode

Kathryn E. Stecke
and
Xuying Zhao

The University of Texas at Dallas,
School of Management,
Richardson, TX

kstecke@utdallas.edu
and
xuying.zhao@student.utdallas.edu

December, 2004
Abstract

When a make-to-order manufacturing company adopts a commit-to-delivery mode, it commits a delivery due date for an order and is responsible for the shipping cost. Without loss of generality, we consider that transportation is done by a third party logistics company such as FedEx or UPS, which provides multiple shipping modes such as overnight, one-day, two-day delivery, and more. When the shipping time has to be short, clearly shipping cost is more expensive than it could have been. How should a company schedule production for all accepted orders so that the company can leave enough shipping time for orders to take slow shipping modes to reduce the shipping cost? We study this problem of integrating the production and transportation functions for a manufacturing company producing a variety of products in a make-to-order environment with a commit-to-delivery mode of business. Because of the nature of the problem, it is sufficient to model the production function as a capacity-constrained single machine.

A portfolio of eight distribution scenarios are considered: (a) two shipping cost functions, linear or nonlinear shipping cost functions (of shipping time); (b) two modes of deliveries, partial delivery allowed or not allowed; and (c) two ways to deal with customer location, considering or not considering customer location in shipping cost. We first focus on the four scenarios where shipping cost does not account for customer location. In the two scenarios where partial delivery is allowed, we provide LP models and show that nonpreemptive EDD production schedules are optimal except for when a shipping cost function is a nonlinear and concave function of shipping time. In the two scenarios when partial delivery is not allowed, we develop MIP models and prove that the problem is NP-hard. For the scenario with linear shipping cost, we show that the worst case bound for any feasible solution to the NP-hard problem is less than two and close to one when the production planning horizon is long. This result is unusual for an NP-hard problem. To our knowledge, the only other NP-hard problem having a similar worst case bound (approximation ratio) is the famous edge coloring problem. An efficient heuristic algorithm with polynomial computation time is provided for the NP-hard problem. It gives near-optimal production schedules, as shown via many numerical experiments. We also provide models and analysis for the scenarios where shipping cost accounts for customer location.

Key Words and Phrases: integration, make-to-order, commit-to-deliver, transportation, production scheduling.
1 Introduction

Some companies produce products and stock them as inventory until they are sold (make-to-inventory). Other firms produce products only after they are ordered (make-to-order). This can reduce inventory and can allow customization. However, the make-to-order mode also has its disadvantages. Customers have to wait for the orders to be manufactured and shipped. Orders may miss their due dates because of production and lead time variability, or insufficient production capacity.

In a make-to-order environment, it is common that the transportation is done by a third party logistics company such as FedEx or UPS, which provides multiple shipping modes such as overnight, one-day, two-day delivery, etcetera. A shipping date is the date when the manufacturing company gives products to a third party logistics company. A delivery date is the date when the customer receives the products.

Some companies in a make-to-order environment, such as Dell, Ebay, and Amazon, which use a commit-to-ship business mode, promise customers to ship products on or before committed shipping dates. In a commit-to-ship mode, customers choose a shipping mode and pay the shipping cost when they place an order.

Many companies in a make-to-order environment use a commit-to-delivery business mode by promising to deliver orders on or before committed delivery dates at the customers’ door steps. Companies such as Cemex, a Mexican cement producer with $6.5 billion revenue per year and MEI, an electronics company with $14 billion consolidated sales per year, have adopted the commit-to-delivery business mode. In commit-to-delivery, companies select a shipping mode dynamically according to the production finishing time for each order and pay the shipping cost.

Among the two modes, commit-to-delivery could reduce the lead time variability and then improve customer satisfaction. This is because if an order is produced late, the company would select a faster shipping mode for the order to increase the chance that it arrives on or before the committed delivery date.

A firm in a make-to-order environment may improve profits from the commit-to-delivery business mode as customers might be willing to pay more for a guaranteed delivery date instead of just a shipping date. In addition, if an order is finished early, a slow shipping mode could be applied. Consequently, the shipping cost can be reduced and thus the profit can be improved.

It is common for customers to receive compensation when their orders miss due dates. In commit-to-delivery, a customer could get compensation from the firm even if the delay is not caused by manufacturing but by shipping. The firm will determine whether or not the logistics company should share the penalty cost incurred from such compensation. However, in commit-to-ship, the customer has to deal with both
the firm and the logistics company in case of delayed service.

For the reasons mentioned above, commit-to-delivery could be better than commit-to-ship for both customers and companies. Some companies such as Dell are considering providing commit-to-delivery service to customers with additional premium payment (Killian (2003)).

The integration of production and transportation is very important when adopting a commit-to-delivery mode in practice. In order to avoid penalty costs, a firm desires that all orders meet their due dates. Poor production schedules may cause manufacturing delays. Consequently, the firm may have to expedite shipping. The shipping cost may then increase which in turn may reduce profit.

In this paper, we address the integration of production and transportation for a manufacturing company producing a single product in a make-to-order environment with a commit-to-delivery mode, where companies are responsible for shipping cost and commit a delivery date when a customer places an order. In order to improve profit, the company wants to reduce the total shipping cost paid for all fulfilled orders by selecting a slow shipping mode for each order while meeting its due date. We show that we can take advantage of the slow shipping modes and then reduce the shipping cost to some extent through adjusting the production schedule.

The motivation for this research comes from a project that we have completed at Dell. As mentioned in a report by Best Practice LLC, Dell Inc. successfully reduced its inventory to only a five-day supply. 95% of customer orders are shipped within eight hours. This is much earlier than a committed shipping date, which on average is five days after the order is placed. Short manufacturing lead time (eight hours for 95% of customer orders) helps Dell to lower inventory, which is a key competitive advantage as a low-cost producer in the computer industry. Dell can reap the benefits of lower inventory costs almost immediately and reflect that benefit in its consumer product prices.

Dell is looking for ways to increase profit while still providing low prices to customers. Commit-to-delivery could help Dell to get extra profit from those 95% of orders which are shipped within eight hours. For example, suppose that a customer places an order and she chooses overnight shipping. Dell commits to ship the computer after five days. Suppose that the customer pays $700 including a $100 overnight shipping fee, which is normal for a 25-pound-computer. In this model, Dell gets $600 and the logistics company gets $100. Now if Dell adopts commit-to-delivery, Dell can adjust the production schedule and try to finish production for the order within eight hours. Then Dell could select a slow ship mode, such as four-day delivery, which may just cost $20. With a commit-to-delivery model, Dell would get $680 and the logistics company would get $20.

In order to adopt commit-to-delivery, Dell needs to answer the following question: What production schedule could help Dell to reduce the total shipping cost as much as possible and also deliver orders on
time? In this paper, we model this problem and answer this question.

Dell assembles computers for customers when the orders are placed. As a result, it suffices to model
the production (assembly) function as a capacitated single machine. Our models are appropriate for any
paced, capacity-constrained production line that produces a variety of products.

A portfolio of eight distribution scenarios are considered: (a) two cost functions, linear or nonlinear
shipping cost; (b) two modes of deliveries, partial delivery allowed or not allowed; and (c) considering or
not considering customer locations in the various shipping cost functions. With a linear (nonlinear)
shipping cost function, shipping cost per item linearly (nonlinearly) decreases with shipping time. When
partial delivery is allowed, a firm could divide a customer’s order into multiple smaller orders and ship
them at different times. This increases the flexibility of the production schedule and may help a firm to
save more on shipping costs. It is common in practice that shipping cost does not account for customer
location as long as customers are inside the US continent. For example, a lot of sellers in Ebay charge
the same shipping cost to all customers inside the US.

We organize the rest of the paper as follows. Section 2 is the literature review. Sections 3 through
6 study the integration problems of the first four scenarios where shipping cost does not account for
customer location. The main results of the paper are as follows.

1. The two scenarios (linear and nonlinear shipping cost) when partial delivery is allowed are modeled
using linear programming in sections 3 and 5, respectively. We also show that nonpreemptive EDD
production schedules are optimal (except in the scenario where a shipping cost function is nonlinear
and concave in section 5.2).

2. The two scenarios where partial delivery is not allowed are studied in sections 4 and 6, where the
MIP models are presented. We also prove that the problem is NP-hard.

3. For the scenario with linear shipping cost and no partial delivery in section 4, we show that the worst
case bound for any feasible solution to the NP-hard problem is $1 + \delta$, where $\delta$ is a small number
in the interval $(0,1)$. When the production planning horizon goes to infinity, $\delta$ goes to zero and
the bound goes to one. This result is unusual for a NP-hard problem. To our knowledge, the only
other NP-hard problem having a similar worst case bound is the edge coloring problem. See Vizing
(1964).

4. An efficient heuristic algorithm is provided for the NP-hard problem in section 4. It gives near-optimal
production schedules, as shown via many numerical experiments in section 6.

