

Article

# Innovative Production Scheduling with Customer Satisfaction Based Measurement for the Sustainability of Manufacturing Firms

Sang-Oh Shim <sup>1</sup> , KyungBae Park <sup>2,\*</sup>  and SungYong Choi <sup>3</sup>

<sup>1</sup> Department of Business Administration and Accounting, Hanbat National University, 125 Dongseo-daero, Yuseong-gu, Daejeon 34158, Korea; soshim@hanbat.ac.kr

<sup>2</sup> Department of Business Administration, Sangji University, 83 Sangjidae-gil, Wonju, Gangwon-do 26339, Korea

<sup>3</sup> Division of Business Administration, Yonsei University, 1 Yonseidae-gil, Wonju, Gangwon-do 26493, Korea; sungyong.choi@yonsei.ac.kr

\* Correspondence: kbpark@sangji.ac.kr; Tel.: +82-10-2896-0745

Received: 1 September 2017; Accepted: 28 November 2017; Published: 5 December 2017

**Abstract:** Scheduling problems for the sustainability of manufacturing firms in the era of the fourth industrial revolution is addressed in this research. In terms of open innovation, innovative production scheduling can be defined as scheduling using big data, cyber-physical systems, internet of things, cloud computing, mobile network, and so on. In this environment, one of the most important things is to develop an innovative scheduling algorithm for the sustainability of manufacturing firms. In this research, a flexible flowshop scheduling problem is considered with the properties of sequence-dependent setup and different process plans for jobs. In a flexible flowshop, there are serial workstations with multiple pieces of equipment that are able to process multiple lots simultaneously. Since the scheduling in this workshop is known to be extremely difficult, it is important to devise an efficient and effective scheduling algorithm. In this research, a heuristic algorithm is proposed based on a few dispatching rules and economic lot size model with the objective of minimizing total tardiness of orders. For the purposes of performance evaluation, a simulation study is conducted on randomly generated problem instances. The results imply that our proposed method outperforms the existing ones, and greatly enhances the sustainability of manufacturing firms.

**Keywords:** fourth industrial revolution; industry 4.0; scheduling; due date based measure; flexible flowshop; heuristics; sustainability of operation system

---

## 1. Introduction

Currently, in the era of the fourth industrial revolution, or industry 4.0, for sustainability of manufacturing firms and operation systems, it is necessary to make products in a timely manner in order to fulfill customers' demands. Therefore, open innovation, sustainability of operation management, and supply value chain become important [1–3]; this is referred to as intelligent production, or a smart factory. For such facilities, it is very important to develop an effective and efficient planning and scheduling algorithm with the objective of maximizing profits and minimizing costs [4,5]. The study considered here addresses a scheduling problem with customer satisfaction-based measurements.

In terms of the fourth industrial revolution, or industry 4.0, the issue of innovative production scheduling has recently been raised in Korea, and is defined as production scheduling that uses big data, cyber-physical systems (CPS), internet of things, cloud computing and mobile networks, and so on in its system of operations [6,7]. Within this environment, one of the most important things is to

develop an efficient and effective scheduling algorithm for enhancement of quality, cost, delivery and flexibility. Subsequently, it would be possible to implement that algorithm for manufacturing applications at the operational level in many types of facilities [8,9]. In this sense, mathematical models can make significant contributions to the literature on open innovation and the fourth industrial revolution, fulfilling the above-mentioned aims in an optimal way.

There are various types of workstation, each utilized according to the physical flows of the manufacturing lot or product; one of these is the flexible flowshop (also known as hybrid flowshop). A flexible flowshop comprises serial workstations with parallel equipment that are able to process multiple lots simultaneously. Currently, flexible flowshops can be found in many modern manufacturing systems in the electronics and machinery industries, including in the production of semiconductors, printed circuit boards, liquid crystal displays (LCDs), steel, solar cells, paper, aircraft, and so on. Therefore, it is important to schedule jobs/products effectively and efficiently, in order to enhance quality, cost, delivery, and flexibility, and to improve the sustainability of manufacturing firms.

In this study, a flexible flowshop scheduling problem is considered with the properties of a sequence-dependent setup and different process plans for jobs. Here, the sequence-dependent setup implies that the time required to prepare processing lots between different jobs is not the same. Also, different process plans for jobs means that each job has different process routing. Here, a job or order is composed of several lots, which are the processing and transportation units in manufacturing systems. In this research, a methodology based on dispatching rules and economic lot size models is suggested, and a simulation study is conducted numerically based on randomly generated problem instances for performance evaluation.

In order to solve the problem considered here, it is necessary to determine the scheduled job and its size of the lot to be processed on the equipment, which has been done in a similar way in other scheduling research [10]. In the proposed method, when equipment becomes idle, a job to be processed is selected according to its priority, which is determined by the rules devised here, including due dates and processing time, as well as the setup time for the jobs. Then, the methodology based on the optimized production technology is applied to schedule the selected lots with the objective of minimizing the total tardiness of the orders/jobs. Dispatching rules are used to select the job to be processed according to its priority, and optimized production technology is used to determine the size of the jobs—i.e., lot size—and provides a bottleneck-focused scheduling algorithm. Here, tardiness is defined as the difference between the completion time and the due date of the job.

