Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 7069291, 9 pages
https://doi.org/10.1155/2020/7069291

Research Article

Production Scheduling considering Outsourcing Options and Carrier Costs

Byung-Cheon Choi [1], **Yunhong Min**, [2] **Myoung-Ju Park** [3], **and Kyung Min Kim** [4]

[1] *Department of Business Administration, Chungnam National University, 99 Daehak-ro, Yuseong-gu, Daejeon 34134, Republic of Korea*
[2] *Graduate School of Logistics, Incheon National University, 119 Academy-ro, Yeonsu-gu, Incheon 22012, Republic of Korea*
[3] *Department of Industrial and Management Systems Engineering, Kyung Hee University, 1732 Deogyeong-daero, Giheung-gu, Yongin-si, Gyeonggi-do 17104, Republic of Korea*
[4] *Department of Industrial Management and Engineering, Myongji University, 116 MyongJi-ro, Choein-gu, Yongin-si, Gyeonggi-do 17058, Republic of Korea*

Correspondence should be addressed to Kyung Min Kim; kmkim@mju.ac.kr

Received 8 August 2019; Accepted 29 October 2019; Published 3 January 2020

Guest Editor: Juneyoung Park

We consider a single-machine scheduling problem with outsourcing options in an environment where the cost information of the downstream is available via some information sharing technologies. The due date is assigned to the position differently from the traditional due date. Each job can be processed in-house or outsourced. Note that, for cost saving, as many due dates as the number of outsourced jobs should be canceled. An in-house job incurs a stepwise penalty cost for tardiness, and an outsourced job incurs an outsourcing cost. Thus, the objective is to minimize the total penalty and outsourcing cost minus the total profit from cost savings. We show that the problem is weakly NP-hard and investigate some polynomially solvable cases. Due to the high complexity of the dynamic programming, we developed heuristics and verified their performance through numerical experiments.

1. Introduction

Information sharing has been known as one of the effective tools that enable parties in a supply chain to coordinate with each other [1, 2]. A number of scenarios that exploit the capability of information sharing might be possible according to the characteristics of the supply chain. In this paper, we consider the problem of manufacturers who want to schedule their processing of jobs so that the cost of a carrier as well as her own costs is simultaneously minimized. We assume that a decision maker, i.e., manufacturer, is well aware of the cost structure of her carrier by some information sharing technologies.

Generally, a manufacturer in the current business environment is looking to entrust part of its production to a subcontractor and third-party logistics providers, respectively. With such outsourcing, the manufacturer can deliver goods to its consumers on time, reduce operating costs, and make its organization more flexible. Thus, the proper use of outsourcing can make a company more competitive [3, 4], which provides us with motivation to consider a production scheduling problem with an outsourcing option [5, 6, 7].

In this paper, we consider a single-machine scheduling problem such that a job can be processed directly by in-house resources or outsourced to a subcontractor. For in-house jobs, a due date is assigned, depending on the job sequence, and a stepwise penalty cost for tardiness can be incurred. A due date can be interpreted as the arrival time of a particular truck operated by an independent carrier. Note that these due dates are given not for specific jobs but for specific positions, and these due dates are referred to as *generalized due dates* (GDD) [8]. This reflects the situation in which a carrier's truck periodically visits a factory according to a planned timetable and picks up not a specific product, but one that is completed upon the truck's arrival. In this case, the truck's arrival time can be regarded as the due date. When some jobs are determined to be outsourced, it is assumed that as many due dates should be canceled as the number of outsourced jobs. In particular, we assume that any due date can be canceled. This is different from the canceling policy of Gerstl and Mosheiov [9], where the due dates are canceled in nonincreasing order.

As mentioned at the outset, the objective to be minimized of manufacturers consists of two components: one for their own cost and the other for the cost of the carrier. For a given set of due dates, the cost of the manufacturer is the outsourcing cost. The cost of the carrier, on the contrary, is the difference between the waiting cost due to late completion of jobs and the profit due to the canceled due dates. The canceling notification from the manufacturer which is sufficiently earlier than the due dates enables the carrier to redirect the corresponding resources, e.g., trucks and drivers. Thus, we assume that the canceling of due dates incurs profit instead of cost. Note that, for the manufacturer, all information on the carrier becomes available as a result of some information sharing technologies.

Remark 1. In Section 3, we will show that our problem is weakly NP-hard and has a pseudopolynomial time algorithm. Moreover, in Section 4, we will introduce some polynomially solvable cases. All these results also hold for the case where canceling due dates incurs costs instead of profits.

The main contributions of this paper are to establish the computational complexities of the problem and its variants, and to design heuristics whose effectiveness was verified through numerical experiments. For practitioners who consider our problem as an appropriate model for their problems, polynomially solvable cases and heuristics can provide some guidelines for tackling their own problems. In particular, manufacturers can align their production schedules with those of carriers or other downstream firms. This practice can result in reducing the transportation costs as well as overhead costs of manufacturers [10].