In section 7, we extend these models to account for customer location in shipping cost. Section 8
summarizes the paper.
Most of the literature for production scheduling problems for a single machine focuses on minimizing total weighted completion time, maximum lateness, number of tardy jobs, total tardiness, total weighted tardiness, or total earliness. (See Pinedo (2000) for details in single machine production scheduling models.) These problems are well studied by, for example, Anderson and Potts (2004), Baker and Scudder (1990), Balas, Lenstra, and Vazacopoulos (1995), Baptiste (1999), Buxey (1989), Carlier (1982), Chand, Traub, and Uzsoy (1996) and (1997), Chen and Bulfin (1993) and (1994), Du and Leung (1990), Emmons (1969), Graves (1981), Kim and Yano (1994), and Leung and Young (1990). However, we study a production scheduling problem with an objective to minimize the total shipping cost, which is different from the objectives in a traditional production scheduling problem.

The study of the integration between transportation and production has attracted attention in recent years. Chandra and Fisher (1994) investigated the value of coordinating production and distribution. They consider a two-level system involving a single manufacturing plant producing finished goods and routing the delivery of those goods to satisfy the requirements of several retailers. They show that an integrated analysis of the production scheduling and distribution routing problems can yield a 3% to 20% reduction in operational cost.

Bhatnagar, Chandra, and Goyal (1993), Thomas and Griffin (1996), and Sarmiento and Nagi (1999) survey the integrated analysis of the production, inventory, and distribution functions in a supply chain. Sarmiento and Nagi (1999) mention that "the investigation of alternative transportation modes is needed. Most of the models reviewed, consider the transportation time in the distribution system to be fixed.” See Burns et al. (1985), Federgruen and Zipkin (1984), Federgruen, Prastacos, and Zipkin (1986), Anily and Federgruen (1990), Viswanathan and Mathur (1997), and Speranza and Ukovich (1994).

De Matta and Miller (2004) study the benefits and costs of coordinating the production and inter-facility transportation. They developed a dynamic production and transportation decision model to determine the cost-minimizing quantity to produce and ship. They claimed that coordinating production and transportation decisions can "control the use of expedited transport mode to a minimum through timely shipments of sufficient intermediate product quantities via the normal inexpensive mode to meet the input requirements of the finishing plant.”

Chen and Vairaktarakis (2005) discuss the integrated scheduling of production and distribution operations. They provide solution approaches for machine scheduling and transportation routing to minimize a weighted sum of the transportation cost and a factor indicating customer service level, which is measured by either the maximum or average delivery time.

Geismar, Lei, and Srisankarajah (2004) address an integrated production and distribution problem
for products with a short life cycle. They prove that the problem is NP-hard and provide heuristics to
give near-optimal solutions.

To our knowledge, no research to date addresses how to integrate production and transportation for
a commit-to-delivery business mode in a make-to-order environment when delivery service is performed
by a third party logistics company, such as FedEx and UPS. We seek to fill this gap with our paper.

Here, the objective function is to minimize the total shipping cost. Shipping cost is a function
of shipping time, which changes depending on shipping mode. Considering customer satisfaction, we
constrain that the production finishing date for an order has to be earlier than its delivery due date.

3 Linear Shipping Cost Problem when Partial Delivery is Allowed

For ease of presentation, we begin with a linear shipping cost function which does not exist in reality.
This is necessary to build up to the real nonlinear shipping cost.

Before presenting the model, first consider the problem scenario. Suppose that a manufacturing
company operates 24 hours a day and seven days a week. Each day a third party logistics company comes
at the same time, say 3:00 PM, to collect the orders ready for shipping. For convenience, we define one
production day as starting at 3:00 PM and ending at 3:00 PM the next day.

Given \( n \) orders with delivery due dates and the daily production capacity of the manufacturing
company, how should these \( n \) orders be scheduled over the planning horizon of \( m \) production days? All
orders should meet their due dates and the total shipping cost for the \( n \) orders should be as small as
possible.

For modeling convenience, we transfer each delivery due date to a production due date. A produc-
tion due date is the latest production day in the planning horizon that the order can be finished and still
arrive on the delivery due date via overnight shipping. That is, in order to meet the delivery due date
of an order, the manufacturer must finish production for the order on or before its production due date
in the planning horizon. Otherwise, the order will not arrive to the customer in time, even via overnight
shipping.

For example, suppose that the planning horizon starts at 3:00 PM today, as depicted in Figure 1,
and the delivery due date for an order is the day after tomorrow. Then the manufacturer must finish
production for the order on or before the end of the first production day, and ship it when the carrier
comes to collect orders at 3:00 PM tomorrow. With overnight shipping, the order should then arrive
at the customer the day after tomorrow. Therefore, the production due date for this order is the first
production day in the planning horizon, that is, from 3:00 PM today to 3:00 PM tomorrow.

To save shipping cost, a manufacturer ships an order as soon as it is ready to ship at 3:00 PM.
A logistics company comes to collect finished orders.

Therefore, the shipping date of an order is equal to its production finishing date in the planning horizon. In the example above, the order is finished production on the first production day. Therefore, the shipping date of this order is production day one.

The shipping mode for an order can be selected dynamically after the order finishes production. If the production finishing date (shipping date) of an order is the same as its production due date, then overnight shipping has to be used. If the production finishing date is \( k \) days earlier than the production due date, a \( k \)-day shipping mode would be selected, where \( k \) is also the shipping time in days.

Below are the terms that we use frequently in the paper.

**Table 1.** Notation.

\[
\begin{align*}
p & = \text{average production rate (in number of items per hour)} \\
c & = \text{available daily production capacity (in number of items per 24 hours)} \\
Q_i & = \text{number of items required in order } i \\
m & = \text{number of production days in the planning horizon} \\
d_i & = \text{production due date for order } i; \ d_i \leq m \\
t_i & = \text{shipping date for order } i; \ t_i \leq d_i \\
r_i & = \text{shipping mode for order } i; \ r_i = d_i - t_i 
\end{align*}
\]

The shipping modes are overnight shipping \((r_i=0)\), one-day shipping \((r_i=1)\), two-day shipping \((r_i=2)\), and so on. Table 2 is a screenshot of a FedEx shipping cost table, which is found on the FedEx website. Three types of overnight shipping are listed in the table. In this paper, overnight shipping means standard overnight shipping, by which packages will arrive before 3:00 PM the next business day to most cities.

In order to make sure that a feasible schedule exists, in which all orders meet their delivery due dates, we assume that the \( n \) orders satisfy the following feasibility check condition:

\[
\frac{\sum_{i \in A_j} Q_i}{c} \leq j, \quad j = 1, \ldots, m,
\]

where \( A_j = \{i|d_i \leq j\} \) denotes a set of orders having a production due date on or before the production day \( j \) in the planning horizon.
Table 2. Screenshot of a FedEx shipping cost table.