To evaluate the performance of the suggested methodology, it is compared with existing methodologies currently used in practical fields. A simulation study is conducted on randomly generated instances in such a way that the real production situation and its data are plausible. The results show that the proposed method outperforms the existing ones.

The remaining parts of the paper are organized as follows. In Section 2, previous studies in the literature are reviewed; the problem formulation is described in more detail in Section 3. Then, a heuristic algorithm is suggested in Section 4. A computational simulation study is performed based on randomly generated problem instances, and the performance results of the proposed algorithms are given in Section 5. Finally, Section 6 concludes the paper with recommendations for future studies.

## 2. Literature Review

Currently, discussions are being raised about open innovation and sustainability at the operational level of the manufacturing hierarchy, since the automation of manufacturing systems is becoming more important for mass customization in the era of industry 4.0 [1–3]. One thing that is required, is to be able to operate sustainability-oriented scheduling systems at the operational level, and develop their algorithms in order to meet the desired requirements in terms of quality, cost, delivery, and flexibility [4,5,8,9]. Yuce et al. [4] considered earliness and tardiness measures, which reflect customer-satisfaction levels for single-machine scheduling problems. Additionally, Shim and Park [5] presented a scheduling algorithm for parallel machines, and Tong et al. [8] provided a scheduling algorithm aimed at saving energy. On the

other hand, Zhang [9] addressed a scheduling problem that dealt with aspects of machine utilization and pollution in dyeing processes.

The current automation of manufacturing systems, usually denoted as the smart factory, comprises big-data systems, cyber-physical systems (CPSs), internet of things (IoT), cloud computing, and mobile networks. Using these components can help to increase the capacity for product customization, mass production, flexibility, and quality of products [6,7]. Mourtzis et al. [11,12] presented an efficient operational policy and application for scheduling systems by using mobile networks, and devised a tool- and machine-scheduling algorithm based on cloud computing [13].

As discussed above, in addition to big-data, CPS, IoT, cloud computing, and mobile networks, intelligent production scheduling is necessary, since its results influence quantity, quality, customer satisfaction, resources, energy, emissions, and so on. Therefore, a technology for the production scheduling of jobs in manufacturing systems is addressed in this paper.

In the context of the flexible flowshop (also known as hybrid flowshop) scheduling problem considered here, since this kind of problem is known to be NP-hard, many papers have developed a number of heuristic algorithms aimed at finding a near-optimal solution [14,15]. For example, Gupta and Tunc [14] developed several heuristic algorithms to find a near-optimal schedule for a case in which the first stage contains only one machine, and the second stage contains  $m$  identical parallel machines. On the other hand, Guinet and Solomon [15] considered a several-stage hybrid flowshop scheduling problem in order to minimize the maximum tardiness or completion time.

In this paper, the minimization of the total tardiness of jobs is used as a performance measure. Even though there are lots of performance measures, including completion time and flow time, in order to maximize customer satisfaction by meeting the due dates of customers' orders, many researchers have used the tardiness measurement [4,15,16]. In addition, there are some functional constraints that reflect the real job/process-related characteristics of the problem descriptions, which will be explored in Section 3.

The flexible (hybrid) flowshop consists of a several (consecutive) production workstations, with each workstation having several parallel pieces of equipment that can process lots/operations simultaneously. Additionally, each lot is processed on only one piece of equipment in each workstation; then, the completed lot is transported to the next workstation for the next operation. Hybrid flowshops can be found in many manufacturing firms, especially in fields related to state-of-the-art technology. Therefore, it is very important to develop an efficient and effective scheduling methodology for use in intelligent automated manufacturing systems. Usually, when the equipment becomes idle, a lot should be chosen to be processed from among the waiting lots. In the flexible flowshop considered here, several dispatching rules for selecting the jobs are devised by using previous results [17].

Also, in this problem, a setup operation is also considered, in which it is necessary to change the recipe of the lot between different jobs. If the time required for the setup operation—i.e., setup time—is relatively long compared with the processing time, opportunity/setup cost may be incurred. Hence, to reduce the number of setup operations, it is necessary to process lots belonging to the same type consecutively. However, if lots of same type of job are processed consecutively, other types of job may be delayed, and holding costs may be increased. That is, there is a trade-off between holding and setup cost. Consequently, the determination of size of lot with the same type job is important in this research, and to decide the size of lot, the economic lot-sizing model is adopted [18–20].

Since there are several workstations on which jobs/lots are processed consecutively, and these usually have different numbers of equipment and specifications, the speed of production flow might not be same on each workstation. Hence, there may be bottleneck in production. In the past, it was known that it was very important to schedule jobs at the bottleneck workstation, since the capacity for production quantity and lead time is dominated by the bottleneck. Consequently, the concept of optimized production technology is used for this research [21,22].

In the proposed method, when a piece of equipment becomes idle, a job to be processed is selected according to its priority, which is determined by the method devised here on the basis of information about due dates, processing time and setup time. For the selected job, the size of the lots are determined

by using an economic lot-size model, and then bottleneck-focused scheduling is conducted by the optimized production technology, using the construction method.