The scheduling problem with GDD was introduced by Hall [8]. In Hall [8] and Hall et al.'s [11] studies, the computational complexity of cases with various performance measures—such as maximum lateness, total weighted completion time, total weighted tardiness, and the weighted number of tardy jobs—is established in various machine environments, such as a single machine, parallel machines, and shops. However, the single-machine scheduling problem that minimizes the total weighted number of tardy jobs has been proven to be NP-hard by Sriskandarajah [12] and Yuan [13], and strongly NP-hard by Gao and Yuan [14]. Mosheiov and Oron [15] considered the problem on parallel identical machines to minimize the maximum tardiness and total tardiness. They showed that the schedule ordered by the shortest processing time (SPT) performs extremely well. Choi and Park [16] considered single-machine scheduling with GDD and identical due date intervals to minimize the weighted number of early and tardy jobs. They showed that the problem is strongly NP-hard and has no ρ-approximation algorithm for any fixed value $\rho > 1$. Gerstl and Mosheiov [9] considered two single-machine scheduling problems with GDD and rejection options with the objective of minimizing the total rejection cost plus the maximum tardiness or the total tardiness. They showed that the two machine problems are weakly NP-hard and developed heuristics whose performance was verified through the numerical experiments. To the best of our knowledge, Gerstl and Mosheiov [9] were the only ones to consider the scheduling problem with GDD and outsourcing options.

The remainder of the paper is organized as follows. Section 2 defines our problems. In Section 3, we prove the weak NP-hardness of our problems. Section 3 presents some conditions that make our problems polynomially solvable. In Section 4, we develop heuristics and conduct numerical experiments. Finally, in Section 6, we present our concluding remarks.

2. Notation and Problem Definition

In this section, we introduce the notations used throughout the paper and formally define our problem.

Let p_j and o_j be the processing time and the outsourcing cost of job $j \in \mathcal{J} := \{1, 2, \ldots, n\}$, respectively. Let d_g be the gth due date and q_g be the profit from canceling d_g for $g \in \mathcal{D} := \{1, 2, \ldots, n\}$. We assume that $d_1 \le d_2 \le \cdots \le d_n$. Let $\sigma = (h, \pi, \theta)$ be a schedule such that the number of in-house jobs is h and

$$\begin{aligned} \pi &= (\pi(1), \pi(2), \ldots, \pi(h)), \\ \theta &= (\theta(1), \theta(2), \ldots, \theta(h)), \end{aligned} \quad (1)$$

where $\pi(i)$ is the ith in-house job assigned to the $\theta(i)$th due date. Without loss of generality, assume that

$$\theta(1) < \theta(2) < \cdots < \theta(h). \quad (2)$$

Let $C_{\pi(i)}(\sigma)$ be the completion time of the ith in-house job in σ, which is calculated as

$$C_{\pi(i)}(\sigma) = \sum_{j=1}^{i} p_{\pi(j)}. \quad (3)$$

Let $V_g(x)$ be the penalty cost function for the tardiness of the job assigned to d_g, which is defined as

$$V_g(x) = \begin{cases} 0, & \text{if } x - d_g \le 0, \\ v_{1,g}, & \text{if } 0 < x - d_g \le \delta, \\ v_{2,g}, & \text{if } \delta < x - d_g \le 2\delta, \\ & \vdots \\ v_{w-1,g}, & \text{if } (w-2)\delta < x - d_g \le (w-1)\delta, \\ v_{w,g}, & \text{if } (w-1)\delta < x - d_g, \end{cases} \quad (4)$$

where $\delta > 0$ is a given threshold. Without loss of generality, assume that

$$0 < v_{1,g} < v_{2,g} < \cdots < v_{w,g}, \quad \text{for each } g \in \mathcal{D}. \quad (5)$$

This stepwise tardiness penalty cost has been considered in the field of transportation and semiconductor manufacturing [17–20].

Then, our problem is to find a schedule σ to minimize

$$z(\sigma) = \sum_{i=1}^{h} V_{\theta(i)}\left(C_{\pi(i)}(\sigma)\right) + \sum_{j\in\mathcal{O}} o_j - \sum_{g\in\mathcal{C}} q_g, \quad (6)$$

where $\mathcal{O}$ and $\mathcal{C}$ are the sets of outsourced jobs and canceled due dates, respectively; that is,

$$\begin{aligned}\mathcal{O} &= \mathcal{J}\backslash\{\pi(1), \pi(2), \ldots, \pi(h)\},\\ \mathcal{C} &= \mathcal{D}\backslash\{\theta(1), \theta(2), \ldots, \theta(h)\}.\end{aligned} \quad (7)$$

Let our problem be referred to as *Problem P*.

Proposition 1. *There exists an optimal schedule such that the in-house jobs are sequenced by SPT order.*

Proof. Proposition 1 holds immediately from the standard pairwise interchange scheme.

Henceforth, we consider only schedules satisfying Proposition 1. □

3. Computational Complexity

In this section, we show that Problem P is weakly NP-hard. For the proof of NP-hardness, we use the equal cardinality partition (ECP) problem, which can be stated as follows: given a set $\mathcal{A}$ of $2m$ elements, a bound $A \in \mathbb{Z}^+$, and a size a_j for each element $j \in \mathcal{A}$, $\mathcal{A}$ can be partitioned into two disjoint sets $\mathcal{A}_1$ and $\mathcal{A}_2$ such that

$$\begin{gathered}\sum_{j\in\mathcal{A}_1} a_j = \sum_{j\in\mathcal{A}_2} a_j = A,\\ \left|\mathcal{A}_1\right| = \left|\mathcal{A}_2\right| = m.\end{gathered} \quad (8)$$

Without loss of generality, assume that for any subset $\mathcal{S} \subseteq \mathcal{A}$ with $|\mathcal{S}| \geq m+1$,

$$\sum_{j\in\mathcal{S}} a_j > A. \quad (9)$$

Lemma 1. *Problem P is NP-hard, even if:*

(i) The step function for each $g \in \mathcal{D}$ has only two intervals, that is,

$$V_g(x) = \begin{cases} 0, & \text{if } x \leq d_g,\\ 1, & \text{if } d_g < x.\end{cases} \quad (10)$$

(ii) The intervals between the consecutive due dates are identical, that is,

$$d_g = g\Delta, \quad \text{for each } g \in \mathcal{D}, \quad (11)$$

where $\Delta > 0$ is a given threshold.