<table>
<thead>
<tr>
<th>Weight</th>
<th>FedEx First Overnight</th>
<th>FedEx Priority Overnight</th>
<th>FedEx Standard Overnight</th>
<th>FedEx 2Day*</th>
<th>FedEx Express Saver*</th>
</tr>
</thead>
<tbody>
<tr>
<td>50 lbs</td>
<td>$40.25</td>
<td>$122.25</td>
<td>$112.00</td>
<td>$51.00</td>
<td>$47.00</td>
</tr>
<tr>
<td>51</td>
<td>$39.50</td>
<td>$119.50</td>
<td>$113.75</td>
<td>$50.80</td>
<td>$47.75</td>
</tr>
<tr>
<td>52</td>
<td>$38.75</td>
<td>$116.75</td>
<td>$115.90</td>
<td>$50.60</td>
<td>$48.55</td>
</tr>
<tr>
<td>53</td>
<td>$38.00</td>
<td>$114.00</td>
<td>$118.15</td>
<td>$50.40</td>
<td>$49.30</td>
</tr>
<tr>
<td>54</td>
<td>$37.25</td>
<td>$111.25</td>
<td>$120.40</td>
<td>$50.20</td>
<td>$50.05</td>
</tr>
<tr>
<td>55</td>
<td>$36.50</td>
<td>$108.50</td>
<td>$122.70</td>
<td>$50.00</td>
<td>$50.80</td>
</tr>
<tr>
<td>56</td>
<td>$35.75</td>
<td>$105.75</td>
<td>$125.00</td>
<td>$49.80</td>
<td>$51.60</td>
</tr>
<tr>
<td>57</td>
<td>$35.00</td>
<td>$103.00</td>
<td>$127.25</td>
<td>$49.60</td>
<td>$52.40</td>
</tr>
<tr>
<td>58</td>
<td>$34.25</td>
<td>$100.25</td>
<td>$129.50</td>
<td>$49.40</td>
<td>$53.20</td>
</tr>
<tr>
<td>59</td>
<td>$33.50</td>
<td>$97.50</td>
<td>$131.75</td>
<td>$49.20</td>
<td>$54.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shipments in All Other Packaging / Maximum Weight In Lbs.</th>
<th>Weight</th>
<th>FedEx First Overnight</th>
<th>FedEx Priority Overnight</th>
<th>FedEx Standard Overnight</th>
<th>FedEx 2Day*</th>
<th>FedEx Express Saver*</th>
</tr>
</thead>
<tbody>
<tr>
<td>101 lbs</td>
<td>$271.69</td>
<td>$249.64</td>
<td>$223.21</td>
<td>$95.95</td>
<td>$80.80</td>
<td></td>
</tr>
<tr>
<td>102</td>
<td>$270.38</td>
<td>$248.32</td>
<td>$221.62</td>
<td>$95.42</td>
<td>$80.36</td>
<td></td>
</tr>
<tr>
<td>103</td>
<td>$269.07</td>
<td>$246.90</td>
<td>$219.92</td>
<td>$94.98</td>
<td>$79.84</td>
<td></td>
</tr>
<tr>
<td>104</td>
<td>$267.76</td>
<td>$245.44</td>
<td>$218.24</td>
<td>$94.46</td>
<td>$79.31</td>
<td></td>
</tr>
<tr>
<td>105</td>
<td>$266.45</td>
<td>$243.90</td>
<td>$216.52</td>
<td>$93.93</td>
<td>$78.80</td>
<td></td>
</tr>
<tr>
<td>106</td>
<td>$265.14</td>
<td>$242.32</td>
<td>$214.78</td>
<td>$93.41</td>
<td>$78.29</td>
<td></td>
</tr>
<tr>
<td>107</td>
<td>$263.74</td>
<td>$240.64</td>
<td>$213.00</td>
<td>$92.88</td>
<td>$77.76</td>
<td></td>
</tr>
<tr>
<td>108</td>
<td>$262.33</td>
<td>$238.84</td>
<td>$211.24</td>
<td>$92.34</td>
<td>$77.23</td>
<td></td>
</tr>
<tr>
<td>109</td>
<td>$260.93</td>
<td>$237.04</td>
<td>$209.40</td>
<td>$91.80</td>
<td>$76.70</td>
<td></td>
</tr>
<tr>
<td>110</td>
<td>$259.53</td>
<td>$235.24</td>
<td>$207.56</td>
<td>$91.26</td>
<td>$76.17</td>
<td></td>
</tr>
<tr>
<td>111</td>
<td>$258.13</td>
<td>$233.44</td>
<td>$205.72</td>
<td>$90.72</td>
<td>$75.63</td>
<td></td>
</tr>
<tr>
<td>112</td>
<td>$256.72</td>
<td>$231.64</td>
<td>$203.88</td>
<td>$90.18</td>
<td>$75.10</td>
<td></td>
</tr>
<tr>
<td>113</td>
<td>$255.32</td>
<td>$229.84</td>
<td>$202.04</td>
<td>$89.64</td>
<td>$74.56</td>
<td></td>
</tr>
<tr>
<td>114</td>
<td>$253.92</td>
<td>$228.04</td>
<td>$200.20</td>
<td>$89.10</td>
<td>$74.02</td>
<td></td>
</tr>
<tr>
<td>115</td>
<td>$252.52</td>
<td>$226.24</td>
<td>$198.36</td>
<td>$88.56</td>
<td>$73.48</td>
<td></td>
</tr>
<tr>
<td>116</td>
<td>$251.12</td>
<td>$224.44</td>
<td>$196.52</td>
<td>$88.02</td>
<td>$72.93</td>
<td></td>
</tr>
<tr>
<td>117</td>
<td>$249.72</td>
<td>$222.64</td>
<td>$194.68</td>
<td>$87.48</td>
<td>$72.39</td>
<td></td>
</tr>
<tr>
<td>118</td>
<td>$248.32</td>
<td>$220.84</td>
<td>$192.84</td>
<td>$86.94</td>
<td>$71.84</td>
<td></td>
</tr>
<tr>
<td>119</td>
<td>$246.92</td>
<td>$219.04</td>
<td>$191.00</td>
<td>$86.40</td>
<td>$71.29</td>
<td></td>
</tr>
<tr>
<td>120</td>
<td>$245.52</td>
<td>$217.24</td>
<td>$189.16</td>
<td>$85.86</td>
<td>$70.75</td>
<td></td>
</tr>
</tbody>
</table>

The first distribution scenario considers a linear shipping cost function when partial delivery is allowed (LSC/PD). The shipping cost per item is linearly decreasing with the shipping time. When partial delivery is accepted by customers, all items completed at the end of each production day can be shipped. There is no need to wait for any remaining items in an order to be produced.

When customers shop online, shipping cost may be independent of customer locations and quoted only according to shipping mode (shipping time). Figure 2 is a real example, where the shipping cost is given before a customer tells her location. Therefore, shipping cost can be indifferent to customer location. This assumption is relaxed in section 7, where we extend our models to also consider customer locations. This scenario occurs, for example, for overseas shipping.

Considering a general linear function, the shipping cost for one item in order \( i \) is \( b - a * (d_i - t_i) \), where \( a > 0, b > 0 \), and \( d_i - t_i \) is the shipping time with delivery mode \( r_i = d_i - t_i \). The value \( b \) is the cost per item for overnight shipping, which has the shortest shipping time and highest cost, while \( a \) is the dollar value of increasing shipping time by one more day. Because shipping cost cannot be negative, we require that \( a < b/m \) so that \( b - a * m > 0 \), where \( m \) is the number of production days in a planning horizon and also a upper bound on the shipping time for all orders.

Notice that the shipping cost in Table 2 is almost linearly increasing with the weight of items shipped. Therefore, for our purposes, shipping cost is linearly increasing with the number of items shipped in a single product scenario. The shipping cost for order \( i \) is \( Q_i \cdot [b - a(d_i - t_i)] \), where \( Q_i, d_i, \) and
Figure 2: Example showing that shipping cost can depend only on shipping time.

$t_i$, respectively, are the order quantity shipped, production due date, and shipping date, respectively, for order $i$.

Without loss of generality, we assume that $Q_i < c$ for each order $i$ in this paper. We also assume the planning horizon $m$ is longer than one day.

### 3.1 Linear Programming Model of the LSC/PD Problem

Let $Q_{ij}$ denote the number of items produced for customer order $i$ in production day $j$, where $i \in \{1, ..., n\}$ and $j \in \{1, ..., m\}$. Because partial delivery is allowed, the shipping date $t_i$ is the same as the production date $j$. Therefore, the shipping cost for $Q_{ij}$ items is $Q_{ij}[b - a(d_i - j)]$.

In the following mixed integer programming model, the objective is to find the optimal production schedule, determined by decision variables $Q_{ij}$, to minimize the total shipping cost.

**MIP-LSC/PD**:

Minimize $\sum_{i=1}^{n} \sum_{j=1}^{d_i} [Q_{ij}(b - a(d_i - j))]$

subject to

$$\sum_{j=1}^{d_i} Q_{ij} = Q_i, \quad i = 1, \ldots, n$$  \hspace{1cm} (1)

$$\sum_{i=1}^{n} Q_{ij} \leq c, \quad j = 1, \ldots, d_j$$  \hspace{1cm} (2)

$$Q_{ij} \geq 0, \quad i = 1, \ldots, n; \quad j = 1, \ldots, d_j$$  \hspace{1cm} (3)

$$Q_{ij} \text{ integer} \quad i = 1, \ldots, n; \quad j = 1, \ldots, m$$  \hspace{1cm} (4)
We minimize the objective function value, which is the total shipping cost for the \( n \) orders. Constraint (1) ensures that all items required in order \( i \) are produced on or before the order’s production due date \( d_i \). Constraint (2) ensures that the daily production quantity does not exceed the daily production capacity \( c \). Constraints (3) and (4) simply ensure that the \( Q_{ij} \) for all \( i \) and \( j \) take nonnegative integer values.

We now show that the constraint matrix of MIP-LSC/PD is totally unimodular. To do so, we list the variables in the sequence \( Q_{11}, Q_{12}, \ldots, Q_{1m}; \ldots; Q_{n1}, Q_{n2}, \ldots, Q_{nm} \). The constraint matrix \( A \) for constraints (1) and (2) in MIP-LSC/PD is as below.

\[
\begin{bmatrix}
1 & \cdots & d_1^\text{th} & \cdots & m & \cdots & m + d_2^\text{th} & \cdots & 2m & \cdots & (n-1)m & \cdots & d_n^\text{th} & \cdots & nm
\end{bmatrix}
\]

The above \((0, 1)\) matrix \( A \) is totally unimodular because it satisfies the following two conditions:

1. Each column contains at most two nonzero elements, which are the number: 1.
2. The rows of A can be partitioned into two sets: \( A_1 \), which includes row 1 to row \( n \), and \( A_2 \) including row \( n + 1 \) to row \( n + m \), such that the two 1 entries in a column are in two different sets of rows.