### 3. Problem Description

The problem considered here is based on a real manufacturing system for printed circuit board (PCB), which is a typical one for flexible flowshops. In the flexible (hybrid) flowshop, there are  $n$  independent jobs ( $i = 1, \dots, n$ ) that need to be processed at each stage. Furthermore, there are  $k$  serial workstations ( $j = 1, \dots, k$ ). In each workstation, there are  $m_j$  parallel pieces of equipment, making it possible to process multiple jobs simultaneously ( $e = 1, \dots, m_j$ ). This can be referred to as a  $k$ -stage flexible (hybrid) flowshop. Each job has a processing time for each stage,  $p_{ij}$ , a due date,  $d_i$ , and a sequence-dependent setup time,  $s_{ih}$ , between jobs  $i$  and  $h$ .

As stated earlier, since a job is composed of several lots, which comprise the processing and transportation units in the manufacturing system, the size of the job implies the number of lots, i.e.,  $l_i$ . Note that  $p_i$  is the time for one lot of job  $i$  to be processed. In terms of the objective of minimizing the total tardiness of jobs, the tardiness of job  $i$ ,  $T_i$ , can be defined as the difference between the completion time of the job and its due date, i.e.,  $T_i = \max(0, C_i - d_i)$ , where  $C_i$  can be defined as the completion time of the last completed lot from job  $i$ . In this study, it is assumed that (i) no lot can be preempted; (ii) there is no breakdown of the equipment; (iii) all lots are available at time zero; (iv) processing times, due dates and sequence-dependent setup times of jobs are given; (v) bottleneck workstations (stages) are known; and (vi) the number of lots of jobs, the number of stages, and the number of pieces of equipment on each stage are known.

To describe the problem clearly, a mathematical model, e.g., using mixed-integer programming, can be formulated as follows. First, the notations are summarized below.

#### Parameters

- $k$  number of stages
- $n$  number of jobs
- $m_j$  number of pieces of equipment on stage  $j$  ( $j = 1, \dots, k$ )
- $l_i$  number of lots associated with job  $i$ , i.e., size of job  $i$  ( $i = 1, \dots, n$ )
- $e$  index of equipment on stage  $j$  ( $e = 1, \dots, m_j$ )
- $f$  index of lot of job  $i$  ( $f = 1, \dots, l_i$ )
- $p_{ij}$  processing time of one lot associated with job  $i$  on stage  $j$  ( $i = 1, \dots, n$  and  $j = 1, \dots, k$ )
- $d_i$  due date of job  $i$  ( $i = 1, \dots, n$ )
- $s_{ih}$  setup time between lots associated with job  $i$  and  $h$  ( $i = 1, \dots, n$  and  $h = 1, \dots, n$ )

Note that  $s_{ii}$  is zero.

- $M$  sufficiently large number

#### Decision variables

- $x_{ihejf}$  =1 if lot  $f$  associated with job  $h$  is processed directly after a lot associated with job  $i$  on equipment  $e$  on stage  $j$ , and 0 otherwise
- $x_{ohejf}$  =1 if lot  $f$  associated with job  $h$  is the first lot to be processed on equipment  $e$  on stage  $j$ , and 0 otherwise
- $x_{ioejf}$  =1 if lot  $f$  associated with job  $h$  is the last lot to be processed on equipment  $e$  on stage  $j$ , and 0 otherwise
- $c_{ijf}$  completion time of lot  $f$  associated with job  $i$  on stage  $j$
- $C_i$  time of the last completed lot associated with job  $i$  on stage  $k$ , i.e., the last stage
- $T_i$  tardiness of job  $i$ ; that is,  $T_i$  is  $\max(0, C_i - d_i)$

Based on the results of Guinet and Solomon [15], a mathematical model is derived as follows:

Minimize  $\sum T_i$   
 subject to

$$\sum_{e=1}^{m_j} \sum_{i=0}^n x_{ihejf} = 1 \text{ for all } h, j, f \text{ and } i \neq h \quad (1)$$

$$\sum_{i=0}^n x_{ohejf} = 1 \text{ for all } e, f \text{ and } j \quad (2)$$

$$\sum_{i=0}^n x_{ihejf} - \sum_{i=0}^n x_{thejf} = 0 \text{ for all } t, e, f \text{ and } j \quad (3)$$

$$i \neq t \quad i \neq t$$

$$c_{ijf} \geq c_{ij[f-1]} + p_{ij} + M \cdot \left( \sum_{e=1}^{m_j} x_{ihejf} - 1 \right) \text{ for all } i, f \text{ and } j \quad (4)$$

$$c_{ijf} \geq c_{hj[f-1]} + p_{ij} + s_{hi} + M \cdot \left( \sum_{e=1}^{m_j} x_{ihejf} - 1 \right) \text{ for all } i, f \text{ and } j \quad (5)$$

$$c_{ijf} \geq c_{i[j-1]f} + p_{ij} \text{ for all } f \text{ and } j \quad (6)$$