Proof. Given an instance of the ECP problem, we can construct an instance of our problems as follows. There are $n = (2m+1)$ jobs with

$$\left(p_j, o_j\right) = \begin{cases} \left(A^3 + A^2 + a_j, a_j/2A\right), & \text{for } j = 1, 2, \ldots, 2m,\\ \left(A^3 - mA^2 - A, 1\right), & \text{for } j = 2m+1.\end{cases} \quad (12)$$

Let $\Delta = A^3$ and

$$q_g = \begin{cases} 0, & \text{for } g = 1, 2, \ldots, m+1,\\ A^2, & \text{for } g = m+2, m+3, \ldots, 2m+1.\end{cases} \quad (13)$$

This reduction can be carried out in polynomial time. Henceforth, we show that a solution to the ECP problem exists if and only if there exists a schedule σ for Problem P with $z(\sigma) \leq 1/2 - mA^2$.

Suppose there exists a solution $(\overline{\mathcal{A}}_1, \overline{\mathcal{A}}_2)$ to the ECP problem. Then, we can construct a schedule $\overline{\sigma} = (\overline{h}, \overline{\pi}, \overline{\theta})$ such that

(i) The number of in-house jobs is $\overline{h} = m+1$

(ii) $\overline{\pi}$ is the SPT sequence of jobs in $\{2m+1\} \cup \overline{\mathcal{A}}_1$, where job $2m+1$ is the first job, according to Proposition 1

(iii) $\overline{\theta} = (1, 2, \ldots, m+1)$

Let $\overline{\mathcal{O}} = \overline{\mathcal{A}}_2$ and $\overline{\mathcal{C}} = \{m+2, m+3, \ldots, 2m+1\}$. Note that, for $i = 1, 2, \ldots, m+1$,

$$\begin{aligned} C_{\overline{\pi}(i)}(\overline{\sigma}) &= A^3 - mA^2 - A + \sum_{j=2}^{i} p_{\overline{\pi}(j)}\\ &= iA^3 - (m-i+1)A^2 - A + \sum_{j=2}^{i} a_{\overline{\pi}(j)}.\end{aligned} \quad (14)$$

Since $(m-i+1)A^2 + A > \sum_{j=2}^{i} a_{\overline{\pi}(j)}$, $i = 1, 2, \ldots, m$ and $\sum_{j=2}^{m+1} a_{\overline{\pi}(j)} = A$,

$$C_{\overline{\pi}(i)}(\overline{\sigma}) < iA^3 = d_i, \quad \text{for } i = 1, 2, \ldots, m, \quad (15)$$

$$C_{\overline{\pi}(m+1)}(\overline{\sigma}) = (m+1)A^3 = d_{m+1}. \quad (16)$$

Inequalities (15) and (16) show that no tardy in-house job exists in $\overline{\sigma}$. Furthermore, according to the reduction scheme above,

$$\begin{aligned}\sum_{j\in\overline{\mathcal{O}}} o_j &= \frac{1}{2A} \sum_{j\in\overline{\mathcal{A}}_2} a_j = \frac{1}{2},\\ \sum_{g\in\overline{\mathcal{C}}} q_g &= \sum_{g=m+2}^{2m+1} q_g = mA^2.\end{aligned} \quad (17)$$

Thus, by inequality (17),

$$z(\overline{\sigma}) = \sum_{j\in\overline{\mathcal{O}}} o_j - \sum_{g\in\overline{\mathcal{C}}} q_g = \frac{1}{2} - mA^2. \quad (18)$$

Suppose there exists a schedule $\widehat{\sigma} = (\widehat{h}, \widehat{\pi}, \widehat{\theta})$ for Problem P with $z(\widehat{\sigma}) \leq 1/2 - mA^2$. Let

$$\begin{aligned}\widehat{\mathcal{O}} &= \mathcal{J}\backslash\left\{\pi(1), \pi(2), \ldots, \pi(\widehat{h})\right\},\\ \widehat{C} &= \mathcal{D}\backslash\left\{\theta(1), \theta(2), \ldots, \theta(\widehat{h})\right\}.\end{aligned} \quad (19)$$

Since $z(\widehat{\sigma}) \le 1/2 - mA^2$,

$$\{m+2, m+3, \ldots, 2m+1\} \subseteq \widehat{\mathscr{C}}, \tag{20}$$

and no tardy job exists in $\widehat{\sigma}$ which implies that job $2m+1$ is the first in-house job by Proposition 1. □

Claim 1. $\widehat{C} = \{m+2, m+3, \ldots, 2m+1\}$.