Combined with the identity constraint matrix \( I \) from constraint (4), the constraint matrix for model MIP-LSC/PD becomes \([A, I]\), which is totally unimodular because \( A \) is totally unimodular.

Therefore, model MIP-LSC/PD has an integral polytope, which has only integral extremal points. Thus, MIP-LSC/PD is equivalent to the following linear programming model, in which we remove constraint (4).

**LP-LSC/PD:**

\[
\text{Minimize } \sum_{i=1}^{n} \sum_{j=1}^{d_i} (Q_{ij}(b-a(d_i-j)))
\]

subject to (1), (2), and (3).

### 3.2 Exact Solution of the LSC/PD Problem

In order to find the exact solution of problem LSC/PD, we analyze the structure of an optimal production schedule for a single machine. In an optimal production schedule, production should be continuous, without any idle time. Any schedule that has idle time during production can be converted into a schedule without idle time by removing all idle time and advancing the production start times of all
orders in the same sequence as they appear in the current schedule. This process does not increase the objective function value.

**Nonpreemptive Earliest Due Date Schedule:** A production schedule is a nonpreemptive earliest due date (NEDD) schedule when orders are sorted according to earliest due date first and processed nonpreemptively and continuously without idle time.

Figure 3 provides an example of a NEDD schedule in which the production finishing time for an order \( j \) is \( \left( \sum_{i=1}^{i=j} Q_i \right)/c \). Next we show that a NEDD schedule is not only feasible but also an optimal schedule for problem LSC/PD.

![Figure 3: Example of a NEDD schedule.](image)

**Lemma 1** A NEDD schedule is a feasible schedule.

**Proof:** A NEDD schedule is feasible, since it satisfies the following feasibility check condition: \( \sum_{i \in A_j} Q_i \leq j, \ \forall j = 1, \ldots, m. \)

**Theorem 1** A NEDD schedule is an optimal schedule for the LSC/PD problem.

![Figure 4: An interchange between two equal-quantity-portions of orders \( i \) and \( j \).](image)

**Proof:** The idea behind the proof is that any feasible preemptive or nonpreemptive schedule can be converted into a NEDD schedule without increasing the objective function value. Note that in any feasible schedule, interchanging the positions of two portions with equal quantities of orders \( i \) and \( j \) will not change the total shipping cost if the resulting schedule remains feasible. This is shown in Figure 4. In addition, any feasible non-NEDD schedule can be transformed into a feasible NEDD schedule by a finite
number of interchanges. Since each such interchange will not increase the total shipping cost, a NEDD schedule is optimal.

4 Linear Shipping Cost Problem Without Partial Delivery

In order to model the linear shipping cost problem when no partial delivery is allowed (LSC/NPD), we begin by analyzing the structure of an optimal production schedule. When partial delivery is not allowed, an order cannot be shipped until the last item in the order has finished production. Interrupting the production of an order and leaving the order half-finished for some time is not beneficial. Therefore, in an optimal schedule, orders are finished one-by-one nonpreemptively and continuously. The solution of the following mixed integer programming model is an optimal production schedule.

4.1 Mixed Integer Programming Model of LSC/NPD

In the MIP model, we use the following additional decision variables:

\[ X_{ik} = \begin{cases} 
1, & \text{if order } i \text{ is the } k^{th} \text{ order to be produced in the optimal schedule;} \\
0, & \text{otherwise.} 
\end{cases} \]

\[ Y_{ij} = \begin{cases} 
1, & \text{if the completion time of customer order } i \text{ falls in production day } j; \\
0, & \text{otherwise.} 
\end{cases} \]

\[ f'_i = \text{exact production finishing time of order } i, \text{ in days from the beginning of the planning horizon.} \]

\[ f_k = \text{exact production finishing time of the } k^{th} \text{ order to be produced according to the production schedule, in days from the beginning of the planning horizon.} \]

\[
\text{MIP-LSC/NPD:} \\
\text{Minimize } \sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ij} Q_i [b - a(d_i - j)] \\
\text{subject to} \\
\sum_{k=1}^{n} X_{ik} = 1, \quad i = 1, \ldots, n \\
\sum_{i=1}^{n} X_{ik} = 1, \quad k = 1, \ldots, n \\
f_k = f_{k-1} + \sum_{i=1}^{n} \frac{Q_i X_{ik}}{c}, \quad k = 1, \ldots, n \\
\sum_{j=1}^{m} Y_{ij} = 1, \quad i = 1, \ldots, n \\
f'_i \leq \sum_{j=1}^{m} j Y_{ij}, \quad i = 1, \ldots, n \\
f'_i \leq d_i, \quad i = 1, \ldots, n \\
f_k - f'_i \leq M Z_{ik}, \quad k, i = 1, \ldots, n
\]
\[ f'_i - f_k \leq MZ_{ik} \quad k, i = 1, \ldots, n \quad (12) \]
\[ X_{ik} \leq 1 - Z_{ik} \quad k, i = 1, \ldots, n \quad (13) \]
\[ f_0 = 0 \quad (14) \]

\[ X_{ik}, Y_{ij}, Z_{ik} \in \{0, 1\} \quad j = 1, \ldots, m \quad k, i = 1, \ldots, n \]

\( M \) is a large positive number

Since an optimal schedule is a sequential queue, one order can occupy only one position and one position can have only one order. Therefore, constraints (5) and (6) hold.

To get the exact production finishing time \( (f_k) \) of the \( k^{th} \) order to be produced in the optimal schedule, we use constraint (7) as an iteration step and constraint (14) gives the initial starting time. In constraint (7), \( f_{k-1} \), which is the production finishing time of the \( (k-1)^{th} \) order in the schedule, is also the production starting time of the \( k^{th} \) order. Since only the order in position \( k \) has \( X_{ik} = 1 \), \( \sum_{i=1}^{n} Q_{i}X_{ik} \) gives the size of the order in position \( k \) and then \( (\sum_{i=1}^{n} Q_{i}X_{ik})/c \) provides the production time (in days) of the order. Given the starting time and the production time, in constraint (7) we can determine the production finishing time for the order in position \( k \) in the schedule, where \( k \) is an integer from 1 to \( n \).

When \( X_{ik} = 1 \), which indicates that order \( i \) is produced in the \( k^{th} \) position in the optimal schedule, \( f'_i \) is equal to \( f_k \). Therefore, constraints (11), (12), and (13) specify the exact production finishing time \( (f'_i) \) of customer order \( i \). Because all orders have to be finished production on or before their production due dates, we have due date constraint (10).

Constraint (9) makes sure that an order ships in the same day it is finished. When the shipping date for order \( i \) is day \( j \), we have \( Y_{ij} = 1 \). Considering the objective function, we want to ship an order as early as possible in order to reduce the shipping cost. Under constraint (9), an order cannot ship earlier than its production finishing time. Therefore, an order ships on its production finishing date.

4.2 Complexity of the Problem

We now show that the LSC/NPD problem is strongly NP-hard. Therefore, it is probable that no polynomial time algorithm can be found to solve it (Garey, Graham, and Johnson (1978)).

**Theorem 2** The LSC/NPD problem is NP-hard in the strong sense.

**Proof:** We prove that 3-PARTITION is reducible to the LSC/NPD problem.

**3-PARTITION:** Given integers \( N \) and \( B \) and given a set of integers \( a_1, \ldots, a_{3N} \) such that \( \frac{B}{4} < a_i < \frac{B}{2}, \)
\( i = 1, \ldots, 3N, \) and \( \sum_{i=1}^{3N} a_i = NB, \) are there \( n \) pairwise disjoint three-element subsets \( A_j \subset \{1, \ldots, 3N\} \)
such that \( \sum_{i \in A_j} a_i = B, \) \( j = 1, \ldots, N? \)

We restate our optimization problem as a decision problem:
Decision Problem: Given a set of $n$ orders, each order $i$ demands $Q_i$ items that must be produced in a single lot on or before its production due date $d_i$. There are $s$ shipping modes for order $i$: $\{r_1, \ldots, r_s\}$, where $s = d_i$. The planning horizon is $m$. The cost of using shipping mode $r_k$, $k = 1, 2, \ldots, m$, is $c_k$ per item, where $c_1 < c_2 < \ldots < c_m$. Does there exist a schedule of customer orders where the total shipping cost is less than or equal to $Z$?