$$C_i \geq c_{ijf} \text{ for all } i, j \text{ and } f \quad (7)$$

$$T_i \geq C_i - d_i \text{ for all } i \quad (8)$$

$$T_i \geq 0 \text{ for all } i \quad (9)$$

$$x_{ihejf} \in \{0, 1\} \text{ for all } i, h, e, j \text{ and } f \quad (10)$$

$$c_{ijf} \geq 0 \text{ for all } i, j \text{ and } f \quad (11)$$

The objective function is to minimize total tardiness of jobs, where the tardiness of each job is specified in Constraints (8) and (9). Constraint (1) shows that each lot from a job should be processed once on each stage. On the other hand, Constraint (2) ensures that each piece of equipment can process, at most, one lot at a time. Constraint (3) specifies that each lot has a predecessor and successor on the equipment. Constraints (4) and (5) state the completion time of each lot on stage  $j$  according to the setup requirement. Furthermore, Constraint (6) shows that the completion time on the current stage should be more than that of the previous stage. Constraint (7) defines the completion time of job  $i$ . Finally, Constraints (10) and (11) state the binary and non-negativity conditions of the decision variables, respectively.

The problem considered here has three decision variables, with the properties of a sequence-dependent setup, and different process plans for jobs. They are: (i) jobs to be processed on the bottleneck workstation when a piece of equipment becomes idle; (ii) lot sizes of the selected jobs to be processed consecutively without setup operation; and (iii) schedule the selected jobs on non-bottleneck workstations.

#### 4. Technology for Production Scheduling

As described above, to minimize the tardiness of orders, this study addresses the hybrid flowshop scheduling problem while considering setup operations between different types of orders in order to determine appropriate lot sizes to be processed consecutively. The proposed algorithm is constructed as described below.

##### 4.1. Developing the Initial Schedule by Using Dispatching Rules

When a piece of equipment becomes idle on a workstation, the priorities of all lots that are waiting to be processed on the workstation are computed by certain dispatching rules, and the lot with the highest priority is selected and processed. Since it does not require much time to compute the priorities, scheduling based on dispatching rules is good for real-time scheduling in manufacturing systems.

In this section, a dispatching algorithm based on several priority rules is suggested to determine the job by using order information, i.e., due dates, processing times, and setup times. The following are the dispatching rules used in this research.

FIFO first in first out: choose the lot with the lowest arrival time, i.e.,  $a_i$

- MNS** minimum number of setup: select the lot with the lowest value for sum of processing and setup time of lot, i.e.,  $p_{ij} + s_{[i-1]i}$  where  $p_{ij}$  is the processing time and setup time of lot  $i$  on the current workstation  $j$ , and  $s_{[i-1]i}$  is the sequence-dependent setup time between the immediate predecessor of job  $i$  and job  $i$ .
- MOD** traditional modified operation due date rule: choose the job with the maximum value between remaining time to the due date and expected time to complete. i.e.,  $\max(d_i - b(w_i - p_{ij}), t + w_i)$ , where  $d_i$ ,  $b$ ,  $w_i$  and  $t$  are the due date of job  $i$ , a parameter for the MOD rule, the sum of the remaining processing time including the current processing time, and the current time, respectively.
- MDD** traditional modified due date rule, choose the lot with the maximum value between due date and expected time to complete, i.e.,  $\max(d_i, t + w_i)$
- NMOD** new modified rule (developed here): the same as the MOD rule, except for the  $w_i$ , which is replaced with  $w_i'$ , where  $w_i'$  is the sum of remaining processing time divided by the number of pieces of equipment that can process job  $i$  on workstation  $j$ , i.e.,  $e_{ij}$ .
- NMDD** new modified due date rule (developed here): the same as the MDD rule, except for replacing  $w_i$  with  $w_i'$ .
- NSLCK** new slack rule: choose the lot with the least expected time to due date, i.e.,  $d_i - t - w_i'$

NMOD, NMDD and NSLCK rules take into consideration the fact that each workstation has multiple pieces of equipment, and are therefore able to process lots simultaneously. In addition, the MNS rule is able to select the lot with the least setup time. A job for idle equipment is selected according to the priority obtained by each rule, with these results being modified by the methods described below. See Kim [17] for more details about the traditional dispatching rules.

#### 4.2. Determining the Lot Size

In this section, the size of a lot to be processed is determined. In practical manufacturing firms, a lot is defined as the minimum unit of processing and transportation. Therefore, lot sizing determines the number and size of lots to be processed consecutively, without setup operation. Usually, the objective of a lot-sizing model is to minimize the setup and inventory holding cost, including production cost.

In this paper, since the tardiness cost (penalty) incurred by setup operation and flow time is addressed, three assumptions—i.e., (i) that a capacity constraint exists; (ii) that the planning horizon is finite; and (iii) that the demand represents the size of the order from customers—are added to the economic order quantity model, which is very widely known in the field of inventory management [18–20]. The following notations are used in the suggested model.