Proof. By relation (20), $\widehat{h} \le m+1$. Thus, Claim 1 holds only by proving $\widehat{h} = m+1$. Suppose that $\widehat{h} \le m$. Then, $|\widehat{\mathscr{O}}| \ge m+1$ holds and, by assumption (9), we have

$$z(\widehat{\sigma}) \ge \sum_{j\in\widehat{\mathscr{O}}} o_j - \sum_{g\in\widehat{\mathscr{C}}} q_g = \frac{1}{2A} \sum_{j\in\widehat{\mathscr{O}}} a_j - mA^2 > \frac{1}{2} - mA^2. \tag{21}$$

This is a contradiction. □

Claim 2. $\sum_{j\in\widehat{\mathscr{O}}} o_j = 1/2$.

Proof. Since $z(\widehat{\sigma}) \le 1/2 - mA^2$ and $\widehat{\mathscr{C}} = \{m+2, m+3, \ldots, 2m+1\}$,

$$\sum_{j\in\widehat{\mathscr{O}}} o_j \le \frac{1}{2}. \tag{22}$$

Suppose that $\sum_{j\in\widehat{\mathscr{O}}} o_j < 1/2$. Then,

$$\sum_{j=2}^{m+1} a_{\widehat{\pi}(j)} > A. \tag{23}$$

By inequality (23), we have

$$C_{\widehat{\pi}(m+1)}(\widehat{\sigma}) = \sum_{j=1}^{m+1} p_{\widehat{\pi}(j)} = (m+1)A^3 - A + \sum_{j=2}^{m+1} a_{\widehat{\pi}(j)} > d_{m+1}, \tag{24}$$

and job $\widehat{\pi}(m+1)$ becomes tardy. This is a contradiction.

Let $\widehat{\mathscr{A}}_1 = \widehat{\mathscr{O}}$ and $\widehat{\mathscr{A}}_2 = \{1, 2, \ldots, 2m\} \backslash \widehat{\mathscr{O}}$. Then, $\widehat{\mathscr{A}}_1$ and $\widehat{\mathscr{A}}_2$ become the solution to the ECP problem.

Henceforth, for simplicity, assume that

$$p_1 \le p_2 \le \cdots \le p_n. \tag{25}$$

□

Lemma 2. *Problem P can be solved in pseudopolynomial time.*

Proof. We reduce Problem P to the shortest path (SP) problem. Let $N(0, 0; 0)$ and $N(n+1, n+1; \cdot)$ be the source and sink nodes, respectively. Let $N(a, b; C)$ be the node indicating:

(i) Which jobs in $\{1, 2, \ldots, a\}$ have to be outsourced
(ii) Which due dates in $\{d_1, d_2, \ldots, d_b\}$ have to be canceled
(iii) The total processing time of in-house jobs in $\{1, 2, \ldots, a\}$ is C

If $0 \le a' < a \le n$ and $0 \le b' < b \le n$, then let $N(a', b'; C - p_a)$ be connected to $N(a, b; C)$ with length

$$V_b(C) + \sum_{j=a'+1}^{a-1} o_j - \sum_{g=b'+1}^{b-1} q_g. \tag{26}$$

If $0 \le a \le n$ and $0 \le b \le n$, then let $N(a, b; C)$ be connected to the sink node with length

$$\sum_{j=a+1}^{n} o_j - \sum_{g=b+1}^{n} q_g. \tag{27}$$

The objective is to find the SP from the source to the sink node. In the reduced SP problem, the number of nodes is observed to be $O(n^2 \sum_{j=1}^{n} p_j)$, and each node has $O(n^2)$ outgoing arcs. Since the graph of the reduced SP problem is acyclic, the reduced SP problem can be solved in $O(n^4 \sum_{j=1}^{n} p_j)$ by the algorithm of Ahuja et al. [21]. □

Theorem 1. *Problem P is weakly NP-hard.*

Proof. Theorem 1 holds immediately from Lemmas 1 and 2. □

4. Polynomially Solvable Cases

In this section, we introduce three polynomially solvable cases.

Theorem 2. *Problem P is polynomially solvable if the processing times are identical, that is, $p_j = p$, $j = 1, 2, \ldots, n$.*

Proof. We will prove Theorem 2 by reducing Problem P to the SP problem such that the total number of nodes is bounded by n. Let $N(0, 0; 0)$ and $N(n+1, n+1)$ be the source and sink nodes, respectively. Let $N(a, b; h)$ be the node indicating

(i) Which jobs in $\{1, 2, \ldots, a\}$ have to be outsourced
(ii) Which due dates in $\{d_1, d_2, \ldots, d_b\}$ have to be canceled
(iii) The total processing time of in-house jobs in $\{1, 2, \ldots, a\}$ is hp

If $0 \le a' < a \le n$ and $0 \le b' < b \le n$, then let $N(a', b'; h-1)$ be connected to $N(a, b; h)$ with length

$$V_b(hp) + \sum_{j=a'+1}^{a-1} o_j - \sum_{g=b'+1}^{b-1} q_g. \tag{28}$$

If $0 \le a \le n$ and $0 \le b \le n$, then let $N(a, b; h)$ be connected to the sink node with length

$$\sum_{j=a+1}^{n} o_j - \sum_{g=b+1}^{n} q_g. \tag{29}$$

The objective is to find the SP from the source to the sink node. In the reduced SP problem, the number of nodes is observed to be $O(n^3)$, and each node has $O(n^2)$ outgoing arcs, respectively. Since the graph of the reduced SP problem is acyclic, the reduced SP problem can be solved in $O(n^5)$ by the algorithm of Ahuja et al. [21]. □

Theorem 3. *Problem P is polynomially solvable, if*

(i) *The intervals between the consecutive due dates are identical, that is,*

$$d_g = g\Delta, \quad \text{for each } g \in \mathscr{D}, \tag{30}$$

where Δ is a given threshold;

(ii) $p_j \leq \Delta$ *for each* $j \in \mathscr{J}$.