We now construct a LSC/NPD problem that has a solution if and only if 3-PARTITION has a solution. The planning horizon $m = N$. The production capacity $c = 3X + B$ items per day. There are $n = 3N$ orders with demands $Q_i = X + a_i$ for each order $i$, $i = 1, \ldots, n$, where $\sum_{i=1}^{n} Q_i = 3NX + NB$. The production due date of each order is $d_i = N$, $i = 1, \ldots, n$. The number of shipping modes for each order is $s = N$. Let $c_1 = 1$ and $c_{k+1} = c_k + 1$, $k = 1, \ldots, N - 1$. We set $Z = \sum_{i=1}^{N} 3Xc_i + \sum_{i=1}^{N} Bc_i$, where $X > \sum_{i=1}^{N} Bc_i$.

3 – PARTITION $\Rightarrow$ LP – LSC/NPD: The existence of a 3-Partition allows demand $Q_i$ to be grouped into $N$ lots, each of which has $3X + B$ items and exactly satisfies the demand of three customer orders using shipping mode $r_i$, $i = 1, \ldots, N$. (See Figure 5.) Note that a lot of $3X + B$ items will be completed every day. Because the shipping cost for each lot is $c_i(3X + B)$ using shipping mode $r_i$, $i = 1, \ldots, N$, the total cost is $Z = \sum_{i=1}^{N} c_i(3X + B)$.

LP – LSC/NPD $\Rightarrow$ 3 – PARTITION: Suppose that there exists a schedule for the LSC/NPD problem such that $Z \leq \sum_{i=1}^{N} 3Xc_i + \sum_{i=1}^{N} Bc_i$. We now show that there exists a 3-Partition by proving the following two claims.

Claim 1: No more than $3k$ orders finish production before production day $k$, where $k = 1, \ldots, N$.

Proof: Four orders cannot finish production before production day 1 because $4X > 3X + B$. Similarly, seven orders cannot finish production before production day 2 because $7X > 6X + 2B$, and so on.

Claim 2: Exactly $3k$ orders finish production before production day $k$, where $k = 1, \ldots, N$.

Proof: Suppose that there is a schedule in which less than $3k$ jobs finish production in production day $k$. The total cost is at least $3Xc_1 + \ldots + 2Xc_k + 4Xc_{k+1} + \ldots + 3Xc_N > Z$.

<table>
<thead>
<tr>
<th>$X + a_1$</th>
<th>$X + a_2$</th>
<th>$X + a_3$</th>
<th>$X + a_4$</th>
<th>$X + a_5$</th>
<th>$X + a_6$</th>
<th>$X + a_3N-2$</th>
<th>$X + a_{3N-1}$</th>
<th>$X + a_{3N}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$3X + B$</td>
<td>$6X + 2B$</td>
<td>$3NX + NB$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Day 1</td>
<td>Day 2</td>
<td>Day $N$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 5: Schedule for producing and delivering orders.

4.3 Worst Case Analysis for Problem LSC/NPD

**Theorem 3** The worst case bound for the NP-hard problem LSC/NPD is less than two and close to one when $m$ grows large.
Proof: To prove the theorem, we first simplify the objective function for problem LSC/NPD. As described in the beginning of section 4, an optimal production schedule should be nonpreemptive and continuous without any idle times during production. Based on a nonpreemptive and continuous production schedule, we can simplify the objective function.

\[
\sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ij} Q_i [b - a(d_i - j)]
\]

\[
= \sum_{i=1}^{n} (\sum_{j=1}^{m} Y_{ij}) Q_i b - \sum_{i=1}^{n} (\sum_{j=1}^{m} Y_{ij}) Q_i a d_i + \sum_{i=1}^{n} (\sum_{j=1}^{m} Y_{ij} Q_i) a j
\]

\[
= \sum_{i=1}^{n} Q_i b - \sum_{i=1}^{n} Q_i a d_i + \sum_{j=1}^{m} (a(c - c_1) + 2a(c + c_1 - c_2) + \ldots + (m - 1)a(c + c_{m-2} - c_{m-1}) + ma(c + c_{m-1}))
\]

\[
= \sum_{i=1}^{n} Q_i (b - a d_i) + acm \frac{1 + m}{2} + a \sum_{j=1}^{m-1} c_j.
\] (15)

Since the first two terms in objective function (15) are constants, we focus on the last term \(a \sum_{j=1}^{m-1} c_j\). When \(c_j = 0, \forall 1 \leq j \leq m - 1\), we obtain a lower bound on the optimal solution. When \(c_j = c, \forall 1 \leq j \leq m - 1\), we get a upper bound on the total shipping cost provided by any feasible nonpreemptive and continuous production schedule, which satisfies the due date constraints and the limited capacity constraints. Therefore, we have a worst case bound for any feasible nonpreemptive and continuous production schedule as follows.

Total shipping cost provided by any feasible nonpreemptive and continuous production schedule
\[
\frac{\text{Total shipping cost provided by an optimal solution}}{\text{Total shipping cost provided by an optimal solution}}
\]
\[\sum_{i=1}^{n} Q_i (b - a d_i) + a c m \frac{1+m}{2} + a (m-1) c \leq \sum_{i=1}^{n} Q_i (b - a d_i) + a c m \frac{1+m}{2} \]
\[= 1 + \frac{\sum_{i=1}^{n} Q_i (b - a d_i) + a c m \frac{1+m}{2}}{a (m-1) c} \]
\[\leq 1 + \frac{\sum_{i=1}^{n} Q_i (b - a d_i) + a c m \frac{1+m}{2}}{a m c} \]
\[\leq 1 + \frac{2}{1+m} \quad (16)\]
\[1 + \frac{2}{1+m} \quad (17)\]

In order to make sure that any order has a positive shipping cost, we require that \(b - a m \geq 0\) in section 3. Also considering that the due date \(d_i\) for any order \(i\) is not beyond the planning horizon \(m\), we get \(b - a d_i \geq 0\). Therefore, inequality (16) holds.

Because the planning horizon \(m\) is at least two days, \(m \geq 2\), the value of equation (17) is less than 2 and decreases to 1 when \(m\) goes to infinity. We now thus proved that the worst case bound is less than 2 and close to 1 when \(m\) grows large.

### 4.4 A Heuristic Algorithm for Problem LSC/NPD

In section 4.3, we simplified the objective function to \(\sum_{i=1}^{n} Q_i (b - a d_i) + a c m \frac{1+m}{2} + a \sum_{j=1}^{m-1} c_j\). Because the first two terms are constants, in order to minimize the objective function, we just need to minimize the third term.

From this observation, we devise the following heuristic algorithm that cuts down the quantity \(c_j\) on each day \(j\), so that the value of \(\sum_{j=1}^{m-1} c_j\) could be reduced. We now describe the steps of the algorithm.

**Algorithm NPD**

**Step 1.** Schedule customer orders according to nondecreasing order of due dates. In the case of a tie, schedule larger order first. Set the initial value of \(i\) to be \(m - 1\).

**Step 2.** If \(0 < c_i < c\), go to Step 3. If \(c_i = 0\), go to Step 4.

**Step 3.** If all orders which are finished in day \(i + 1\) have a larger quantity than \(c_i\), then go to Step 4. Otherwise, among the orders which are started and finished in day \(i + 1\) and also have a less or equal quantity compared with \(c_i\), pick the order with largest quantity \((Q_j)\) and switch it with an equal quantity in \(c_i\). After switching, \(c_i\) is reduced to \(c_i - Q_j\). The switch may preempt an order (say \(O_k\)) into two portions whose finishing time falls on day \(i + 1\). If preemption occurs, there exists a set of orders (say set \(J\)) processed between the two portions of \(O_k\). Now adjust the schedule to be nonpreemptive by interchanging positions of the last portion of \(O_k\) with the orders in set \(J\). This
process achieves a nonpreemptive schedule without increasing the value of the objective function.

Go to Step 2.

**Step 4.** If \( i \) is greater than 1, set \( i = i - 1 \) and go to Step 2 again. Otherwise, STOP.

Step 1 of the NPD Algorithm gives an initial nonpreemptive schedule which satisfies the due dates and limited capacity constraints. Based on the initial schedule, Step 3 reduces the quantity \( c_i \) which is produced in day \( i \) and shipped later than day \( i \). If quantity \( c_i \) is equal to 0, then Step 4 is executed and the NPD Algorithm reduces quantity \( c_{i-1} \), if \( i > 1 \). When the process of reducing \( c_i \) is done for each day \( i \) from day \( m - 1 \) to day 1, the algorithm stops, and we get a better production schedule. In Figure 7, we show how to reduce \( c_{m-1} \), which is shown as a shadow region.

![Figure 7: Example to reduce \( c_{m-1} \).](image)

**Example to reduce \( c_{m-1} \)**

1. **(Step 1)** Schedule customer orders according to nondecreasing order of due dates. In the case of a tie, schedule larger order first. Set the initial value of \( i \) to be \( m - 1 \). Therefore \( c_i \) is equal to \( c_{m-1} \).