$BT$  set of bottleneck workstation

$D_i$  remaining quantity of lots associated with job  $i$ , i.e., size of job  $i$  to be processed

$\alpha$  weight parameter

The decision variable, the size of lot  $i(Q_i)$ , is obtained in such a way that the workstations are categorized as being either the bottleneck workstation or non-bottleneck workstations, since scheduling on the bottleneck is more important [21,22]. Using the total cost equation of the economic order quantity model, the following objective function is developed. It consists of two terms; the first and the second terms represent the expected flow time, and the expected time for which other jobs are delayed, respectively.

$$\text{Total Cost} = \alpha \left\{ \sum_{j \in BT} (s_{[i-1]i} + Q_i p_{ij}) + \sum_{j \notin BT} (s_{[i-1]i} + D_i p_{ij}) \right\} + \left( \frac{D_i - Q_i}{Q_i} \right) \sum_{j \in BT} (s_{[i-1]i} + Q_i p_{ij}) \quad (12)$$

The first term, i.e.,  $\sum_{j \in BT} (s_{[i-1]i} + Q_i p_{ij}) + \sum_{j \notin BT} (s_{[i-1]i} + D_i p_{ij})$  describes the increased flow time incurred by the selected lot of job  $i$  on the bottleneck and non-bottleneck workstations, respectively. That is, on the bottleneck, setup time and processing time for the determined lot size, i.e.,  $Q_i$ , is required. Similarly,

on the non-bottleneck workstations, setup time and processing time for the remaining quantity of lots, i.e.,  $D_i$ , is needed. Therefore, the first term is the expected tardiness penalty caused by the selected job  $i$ . On the other hand, the second term, the number of the expected setup operations for the remaining quantity, i.e.,  $\frac{D_i - Q_i}{Q_i}$ , multiplied by  $\sum_{j \in BT} (s_{[i-1]j} + Q_i p_{ij})$ , shows the increased tardiness penalty for other jobs on the bottleneck workstation. Note that there is no significant impact on the non-bottleneck stations.

If the size of lot ( $Q_i$ ) is small, the number of setup operation can be increased. Hence, the number of pieces of equipment processing the associated job might be increased. Since the setup time and processing time in the bottleneck workstation are usually larger than for other workstations, the increase in setup operation implies that the completion of jobs may be tardy.

The optimal  $Q_i$  for the objective function (12) can be obtained by differentiating it as follows. If the parameter ( $\alpha$ ) is 1, then  $Q_i$  cannot be obtained. Hence,  $\alpha$  is set to the values in which the denominator is not zero, based on the preliminary experiments.

$$Q_i = \sqrt{\frac{D_i \sum_{j \in BT} s_{[i-1]j}}{(\alpha - 1) \sum_{j \in BT} p_{ij}}} \quad (13)$$

From the results of Section 4.1 and this section, the job to be processed for each workstation, when equipment becomes available, is chosen by using the dispatching rules suggested in Section 4.1, and the lot size to be processed consecutively without setup operation is determined by using Equation (13). The detailed procedure for developing the schedule for the whole workstation is described in the following.

#### 4.3. Construction Method for Developing the Schedule

Since the scheduling problem for hybrid flow shops (HFS) consisting of several workstations is considered, it is necessary to schedule jobs for the whole workstation simultaneously. Since the scheduling for the bottleneck workstation is more important than that for non-bottleneck workstations, as discussed in the previous section, the concept of bottleneck-focused scheduling is used in this research [21,22]. In typical bottleneck-focused scheduling, the schedule for the bottleneck workstation is determined first, and then, on the workstations in front of bottleneck, backward scheduling is applied. Forward scheduling is applied for the workstations behind the bottleneck. The following is a detailed procedure for obtaining the complete schedule.

##### Phase 1. Scheduling for the bottleneck workstation

- (i) Gather information about all orders (processing and setup times, due dates, order quantity and so on) and all workstations (number of pieces of equipment and identification of the bottleneck workstation).
- (ii) On the bottleneck workstation, among all waiting jobs, select a job to be processed when a piece of equipment becomes idle by using the suggested dispatching rules in Section 4.1, and determine the lot size using Equation (13) in Section 4.2. If the current lot size of the selected job is smaller than the computed  $Q_i$ , it is set to the current size. Then, schedule the lot to the equipment. Note that the number of schedules obtained is the number of the suggested dispatching rules. Therefore, select the one with the lowest completion time among the schedules.
- (iii) Repeat procedure (ii) until all waiting jobs have been scheduled. Then, scheduling is complete for the bottleneck workstation.

##### Phase 2. Scheduling on the non-bottleneck workstations

- (iv) (Forward scheduling) Check whether or not there are successive non-bottleneck workstations. If these exist, determine the schedule (start time and completion time) of each lot on successive workstations by using the results of the bottleneck workstation. That is, since the completion time of the lots at the bottleneck workstation is the arrival time at the immediate successor, schedule

the lots in ascending order of arrival time, and determine the start time and completion time. Ties are broken by using the suggested dispatching rules in Section 4.1. Among the schedules, select the one with the lowest completion time.

- (v) Repeat forward scheduling until all lots have been scheduled at all successive workstations.
- (vi) (Backward scheduling) Check whether or not there are preceding non-bottleneck workstations. Determine the schedule of each lot at the preceding workstations by using the results of bottleneck workstation. That is, since the start time of the lots at the bottleneck workstation is the completion time of its immediate predecessor, schedule the lots backward in ascending order of completion time, and determine start time and completion time. Ties are broken by using the suggested dispatching rules in Section 4.1. Among the schedules, select the one with the lowest completion time.
- (vii) Repeat backward scheduling until all lots have been scheduled at all preceding workstations.
- (viii) Compute the total tardiness.