Proof. In an optimal schedule, the total penalty cost for tardiness is observed to be zero by sequencing the in-house jobs in SPT order. Based on this observation, we will prove Theorem 3 by reducing Problem P to the weighted matching problem.

Construct a graph $\mathscr{G} = (\mathscr{L} \cup \mathscr{R}, \mathscr{E})$ such that

(i) $\mathscr{L} = \{a(l) \mid l = 1, 2, \ldots, n\}$ and $\mathscr{R} = \{b(r) \mid r = 1, 2, \ldots, n\}$

(ii) For each edge (i, j), $i \in \mathscr{L}$ and $j \in \mathscr{R}$

For $l = 1, 2, \ldots, n$ and $r = 1, 2, \ldots, n$, the weight of edge $(a(l), b(r))$ is calculated as

$$\max\{0, (q_r - o_l)\}. \tag{31}$$

The objective is to find the set of edges $\mathscr{E}^*$ whose total weight is maximized. Note that, if edge $(a(l), b(r)) \in \mathscr{E}^*$, then in an optimal schedule of Problem P, job l is outsourced, and due date d_r is canceled. The reduced weighted matching problem can be solved in $O(n^3)$ by the Hungarian method. □

Theorem 4. *Given* $\pi = (\pi(1), \ldots, \pi(h))$, *it is polynomially solvable to determine* $\theta = (\theta(1), \ldots, \theta(h))$ *to minimize* $z(\sigma)$ *for Problem P.*

Proof. We will prove Theorem 4 by reducing Problem P to the SP problem such that the number of nodes is bounded by n. Let $C_{\pi(i)}$ be the completion time of job $\pi(i)$. Let $N(0,0)$ and $N(h+1, n+1)$ be the source and sink nodes, respectively. Let $N(i,b)$ be the node indicating

(i) Which due dates in $\{d_1, d_2, \ldots, d_b\}$ have to be canceled

(ii) In-house job $\pi(i)$ is assigned to d_b

If $0 < i \leq h$ and $0 \leq b' < b \leq n$, then let $N(i-1, b')$ be connected to $N(i,b)$ with length

$$V_b\left(C_{\pi(i)}\right) - \sum_{g=b'+1}^{b-1} q_g. \tag{32}$$

If $b = n$ and $0 \leq i < h$, then let $N(i,b)$ be connected to the sink node with length ∞. If $i = h$, then let $N(i,b)$ be connected to the sink node with length

$$-\sum_{g=b+1}^{n} q_g. \tag{33}$$

The objective is to find the SP from the source to the sink node. In the reduced SP problem, the number of nodes is observed to be $O(n^2)$, and each node has $O(n)$ outgoing arcs, respectively. Since the graph of the reduced SP problem is acyclic, the reduced SP problem can be solved in $O(n^3)$ by the algorithm of Ahuja et al. [21]. □

5. Heuristics

In this section, due to the high complexity of the dynamic programming in Lemma 2, we present three heuristics. We conduct numerical experiments to evaluate each heuristic. Heuristic H1 is based on the heuristic of Gerstl and Mosheiov [9]. First, Heuristic H1 prioritizes the jobs to be outsourced and the due dates to be canceled in advance and constructs new schedules by deleting jobs and due dates in order of priority from the initial schedule. Then, Heuristic H1 selects the best schedule among the constructed schedules. The detailed description of Heuristic H1 is as follows.

5.1. Heuristic H1

Step 1. Obtain two sequences τ_1 and τ_2 by sequencing the jobs in the nonincreasing orders of $(o_j - p_j)$ and o_j/p_j, respectively, and a sequence ς by sequencing the due dates in the nonincreasing orders of $d_g q_g$.

Step 2. Set

$$\begin{aligned} \pi_{i,0} &= (1, 2, \ldots, n), \\ \theta_{i,0} &= (1, 2, \ldots, n), \\ \text{for } i &= 1, 2. \end{aligned} \tag{34}$$

Step 3. Construct

$$\mathscr{S}_1 = \left\{\left(n-k, \pi_{i,k}, \theta_{i,k}\right) \mid i = 1, 2,\ k = 1, 2, \ldots, n\right\}, \tag{35}$$

where $\pi_{i,k}$ and $\theta_{i,k}$ are the subsequences constructed from $\pi_{i,k-1}$ and $\theta_{i,k-1}$ by deleting job $\tau_i(k)$ and $\varsigma(k)$, respectively.