2. **(Step 2)** Since \( c_{m-1} \) (in shadow) is larger than zero, go to Step 3.

3. **(Step 3)** In day \( m \), the completed orders \( O_u \) and \( O_w \) have smaller quantities than \( c_{m-1} \). Since \( Q_w > Q_u \), pick up \( O_w \) to switch with an equal quantity in \( c_{m-1} \). After the switch, the shadow part \( c_{m-1} \) is reduced to \( c_{m-1} - Q_i \) which is shown in shadow in the second schedule in Figure 7. Now order \( Q_k \) is preempted after switch. Set \( J = \{Q_u\} \). Without increasing the objective function, we interchange the second portion of \( O_k \) with the orders in set \( J \). Then we obtain a nonpreemptive schedule as the third schedule in Figure 7. Go to Step 2.
4. **(Step 2)** Since the new $c_{m-1}$ is still larger than zero, go to Step 3.

5. **(Step 3)** Repeat the process and switch the order $O_u$ with an equal quantity in $c_{m-1}$. Note that $c_{m-1}$ is further reduced as shown in the fourth schedule in Figure 7. Go to Step 2.

6. **(Step 2)** Go to Step 3 again because $c_{m-1} > 0$.

7. **(Step 3)** As there is no completed orders in day $m$ which has a smaller quantity than the new $c_{m-1}$, go to Step 4.

8. **(Step 4)** Since $i$ is greater than 1, we set $i = i - 1$ and start the process to reduce $c_{m-2}$.

   After we repeat the process and finish reducing $c_1$, Algorithm NPD stops. The algorithm has polynomial computation time and it gives a near-optimal schedule, as shown in section 6 via many numerical experiments.

5 **The Nonlinear Shipping Cost Problem When Partial Delivery is Allowed**

In the real world, shipping cost may not be a linear function of shipping time. For example, if we quote the shipping cost from the FedEx website for an item, we could get a table including multiple shipping modes and their corresponding shipping costs as shown in Table 2. Plotting the shipping time and cost in a chart reveals that shipping cost is a convex piecewise function of shipping time as in Figure 8.

![Fedex rate piecewise function](image)

**Figure 8:** Piecewise rate function from FedEx.

Denote $G(x)$ as a shipping cost function, where $x$ is the shipping time in days. $G(x)$ is convex when the cost difference between overnight and one-day shipping is much larger than the difference between two-day and three-day shipping. A concave function has the opposite property. We illustrate the convex and concave forms of $G(x)$ in Figure 9. In the following subsections, we study the nonlinear shipping cost problem when partial delivery is allowed (NLSC/PD). We consider problem NLSC/PD with a convex shipping cost function $G(x)$ in subsection 5.1. In subsection 5.2, we discuss problem NLSC/PD when $G(x)$ is concave.
5.1 Exact Solution for the NLSC/PD Problem with Convex Shipping Cost

Here we show that a NEDD schedule is optimal for the NLSC/PD problem with a convex shipping cost function. In this scenario, the shipping cost function $G(x)$ is convex and partial delivery is accepted by customers.

Theorem 4 The NEDD schedule is optimal for the NLSC/PD problem with a convex shipping cost.

Proof: In NEDD, all orders are scheduled in nondecreasing order of due dates without preemption. Let us look at the ideas behind the proof first. Suppose that $S^*$ is a feasible schedule which is not a NEDD schedule. We show that $S^*$ can be transformed into a NEDD schedule by a series of order interchanges to be described shortly. We further show that in each such interchange, the objective function value either improves or remains the same. This proves the theorem.
Now we provide details of the proof. Since $S^*$ is not a NEDD schedule, there exist two orders $O_i$ and $O_j$ (or two partial orders) such that $d_j < d_i$ and order $O_i$ finishes before order $O_j$ (see Figure 10). Without loss of generality, we may assume that orders $O_i$ and $O_j$ (or two partial orders) have the same quantity $q$, $q \leq c$, and the order $O_i$ ($O_j$) starts and finishes production on the same day $i$ ($j$). It is obvious that interchanging the positions of $O_i$ and $O_j$ will not change the objective function value if $i = j$. For $i < j$, we know that $i < j \leq d_j < d_i$. Let $k = j - i$, $x_i = d_i - i$, and $x_j = d_j - j$, where $x_i$ ($x_j$) denotes the shipping time in number of days for order $O_i$ ($O_j$) in schedule $S^*$. The cost contributions of these two orders in schedule $S^*$ is $q(G(x_i) + G(x_j))$. Now we interchange the positions of $O_i$ and $O_j$. See Figure 10 for the new schedule. The cost contributions of these two orders in the new schedule is $q(G(x'_i) + G(x'_j))$, where $x'_i = d_i - j$ and $x'_j = d_j - i$. Note the shipping time of order $O_i$ decreases by $k$ days while the shipping time of order $O_j$ increases by $k$ days. Since $x_i > x_j + k$ and the cost function $g(x)$ is convex, we can easily show that $q(G(x'_i) + G(x'_j)) < q(G(x_i) + G(x_j))$. By performing a finite number of such interchanges, we could convert $S^*$ into a NEDD schedule by either improving the objective function value or leaving this value unchanged for each such interchange. Thus, the NEDD schedule is optimal.

In practice, most logistics companies have a convex shipping rate function similar to the one in Figure 8. Thus, NEDD is a useful schedule for real world NLSC/PD problems.

5.2 Exact Solution for the NLSC/PD Problem with Concave Shipping Cost

For the NLSC/PD problem with a concave shipping cost function, NEDD may not be an optimal solution. Figure 11 shows an example in which the NEDD schedule is not optimal.

In the example, a company has four accepted orders. Orders $Q_1$, $Q_2$, $Q_3$, and $Q_4$ require 60, 80, 80, 80, and the due dates are 1, 2, 3, and 3 days, respectively. The shipping rates and modes are as follows:

- Q1: 60 units, due in 1 day, Overnight, rate 380
- Q2: 80 units, due in 2 days, One day, rate 380
- Q3: 80 units, due in 3 days, Overnight, rate 280
- Q4: 80 units, due in 3 days, Two days, rate 160

The total ship cost for the NEDD schedule is $60 \times 30 + 80 \times 20 + 80 \times 30 + 60 \times 20 + 20 \times 30 + 80 \times 30 = 8000$.

The total ship cost for another schedule is $60 \times 30 + 40 \times 5 + 80 \times 30 + 20 \times 20 + 20 \times 30 + 80 \times 30 = 7800$.

Figure 11: NEDD is not optimal.
80, and 80 items respectively, which have to be delivered before the end of day 1, day 2, day 3, and day 3, respectively. The company can only produce 100 items in one day. If the rates for overnight shipping, one-day delivery, two-day delivery are $30, $20, $5 for one item, then the piecewise shipping cost function is concave in shipping time. From Figure 11, the total shipping cost based on the NEDD production schedule is $8000, while another feasible schedule lowers the total shipping cost to $7800. Therefore, NEDD is not an optimal schedule in this example.

Similar to the LSC/PD problem in section 3, the NLSC/PD problem can be solved with linear programming. Here we have a piecewise function $G(x)$, where $x$ is the shipping time in number of days. All of the constraints from the LSC/PD problem still hold for the NLSC/PD problem.

**LP-NLSC/PD (Nonlinear shipping cost with partial delivery allowed):**

Minimize $\sum_{i=1}^{n} \sum_{j=1}^{m} Q_{ij} G(d_i - j)$

subject to (1), (2), and (3).

6. The Nonlinear Shipping Cost Problem without Partial Delivery

In section 4, we studied the LST/NPD problem with linear shipping cost and without partial delivery. We constructed a mixed integer programming model MIP-LST/NPD for this problem. Similar to the LST/NPD problem, the problem with nonlinear shipping cost and without partial delivery (NLST/NPD) can be solved using a mixed integer programming model MIP-NLSC/NPD.

**6.1 Mixed Integer Programming Model for Problem NST/NPD**

MIP-NLSC/NPD is similar to MIP-LSC/NPD in section 4; we simply replace the linear shipping rate function $[b - a(d_i - j)]$ with a piecewise shipping rate function $G(d_i - j)$ in the objective.

MIP-NLSC/NPD (Nonlinear shipping cost without partial delivery):

Minimize $\sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ij} Q_i G(d_i - j)$

subject to

Constraints which are the same as in MIP-LSC/NPD.