During the procedure of bottleneck-focused scheduling, note that the order in which forward and backward scheduling are conducted is arbitrary. Furthermore, if there are multiple bottleneck workstations in the hybrid flowshop, there may non-bottleneck workstations between bottlenecks that could be either successive or preceding. In this case, any non-bottleneck workstation can be selected as successive or preceding arbitrarily. For example, if there is a 5-stage flexible workstation, and 2nd and 4th stages are bottlenecks, then the 3rd stage could be either a preceding one for the 4th stage, or a successive one for the 2nd stage. In that case, after scheduling for the 2nd and 4th stages has been completed, the 3rd stage can be scheduled either as a successor of the 2nd or as a predecessor of the 4th, arbitrarily.

A flow chart showing the procedures of the suggested algorithm as described above is displayed in Figure 1. Let the devised algorithm denote *Heuristic algorithm based on Dispatching rules, Lot-sizing model and Bottleneck-focused scheduling* (HDLB).

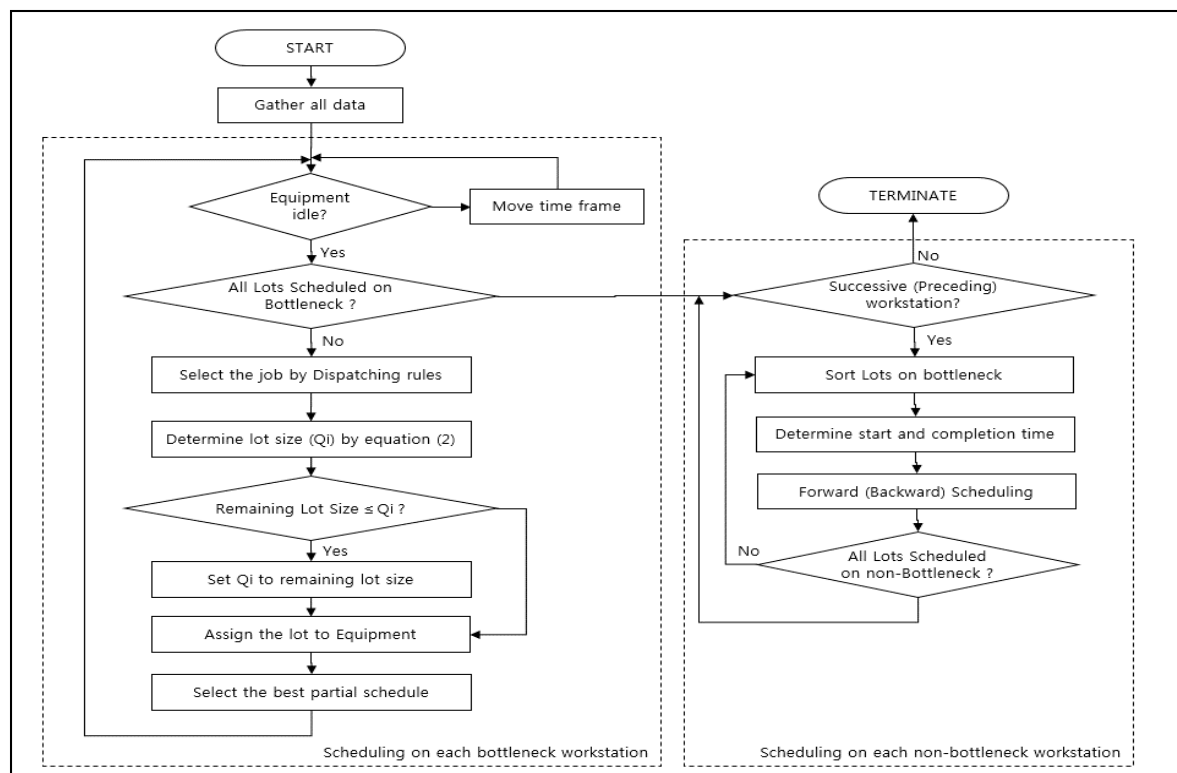


Figure 1. HDLB Flow Chart.



## 5. Computational Results

In this Section, a series of computational tests with randomly generated problem instances reflecting real manufacturing firms is performed to evaluate the performance of the proposed algorithm. For the experiment, 450 problems were generated; 50 instances each for the 9 selected combinations of three levels of the workstation numbers (3, 6 and 9), three levels of the number of jobs (10, 20 and 30), and three levels of the number of pieces of equipment on each workstation (3, 6 and 9). Other data were generated as follows. Here,  $DU(a,b)$  denotes discrete uniform distribution within the range (a, b).

- (1) The number of lots associated with a job is randomly chosen from  $DU(5, 10)$ .
- (2) The number of job types was set to the integer number generated from  $DU(0.5, 1)$ , multiplied by the number of jobs.
- (3) The processing time of a lot was generated from  $DU(5, 25)$ .
- (4) The due date of an order was set to the value from generated from  $DU(0.5, 3)$ , multiplied by total processing time of the order, i.e., sum of setup time and processing time for all workstations.
- (5) The setup time was set to the value generated from  $DU(0.5, 2)$ , multiplied by the processing time.
- (6) The number of bottleneck workstations was set to 1 when the number of workstation was 3, 2 when the number of workstations was 6, and 3 when the number of workstations was 9.
- (7) The parameter  $\alpha$  was set to 1.1 based on preliminary experiments.