Step 4. Return σ^* such that

$$z(\sigma^*) = \min\left\{z(\sigma) \mid \sigma \in \mathscr{S}_1 \cup \left\{\left(n, \pi_{1,0}, \theta_{1,0}\right)\right\}\right\}. \tag{36}$$

Heuristic H1 can be improved by the property in Theorem 4. Instead of $\theta_{i,k}$ in Step 3 of Heuristic H1, Heuristic H2 assigns due dates according to the algorithm in the proof of Theorem 4. The detailed procedure is as follows.

5.2. Heuristic H2

Step 1. Obtain two sequences τ_1 and τ_2 by sequencing the jobs in the nonincreasing orders of $(o_j - p_j)$ and o_j/p_j, respectively.

Step 2. Set

$$\pi_{i,0} = (1, 2, \ldots, n), \quad \text{for } i = 1, 2. \tag{37}$$

Step 3. Construct

$$\mathcal{S}_2 = \left\{ \left(\pi_{i,k}, \theta_{i,k} \right) \middle| \, i = 1, 2, \, k = 1, 2, \ldots, n \right\}, \tag{38}$$

where $\pi_{i,k}$ is the subsequence from $\pi_{i,k-1}$ by deleting job $\tau_i(k)$ and $\theta_{i,k}$ is the sequence obtained by applying the approach in Theorem 4 when $\pi = \pi_{i,k}$.

Step 4. Return σ^* such that

$$z(\sigma^*) = \min\left\{ z((\sigma) \mid \sigma \in \mathcal{S}_2 \cup \left\{ \left(n, \pi_{1,0}, \theta_{1,0} \right) \right\} \right\}. \tag{39}$$

Unlike Heuristics H1 and H2, Heuristic H3 constructs new schedules by optimally deleting jobs and due dates from the initial schedule based on Theorem 4. Then, Heuristic H3 selects the best schedule among the constructed schedules. The detailed description of Heuristic H3 is as follows.

5.3. Heuristic H3

Step 1. Set $\sigma^* = (n, \pi, \theta)$ and $h = n$, where

$$\begin{aligned} \pi &= (1, 2, \ldots, n), \\ \theta &= (1, 2, \ldots, n). \end{aligned} \tag{40}$$

Step 2. Construct

$$\mathcal{S}_3 = \left\{ \left(h, \pi_j, \theta_g \right) \middle| \, j \in \mathcal{J}, \, g \in \mathcal{D} \right\}, \tag{41}$$

where π_j and θ_g are the subsequences constructed from π and θ by deleting job j and due date g, respectively.

Step 3. Find $\sigma^*_{j,g} = (h, \pi^*_{j,g}, \theta^*_{j,g})$ such that

$$z\left(\sigma^*_{j,g}\right) = \min\{ z(\sigma) \mid \sigma \in \mathcal{S}_3 \}. \tag{42}$$

Step 4. Set $h = h - 1$, $\pi = \pi^*_{j,g}$, $\theta = \theta^*_{j,g}$, $\mathcal{J} = \mathcal{J} \backslash \{j\}$ and $\mathcal{D} = \mathcal{D} \backslash \{g\}$.

Step 5. If $z(\sigma^*_{j,g}) < z(\sigma^*)$, then set $\sigma^* = \sigma^*_{j,g}$.

Step 6. If $\mathcal{J} = \emptyset$, then return σ^* and STOP; otherwise, go to Step 2.

6. Experimental Results

To evaluate the performance of Heuristics H1–H3, we conducted numerical experiments with randomly generated instances in various settings. We implemented the heuristics with Python language. To implement the pseudopolynomial time algorithm introduced in Lemma 2, the Python NetworkX package was used. All our experiments were performed on a laptop computer with 32 GB of RAM and a 4.00 GHz processor.

6.1. Instances. All instances were categorized with respect to the number of jobs (n) and the *densities* of the generalized due dates (α) as defined by Gerstl and Mosheiov [9]. More specifically, we considered $n = 20, 40, 60$, and 80 jobs. The job processing times and outsourcing costs were randomly generated in the intervals $[1, 40]$ and $[1, 20]$, respectively. The generalized due dates were generated uniformly from the interval $[1, d_n]$, where $d_n = \alpha \sum_{j=1}^{n} p_j$. We considered the cases $\alpha = 0.2, 0.4, 0.6, 0.8$, and 1.0. The profits for canceling due dates were randomly generated in the interval $[1, 40]$. Finally, for the tardiness penalty cost $V_g(x)$ introduced in Section 1, $v_{i,g} = q_g \times \beta^i$, $\beta = 1.3, 1.5, 1.7, 2.0$, and δ is the standard deviation of the job processing times. We generated 30 instances for each (n, α, β).

6.2. Results. To evaluate the quality of the solutions obtained by Heuristics H1–H3, we compare these three solutions with the optimal solution, which is obtained by the pseudopolynomial time algorithm in Lemma 2. For instances with $n \geq 40$; however, the optimal solutions are not available due to lack of computer memory.

Table 1 shows the objective values of the schedules by applying Heuristics H1–H3 and the optimal algorithm for the instances under the setting $(n, \alpha, \beta) = (20, 0.8, 1.5)$. H3 outperforms the other heuristics in all instances, and in particular, finds optimal solutions for 80% (= 24/30) of instances. Thus, Heuristic H3 is the best of the three heuristics.