6.2 Performance Evaluation of Algorithm NPD for Problem NST/NPD

Since the nonlinear shipping cost function $G(x)$ decreases with shipping time $x$, the manufacturer can also reduce total shipping cost by using the NPD algorithm in section 4, which helps ship some finished orders earlier without shipping other orders later. Thus, we use Algorithm NPD as a heuristic for the NLSC/NPD problem.
In order to evaluate the performance of the heuristic NPD Algorithm, we coded the algorithm in Java and tested it with three problem sets having different planning horizons $m$ and number of orders $n$. Each problem set has 10 problem instances. We fixed the daily production capacity $c$ at 10 in the first two problem sets and at 500 in the last problem set. We then compare these results to the optimal solutions.

According to Hall and Posner (2001), there are two approaches to generate problem instances. One is to “generate data without regarding for feasibility and then discard instances that have no feasible schedule.” For example, given the number of orders $n$ and the planning horizon $m$, each order’s due date $d_i$ is an integer generated randomly from a uniform distribution in the interval $[1, m]$. The quantity $Q_i$ for each order is an integer generated randomly from a uniform distribution in the interval $[1, c]$, where $c$ is the daily production capacity. After an order with $d_i$ and $Q_i$ is generated, we apply feasibility condition

$$\sum_{i \in A_j} Q_i \leq c, \quad j = 1, \ldots, m,$$

where $A_j = \{i|d_i \leq j\}$ denotes a set of accepted orders having delivery due date on or before day $j$. When the order violates the feasibility condition, we discard it and generate a new order.

**Table 3.** Numerical results when $n = 5$ and $m = 5.$

<table>
<thead>
<tr>
<th></th>
<th>Optimal Solution</th>
<th>CPU Time (Seconds)</th>
<th>Heuristic Algorithm NPD</th>
<th>CPU Time (Seconds)</th>
<th>Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4740.50</td>
<td>0.05</td>
<td>4740.50</td>
<td>0.03</td>
<td>0.00%</td>
</tr>
<tr>
<td>2</td>
<td>4643.33</td>
<td>1.04</td>
<td>4643.33</td>
<td>0.02</td>
<td>0.00%</td>
</tr>
<tr>
<td>3</td>
<td>6179.00</td>
<td>0.11</td>
<td>6179.00</td>
<td>0.07</td>
<td>0.00%</td>
</tr>
<tr>
<td>4</td>
<td>5104.50</td>
<td>0.08</td>
<td>5104.50</td>
<td>0.04</td>
<td>0.00%</td>
</tr>
<tr>
<td>5</td>
<td>2792.00</td>
<td>0.03</td>
<td>2792.00</td>
<td>0.01</td>
<td>0.00%</td>
</tr>
<tr>
<td>6</td>
<td>3905.50</td>
<td>0.06</td>
<td>3905.50</td>
<td>0.02</td>
<td>0.00%</td>
</tr>
<tr>
<td>7</td>
<td>5190.00</td>
<td>0.05</td>
<td>5190.00</td>
<td>0.03</td>
<td>0.00%</td>
</tr>
<tr>
<td>8</td>
<td>4818.83</td>
<td>0.06</td>
<td>4818.83</td>
<td>0.03</td>
<td>0.00%</td>
</tr>
<tr>
<td>9</td>
<td>5866.50</td>
<td>0.13</td>
<td>5866.50</td>
<td>0.06</td>
<td>0.00%</td>
</tr>
<tr>
<td>10</td>
<td>5471.17</td>
<td>0.09</td>
<td>5471.17</td>
<td>0.05</td>
<td>0.00%</td>
</tr>
</tbody>
</table>

For performance comparison, we use CPLEX to obtain the optimal solution for the MIP model. Tables 3 and 4 provide the performance measures for the heuristic NPD Algorithm for two problem sets: $(n, m) = (5, 5)$ and $(n, m) = (8, 8)$. Each line is one problem instance. The second through sixth columns show, respectively, the optimal total shipping cost from the MIP model, the time (in seconds) that CPLEX takes to solve the MIP model, the total shipping cost given by the heuristic NPD Algorithm, the time that the algorithm takes to give the heuristic solutions, and the percentage gap between the heuristic value and the optimal value. From Tables 3 and 4, the NPD algorithm is efficient, takes much less than one tenth of a second, and gives an optimal or a very close optimal value when the number of
orders is small and the planning horizon is short.

For problems with a large number of orders and a long planning horizon, CPLEX takes a long time to give the solution. Therefore, we consider the lower bound provided by relaxing the MIP to an LP and the lower bound from relaxing the problem to allow partial delivery of orders.

Let LP\textsubscript{relaxation} denote the LP model generated by treating all integer variables as real variables in MIP-NLSC/NPD. Using CPLEX, we can solve the LP\textsubscript{relaxation} model in seconds and obtain a lower bound for the NLSC/NPD problem.

If we relax the no partial delivery requirement in the NLSC/NPD problem, the problem becomes NLSC/PD. Thus we can use the LP-NLSC/PD model, which CPLEX solves efficiently, to provide a lower bound on the optimal solution. For convenience of discussion, we denote the LP-NLSC/PD model as PD\textsubscript{relaxation}.

In order to evaluate the quality of the two lower bounds, we test two problem sets and compare the results in Tables 5 and 6. The sixth and ninth columns, respectively, show the percentage gaps between the lower bound value and the optimal value, for LP\textsubscript{relaxation} and PD\textsubscript{relaxation}, respectively. From the gaps, we observe that PD\textsubscript{relaxation} provides a better lower bound for the objective function value.

Table 5. Lower bound performance when \( n = 5 \) and \( m = 5 \).

<table>
<thead>
<tr>
<th></th>
<th>Optimal Solution</th>
<th>CPU (Seconds)</th>
<th>LP\textsubscript{relaxation}</th>
<th>CPU</th>
<th>Gap</th>
<th>PD\textsubscript{relaxation}</th>
<th>CPU</th>
<th>GAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1286.50</td>
<td>0.05</td>
<td>1218.00</td>
<td>0.04</td>
<td><strong>05.325%</strong></td>
<td>1286.50</td>
<td>0.02</td>
<td><strong>0.00%</strong></td>
</tr>
<tr>
<td>2</td>
<td>1679.83</td>
<td>0.07</td>
<td>1520.00</td>
<td>0.03</td>
<td><strong>09.515%</strong></td>
<td>1679.83</td>
<td>0.02</td>
<td><strong>0.00%</strong></td>
</tr>
<tr>
<td>3</td>
<td>2086.87</td>
<td>0.07</td>
<td>1822.00</td>
<td>0.04</td>
<td><strong>12.692%</strong></td>
<td>2073.17</td>
<td>0.03</td>
<td><strong>0.66%</strong></td>
</tr>
<tr>
<td>4</td>
<td>4288.00</td>
<td>0.19</td>
<td>3130.35</td>
<td>0.08</td>
<td><strong>26.997%</strong></td>
<td>3922.67</td>
<td>0.05</td>
<td><strong>8.52%</strong></td>
</tr>
<tr>
<td>5</td>
<td>3369.50</td>
<td>0.04</td>
<td>2652.53</td>
<td>0.03</td>
<td><strong>21.278%</strong></td>
<td>3118.33</td>
<td>0.03</td>
<td><strong>7.45%</strong></td>
</tr>
<tr>
<td>6</td>
<td>2565.17</td>
<td>0.05</td>
<td>2174.72</td>
<td>0.02</td>
<td><strong>15.221%</strong></td>
<td>2382.50</td>
<td>0.02</td>
<td><strong>7.12%</strong></td>
</tr>
<tr>
<td>7</td>
<td>1788.23</td>
<td>0.05</td>
<td>1696.90</td>
<td>0.04</td>
<td><strong>05.107%</strong></td>
<td>1772.25</td>
<td>0.04</td>
<td><strong>0.89%</strong></td>
</tr>
<tr>
<td>8</td>
<td>6399.00</td>
<td>0.13</td>
<td>4085.98</td>
<td>0.06</td>
<td><strong>36.147%</strong></td>
<td>6125.00</td>
<td>0.05</td>
<td><strong>4.28%</strong></td>
</tr>
<tr>
<td>9</td>
<td>5206.50</td>
<td>0.39</td>
<td>3608.17</td>
<td>0.09</td>
<td><strong>30.699%</strong></td>
<td>4932.50</td>
<td>0.07</td>
<td><strong>5.26%</strong></td>
</tr>
<tr>
<td>10</td>
<td>3129.27</td>
<td>0.06</td>
<td>2426.00</td>
<td>0.05</td>
<td><strong>22.474%</strong></td>
<td>3088.17</td>
<td>0.05</td>
<td><strong>1.31%</strong></td>
</tr>
</tbody>
</table>
Table 6. Lower bound performance when \( n = 8 \) and \( m = 8 \).