The proposed algorithm was tested as follows. All dispatching rules developed in Section 4.1 were used to select a job to be processed when a piece of equipment becomes idle, and for each selected job, the lot size was determined by using the equation developed in Section 4.2. Then, bottleneck-focused scheduling, as described in Section 4.3, was performed, and the complete schedule for the flexible flowshop was obtained. During the procedure, the partial schedule with the lowest completion time on the current workstation was used, and the solution values, i.e., total tardiness of jobs, were computed through the simulation study.

The methods used for comparison are the existing ones, using only dispatching rules, i.e., first in first out (FIFO), earliest due date (EDD), shortest processing time (SPT) and longest processing time (LPT) rules, which are the most-preferred processes, due to managerial convenience when applied in real manufacturing systems. That is, every time a piece of equipment becomes idle, a job is selected by those dispatching rules (FIFO, EDD, SPT and LPT), and only one lot of the selected job is processed. Note that there is no procedure for determining the size of a lot. For all workstations, regardless of bottleneck and non-bottleneck, these procedures are repeated until all lots of all jobs are scheduled. In this paper, four schedules by four dispatching rules were obtained, and the total tardiness of the jobs was computed.

For the evaluation of the algorithms, i.e., HDLB, two performance measures are used. One is the number of best cases, in which each algorithm shows the best—i.e., minimum—total tardiness, and the other is the percentage reduction of the solutions of the algorithm compared to the solution based on the method currently used in practice. The PR is defined as  $100 \times (1 - T_H/T_p)$ , where  $T_H$  is the solution value (tardiness) obtained using the proposed method, i.e., HDLB.  $T_p$  is the best solution value obtained from the methods currently used in practice. That is, from among the four solution values obtained using the dispatching rules (FIFO, EDD, SPT and LPT), that for which the minimum tardiness was obtained is chosen as  $T_p$ . A simulation study was coded in the C programming language, and computational experiments were performed on a personal computer with a Pentium i3-4030U processor operating at 1.4 GHz clock speed.

Table 1 gives the overall results of the tests for the suggested algorithm. In all problem instances, there was significant improvement compared with the existing methods. First, in small-sized problems in which the number of jobs was less than 10 and the number of pieces of equipment was 6, the compared methods (FIFO, EDD, SPT and LPT) gave the best solutions for only a few problem instances. On the other hand, the proposed method obtained the best solutions in most

problem instances (more than 45 among 50). Additionally, since the proposed method showed better performance in terms of the measure of percentage reduction, it is implied that the total tardiness of jobs could be reduced by more than 12 percent compared to the existing methods. That is, if the HDLB were applied to a real manufacturing system, it would help increase customer satisfaction for due-date related measures, since meeting due dates is very important in the current era.

Moreover, as the size of the problem instance increased, it appears that the performance of the proposed method became much better than those of the methods used in real systems. For the all the mid-sized (20 jobs) and large-sized (30 jobs) problem instances, the proposed method outperformed the other methods. Additionally, the best percentage reduction was obtained for the larger-sized problems. Finally, the computation time for each problem instance was less than 0.001 s in the simulation study.

Even though optimal solution approaches such as branch and bound algorithms, dynamic programming and mixed-integer programming guarantee finding the optimal solution, it is very difficult to use them in the real world, since modeling is very complex, and spends a lot of time obtaining the optimal solution, while operators find it hard to understand them. However, the HDLB requires very little computation time to find a solution that is markedly better than those obtained by existing methods.

Consequently, the proposed algorithm, HDLB, shows better performance for practical use in terms of solution quality (total tardiness) and computation time (real-time scheduling), maximizing customer satisfaction.

**Table 1.** Overall results of the simulation study.

Number of Jobs	10			20			30		
Number of Equipments	3	6	9	3	6	9	3	6	9
3 *	12.58 <sup>†</sup> (44) <sup>††</sup>	15.41 (46)	17.35 (48)	15.69 (48)	18.64 (50)	21.34 (50)	19.86 (50)	23.66 (50)	27.81 (50)
6	14.87 (45)	16.44 (47)	18.66 (49)	17.98 (49)	20.04 (50)	22.31 (50)	22.06 (50)	27.86 (50)	30.46 (50)
9	16.32 (46)	19.56 (49)	21.98 (50)	19.55 (49)	21.31 (50)	23.55 (50)	25.34 (50)	28.64 (50)	32.67 (50)

<sup>†</sup> Average percentage reduction of HDLB compared to the result of practice. <sup>††</sup> Number of cases in which the heuristic algorithm found the best solution among 50 instances. \* Number of workstations.

## 6. Concluding Remarks

Since the needs and importance of sustainability and open innovation are growing all over the world in the era of the fourth industrial revolution, or industry 4.0 [23–25], innovative production at an operational level of manufacturing is becoming increasingly necessary and important. In particular, mass customization, delivery, flexibility and quality are becoming increasingly important for sustainable operation systems in manufacturing firms. A smart factory is one means for achieving such sustainable and innovative production scheduling, since the quantity, quality, and customer satisfaction are greatly affected by the scheduling results at an operational level. Therefore, effective and efficient scheduling is one of the most critical success factors for smart factories.