Table 2 compares the computation times of the three heuristics with the optimal algorithm. Heuristic H1 is the fastest, since it simply sorts the jobs and due dates according to predefined measures. Heuristic H3 ranks second, though it requires repetitive enumerations. On the contrary, although Heuristic H2 is similar to Heuristic H1, it exhibits longer computation times than Heuristic H1 since the SP problem needs to be solved as a subroutine.

According to the results in Tables 1 and 2, Heuristic H3 could be the most competitive alternative among the optimal algorithm and heuristics. Indeed, a similar conclusion can be drawn for $(n = 20, \alpha = 0.8)$ setting with different β values.

The effect of the densities of generalized due dates (controlled by α in our experiments) is also reported in Table 3. Interestingly, Heuristic H3 finds the optimal solutions for all 30 instances when $\alpha = 1.0$.

In Table 4, we investigate the effects of β. Note that Heuristic H1 does not consider β in its implementation and thus finds the same solution, regardless of β. On the contrary, Heuristics H2 and H3 consider β and thus find different solutions for different β values.

Table 5 demonstrates the average objective values for each setting of n and α. We found that the performances are consistent with the previous results.

Finally, we report the increases in computation times with respect to the number of jobs. Table 6 shows the averages and standard deviations of the computation times for the instances with $(\alpha, \beta) = (0.8, 1.5)$. Heuristic H1 finds solutions fastest, with little variation, since it simply sorts

Table 1: Objective values for instances with $(n, \alpha, \beta) = (20, 0.8, 1.5)$.

No.	OPT	H1	H2	H3
1	−90	−22	−76	−88
2	−61	−1	−56	−61
3	−71	−13	−54	−71
4	−81	+12	−71	−81
5	−97	−15	−95	−97
6	−127	−46	−118	−127
7	−65	−2	−48	−65
8	−93	−19	−86	−93
9	−76	−16	−71	−74
10	−85	−20	−70	−85
11	−140	−74	−125	−140
12	−104	−54	−93	−95
13	−98	−38	−88	−98
14	−118	−31	−90	−118
15	−66	−2	−59	−66
16	−81	−12	−72	−81
17	−96	−35	−87	−96
18	−92	−24	−81	−88
19	−118	−31	−90	−118
20	−66	−2	−59	−66
21	−101	−7	−91	−100
22	−109	−55	−98	−109
23	−99	−38	−96	−99
24	−73	−20	−62	−67
25	−82	−47	−72	−82
26	−94	−7	−75	−94
27	−103	−23	−100	−103
28	−101	−11	−89	−101
29	−109	−16	−100	−109
30	−96	−19	−90	−96

Table 2: Average and standard deviation of computation times for instances with $(n, \alpha, \beta) = (20, 0.8, 1.5)$.

	OPT	H1	H2	H3
Avg. times	475.589 s	0.002 s	0.723 s	0.066 s
Std.	69.201 s	0.001 s	0.284 s	0.085 s

Table 3: Average objective values for instances with $(n, \beta) = (20, 1.5)$ and different α's.

α	OPT	H1	H2	H3
0.2	−68.104	−5.967	−63.175	−66.471
0.4	−83.550	−5.900	−73.117	−80.575
0.6	−89.683	−16.067	−80.1	−87.267
0.8	−93.233	−23.100	−82.867	−92.133
1.0	−92.500	−22.933	−80.400	−92.500

Table 4: Average objective values for instances with $(n, \alpha) = (20, 0.2)$ and different βs.

β	OPT	H1	H2	H3
1.3	−68.956	−5.967	−63.696	−67.716
1.5	−68.104	−5.967	−63.175	−66.471
1.7	−65.744	−5.967	−62.949	−64.591
2.0	−67.133	−5.967	−62.800	−63.433

jobs. On the contrary, the computation times of Heuristic H2 show steep increases as the number of jobs increases since it uses an SP path subroutine. The computation time of Heuristic H3 lies between those of H1 and H2.

7. Concluding Remarks

Information sharing among parties in a supply chain provides an opportunity for members of the supply chain to

Table 5: Average objective values for instances with $\beta = 1.5$ and different n and α values.

α	$n = 40$			$n = 60$			$n = 80$		
	H1	H2	H3	H1	H2	H3	H1	H2	H3
0.2	−7.200	−134.375	−139.500	+16.400	−191.440	−200.026	−4.183	−253.879	−281.258
0.4	−20.283	−157.508	−168.700	−26.867	−238.733	−253.233	−55.900	−335.833	−361.267
0.6	−15.467	−153.633	−168.733	−30.267	−237.233	−267.333	−53.750	−320.200	−364.500
0.8	−36.100	−170.033	−194.733	−49.633	−241.667	−283.733	−74.500	−337.800	−382.433
1.0	−35.133	−160.033	−185.033	−46.667	−252.433	−290.833	−57.033	−320.700	−373.533

Table 6: Averages and standard deviations of the computation times for instances with $(\alpha, \beta) = (0.8, 1.5)$ and different n values.

n	H1		H2		H3	
	Avg.	Std.	Avg.	Std.	Avg.	Std.
20	0.002 s	0.001 s	0.723 s	0.284 s	0.066 s	0.085 s
40	0.006 s	0.002 s	11.468 s	0.792 s	1.014 s	0.110 s
60	0.011 s	0.002 s	54.759 s	2.740 s	5.664 s	0.341 s
80	0.016 s	0.000 s	110.108 s	3.014 s	13.913 s	1.109 s

coordinate with each other. In this paper, we consider an alignment problem of production and transportation schedules that can be modeled as a single-machine scheduling problem with generalized due dates and outsourcing options. Tardiness and outsourcing costs occur because of in-house and outsourced jobs, respectively, and cost savings arise from canceled due dates. The objective is to minimize the total outsourcing and canceling costs minus the total benefit from cost savings. We showed that the problem is NP-hard, even if each step function has only two intervals, and we developed a pseudopolynomial time algorithm. Due to the high complexity of the pseudopolynomial time algorithm, however, we developed three heuristics and verified their performance by conducting numerical experiments in various settings. These findings can be used to design and implement production schedules considering the alignment of production and transportation to downstream firms.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the 2019 Research Fund of Myongji University.