<table>
<thead>
<tr>
<th>Optima</th>
<th>CPU</th>
<th>LP_relaxation</th>
<th>cpu</th>
<th>Gap</th>
<th>PD_relaxation</th>
<th>CPU Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2043.70</td>
<td>50</td>
<td>1900.50</td>
<td>0</td>
<td>0.007%</td>
<td>0.00%</td>
</tr>
<tr>
<td>2</td>
<td>2568.74</td>
<td>128</td>
<td>2313.50</td>
<td>0</td>
<td>0.936%</td>
<td>0.09%</td>
</tr>
<tr>
<td>3</td>
<td>3155.44</td>
<td>104</td>
<td>2726.50</td>
<td>0</td>
<td>13.594%</td>
<td>0.52%</td>
</tr>
<tr>
<td>4</td>
<td>3789.92</td>
<td>63</td>
<td>3139.50</td>
<td>0</td>
<td>17.162%</td>
<td>0.25%</td>
</tr>
<tr>
<td>5</td>
<td>4621.92</td>
<td>40</td>
<td>3552.50</td>
<td>0</td>
<td>23.138%</td>
<td>0.38%</td>
</tr>
<tr>
<td>6</td>
<td>5867.35</td>
<td>9</td>
<td>3965.50</td>
<td>0</td>
<td>32.414%</td>
<td>0.48%</td>
</tr>
<tr>
<td>7</td>
<td>4804.90</td>
<td>0</td>
<td>3539.12</td>
<td>0</td>
<td>26.344%</td>
<td>0.52%</td>
</tr>
<tr>
<td>8</td>
<td>5989.67</td>
<td>0</td>
<td>4187.46</td>
<td>0</td>
<td>30.089%</td>
<td>0.25%</td>
</tr>
<tr>
<td>9</td>
<td>7567.17</td>
<td>0</td>
<td>4835.81</td>
<td>0</td>
<td>36.095%</td>
<td>0.48%</td>
</tr>
<tr>
<td>10</td>
<td>2187.55</td>
<td>60</td>
<td>1988.57</td>
<td>0</td>
<td>0.007%</td>
<td>0.00%</td>
</tr>
</tbody>
</table>

With the lower-bound values provided by PD_\(_{\text{relax}}_\) relaxation, we test the performance of the heuristic NPD Algorithm in a problem set with a large number of orders \( (n = 500) \) and a long planning horizon \( (m = 15) \). We set the daily production capacity \( c \) as 500 items. In Table 7, we compare the result from the PD_\(_{\text{relax}}_\) relaxation lower bound and the lower bound from the heuristic NPD Algorithm in the sixth column. The gap values are very small. Thus, the NPD Algorithm provides a near optimal solution for the NLSC/NPD problem even when the number of orders is large and the planning horizon is long.

Table 7. Heuristic performance when \( n = 500 \) and \( m = 15 \).

<table>
<thead>
<tr>
<th>PD_relaxation</th>
<th>CPU Time (Seconds)</th>
<th>Heuristic Algorithm NPD</th>
<th>CPU Time (Seconds)</th>
<th>Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1035660.0</td>
<td>75.39</td>
<td>1037302.0</td>
<td>0.09</td>
</tr>
<tr>
<td>2</td>
<td>1070380.0</td>
<td>83.86</td>
<td>1073323.0</td>
<td>0.08</td>
</tr>
<tr>
<td>3</td>
<td>1070350.0</td>
<td>73.96</td>
<td>1072679.5</td>
<td>0.09</td>
</tr>
<tr>
<td>4</td>
<td>954790.0</td>
<td>64.24</td>
<td>958078.0</td>
<td>0.12</td>
</tr>
<tr>
<td>5</td>
<td>888835.0</td>
<td>69.53</td>
<td>891849.4</td>
<td>0.09</td>
</tr>
<tr>
<td>6</td>
<td>863386.0</td>
<td>70.25</td>
<td>863751.1</td>
<td>0.07</td>
</tr>
<tr>
<td>7</td>
<td>1031470.0</td>
<td>80.86</td>
<td>1032906.0</td>
<td>0.15</td>
</tr>
<tr>
<td>8</td>
<td>1150210.0</td>
<td>76.92</td>
<td>1150482.0</td>
<td>0.08</td>
</tr>
<tr>
<td>9</td>
<td>1179820.0</td>
<td>75.30</td>
<td>1180438.0</td>
<td>0.09</td>
</tr>
<tr>
<td>10</td>
<td>1147010.0</td>
<td>76.15</td>
<td>1147626.5</td>
<td>0.11</td>
</tr>
</tbody>
</table>

7 Extensions of Shipping Cost to Account for Customer Locations

We now discuss extensions of the models for the nonlinear shipping cost problems. In sections 5 and 6, we studied the integration problem when the nonlinear shipping cost \( G(x) \) is independent of customer locations. However, some third party logistics companies such as FedEx may charge shipping cost according to not only shipping time but also customer location. Below in Table 8 is a screenshot of a shipping cost table on the FedEx website, which categorizes customers into different zones according to the zip codes of their locations. Shipping cost is different for each zone.
To consider customer location in a shipping cost function, we model the shipping cost per item for order $i$ using $G_i(x)$, $i = 1, 2, \ldots, n$. Functions $G_i(x)$ and $G_j(x)$ may be identical if order $i$ and order $j$ are from customers in a same zone. Otherwise, there is a gap between these two function values. The gap decreases when shipping time increases. We illustrate the shipping cost functions in Figure 12 when order $i$ and order $j$ are from customers in different zones.

Table 8. Shipping cost varies with customer location.

When partial delivery is allowed, we can slightly modify model LP-NLSC/PD to account for customer locations. For each order $i$, we use its shipping cost function $G_i(x)$ in the objective function. Therefore, we get the following LP model. The decision variables $Q_{ij}$ determine an optimal production schedule. CPLEX solves the LP model in polynomial time.
subject to

Constraints which are the same as in LP-NLSC/PD.

When partial delivery is not accepted by customers, in order to consider customer location in shipping cost, we replace the shipping cost function $G(d_i - j)$ in the objective function of the model MIP-NLSC/NPD with $G_i(d_i - j)$. We obtain the following MIP model.

$$\text{Minimize} \quad \sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ij} Q_i G_i(d_i - j)$$

subject to

Constraints which are the same as in MIP-NLSC/NPD.

The problem considering customer location in a nonlinear shipping cost function when partial delivery is not allowed is strongly NP-hard. In order to get a heuristic algorithm for the problem, we slightly modify the NPD Algorithm in section 4. We adjust the first step in the NPD Algorithm to get an initial schedule using earliest due date first and break ties by putting an order with longer distance first. This practice makes sense because the shipping cost for a long distance is more expensive than that for a short distance.

8 Summary

In this paper, we describe practically useful models and algorithms for developing schedules to coordinate production and transportation in a make-to-order firm that operates under a commit-to-delivery business mode. Our objective is to minimize the total shipping cost. We consider eight realistic distribution scenarios: (a) two cost functions, linear or nonlinear shipping cost function of shipping time; (b) two modes of deliveries, partial delivery allowed or not allowed, and (c) two ways to deal with customer location, considering or without considering customer location in shipping cost. In the case of partial delivery, we can split a customer order into several small orders that have the same due dates. We can produce and ship each small order separately. When partial delivery is not allowed, customer orders cannot be split and must be delivered all together.

We show that problems with partial delivery can be formulated as an LP and solved efficiently. On the other hand, problems with no partial delivery are NP-hard. We show that this latter problem can be formulated as an MIP.

For the first four scenarios where customer location is not considered in shipping cost, we have the following results. In the two scenarios when partial delivery is allowed, we show that nonpreemptive
EDD production schedules are optimal except when a shipping cost function is not only nonlinear but also concave. In the two scenarios when partial delivery is not allowed, we prove that the problem is NP-hard. When shipping cost is a linear function of shipping time, we show that the worst case bound for any feasible solution to the NP-hard problem is $1 + \delta$, where $\delta$ is a small number in the interval $(0,1)$. When the production planning horizon goes to infinity, $\delta$ goes to zero and the bound goes to one. This result is unusual for an NP-hard problem. To our knowledge, the only other NP-hard problem with the worst case bound close to one is the graph coloring problem. An efficient heuristic algorithm is provided for the NP-hard problem.

One contribution of our work is to model some important practical order-scheduling problems encountered in the domain of supply chain management. These problems differ from classical scheduling problems since these problems minimize total shipping cost for all orders. Second, we devise an efficient algorithm to solve the problems optimally or near optimally. Because the solution of partial delivery problems serves as an effective lower bound for no partial delivery problems, we can evaluate the performance of our heuristic algorithm. We tested the algorithm with over 120 problem instances and presented some results. All computational tests reveal that the heuristic algorithm performs well as compared to the lower bound, even for large problems. Finally, we propose ways to extend our models to include transportation cost functions that are varied for different customer locations.

Acknowledgment

This research was supported in part by summer research grants from the University of Texas at Dallas.
References


Killian, Vida, Dell, personal communication (2003).