References

[1] H. L. Lee and S. Whang, "Information sharing in a supply chain," *International Journal of Manufacturing Technology and Management*, vol. 1, no. 1, pp. 79–93, 2000.

[2] H. Zhou Jr. and W. C. Benton, "Supply chain practice and information sharing," *Journal of Operations Management*, vol. 25, no. 6, pp. 1348–1365, 2007.

[3] G. P. Cachon and P. T. Harker, "Competition and outsourcing with scale economies," *Management Science*, vol. 48, no. 10, pp. 1314–1333, 2002.

[4] M. Webster, C. Alder, and A. P. Muhlemann, "Subcontracting within the supply chain for electronics assembly manufacture," *International Journal of Operations & Production Management*, vol. 17, no. 9, pp. 827–841, 1997.

[5] M. Haoues, M. Dahane, and N. K. Mouss, "Outsourcing optimization in two-echelon supply chain network under integrated production-maintenance constraints," *Journal of Intelligent Manufacturing*, vol. 30, no. 2, pp. 701–725, 2019.

[6] B. Liao, Q. Song, J. Pei, S. Yang, and P. M. Pardalos, "Parallel-machine group scheduling with inclusive processing set restrictions, outsourcing option and serial-batching under the effect of step-deterioration," *Journal of Global Optimization*, 2018.

[7] H. Safarzadeh and F. Kianfar, "Job shop scheduling with the option of jobs outsourcing," *International Journal of Production Research*, vol. 57, no. 10, pp. 3255–3272, 2019.

[8] N. G. Hall, "Scheduling problems with generalized due dates," *IIE Transactions*, vol. 18, no. 2, pp. 220–222, 1986.

[9] E. Gerstl and G. Mosheiov, "Single machine scheduling problems with generalised due-dates and job-rejection," *International Journal of Production Research*, vol. 55, no. 11, pp. 3164–3172, 2017.

[10] Y. C. E. Lee, C. K. Chan, A. Langevin, and H. W. J. Lee, "Integrated inventory-transportation model by synchronizing delivery and production cycles," *Transportation Research Part E: Logistics and Transportation Review*, vol. 91, pp. 68–89, 2016.

[11] N. G. Hall, S. P. Sethi, and C. Sriskandarajah, "On the complexity of generalized due date scheduling problems," *European Journal of Operational Research*, vol. 51, no. 1, pp. 100–109, 1991.

[12] C. Srikandarajah, "A note on the generalized due dates scheduling problem," *Naval Research Logistics*, vol. 37, no. 4, pp. 587–597, 1990.

[13] J. J. Yuan, "The NP-hardness of the single machine common due date weighted tardiness problem," *Systems Science and Mathematical Science*, vol. 5, no. 4, pp. 328–333, 1992.

[14] Y. Gao and J. Yuan, "Unary NP-hardness of minimizing total weighted tardiness with generalized due dates," *Operations Research Letters*, vol. 44, no. 1, pp. 92–95, 2016.

[15] G. Mosheiov and D. Oron, “A note on the SPT heuristic for solving scheduling problems with generalized due dates,” *Computers & Operations Research*, vol. 31, no. 5, pp. 645–655, 2004.

[16] B.-C. Choi and M.-J. Park, “Just-in-time scheduling with generalized due dates and identical due date intervals,” *Asia-Pacific Journal of Operational Research*, vol. 35, no. 6, Article ID 1850046, 2018.

[17] S. N. Chaurasia, S. Sundar, and A. Singh, “Hybrid metaheuristic approaches for the single machine total stepwise tardiness problem with release dates,” *Operational Research*, vol. 17, no. 1, pp. 275–295, 2017.

[18] J. Curry and B. Peters, “Rescheduling parallel machines with stepwise increasing tardiness and machine assignment stability objectives,” *International Journal of Production Research*, vol. 43, no. 15, pp. 3231–3246, 2005.

[19] B. Detienne, S. Dauzère-Pérès, and C. Yugma, “Scheduling jobs on parallel machines to minimize a regular step total cost function,” *Journal of Scheduling*, vol. 14, no. 6, pp. 523–538, 2011.

[20] C.-T. Tseng and K.-H. Chen, “An electromagnetism-like mechanism for the single machine total stepwise tardiness problem with release dates,” *Engineering Optimization*, vol. 45, no. 12, pp. 1431–1448, 2013.

[21] R. K. Ahuja, K. Mehlhorn, J. Orlin, and R. E. Tarjan, “Faster algorithms for the shortest path problem,” *Journal of the ACM*, vol. 37, no. 2, pp. 213–223, 1990.