In this paper, a flexible flowshop scheduling problem with properties of sequence-dependent setup time and different process plans was considered, with the aim of minimizing the total tardiness of jobs. In the proposed algorithm, several dispatching rules are developed, including a lot-sizing model and bottleneck-focused scheduling. When the equipment on the bottleneck workstation becomes idle, a job is chosen according to its priority, as computed by the dispatching rules. Then, the lot size of the selected job to be processed consecutively without setup operation is determined by the economic lot-size model. Finally, the schedule for the remaining, non-bottleneck, workstations is performed using concepts of optimized production technology. Based on the computational experiments, the proposed algorithm showed better performance than the existing algorithms currently used in real manufacturing

systems. Additionally, since it requires a shorter computational time to obtain a solution, it can be argued that the proposed method is good for real-time scheduling systems.

In most previous studies in scheduling literature, a few traditional concepts have been used separately, including dispatching rules, lot-sizing models, and bottleneck-focused scheduling. However, in this study, an effort has been made to combine them all together, and it shows better results than existing approaches. There might be limitations to this work, since this scheduling problem is specified at a lower level of the production hierarchy in manufacturing firms. That is, since the proposed algorithm can only be applied in manufacturing systems with flexible flowshops, it might be worth considering various types of scheduling problem, such as re-entrant flexible flowshops. Furthermore, at a higher level, such as with respect to production strategies and policies of the company, one may need to devote effort to developing production plans to apply. With regard to problem setting, it may be necessary to test more extensive scenarios, based on various combinations of parameter values in the model, i.e., process plans, due dates and setup times, to verify the robustness of the proposed algorithm. Finally, a comparison with the optimal solution or lower bound may be needed to check the sub-optimality of the solution.

**Author Contributions:** All authors significantly contributed to the scientific study and writing. Sang-Oh Shim contributed to the overall idea, algorithms, and writing of the manuscript; KyungBae Park contributed to the detailed writing, ideas, and discussions on open innovation and sustainability of operation systems and manufacturing firms, as well as the preparation and publishing of the paper; SungYong Choi contributed to the detailed writing and concepts on the economic lot-sizing model.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Kleindorfer, P.R.; Singhal, K.; Wassenhove, L.N.V. Sustainable operations management. *Prod. Oper. Manag.* **2005**, *14*, 482–492. [[CrossRef](#)]
2. Giret, A.; Trentesaux, D.; Prabhu, V. Sustainability in manufacturing operations scheduling: A state of the art review. *J. Manuf. Syst.* **2015**, *37*, 126–140. [[CrossRef](#)]
3. Alsaffar, A.J.; Raoufi, K.; Kim, K.-Y.; Okudan Kremer, G.E.; Haapala, K.R. Simultaneous Consideration of Unit Manufacturing Processes and Supply Chain Activities for Reduction of Product Environmental and Social Impacts. *ASME J. Manuf. Sci. Eng.* **2016**, *138*, 101009. [[CrossRef](#)]
4. Yuce, B.; Fruggiero, F.; Packianather, M.S.; Fera, M. Hybrid Genetic Bees Algorithm applied to Single Machine Scheduling with Earliness and Tardiness Penalties. *Comput. Ind. Eng.* **2017**, *113*, 842–858. [[CrossRef](#)]
5. Shim, S.O.; Park, K. Technology for production scheduling of jobs for open innovation and sustainability with fixed processing property on parallel machines. *Sustainability* **2016**, *8*, 904. [[CrossRef](#)]
6. Lee, J.; Bagheri, B.; Kao, H.A. Recent advances and trends of cyber-physical systems and big data analytics in industrial informatics. *Int. Conf. Ind. Inform.* **2014**. [[CrossRef](#)]
7. Mourtzis, D.; Vlachou, E. Cloud-based Cyber-physical Systems and Quality of Services. *TQM J.* **2016**, *28*, 704–733. [[CrossRef](#)]
8. Tong, Y.; Li, J.; Li, S.; Li, D. Research on energy-saving production scheduling based on a clustering algorithm for a forging enterprise. *Sustainability* **2016**, *8*, 136. [[CrossRef](#)]
9. Zhang, R. Sustainable Scheduling of Cloth Production Processes by Multi-Objective Genetic Algorithm with Tabu-Enhanced Local Search. *Sustainability* **2017**, *9*, 1754. [[CrossRef](#)]
10. Karamyar, F.; Sadeghi, J.; Yazdi, M.M. A Benders decomposition for the location-allocation and scheduling model in a healthcare system regarding robust optimization. *Neural. Comput. Appl.* **2016**. [[CrossRef](#)]
11. Mourtzis, D.; Doukas, M.; Vlachou, E. A mobile application for knowledge-enriched short-term scheduling of complex products. *Logist. Res.* **2016**, *9*, 3. [[CrossRef](#)]
12. Mourtzis, D.; Vlachou, E.; Zogopoulos, V.; Fotini, X. Integrated production and maintenance scheduling through machine monitoring and augmented reality: An Industry 4.0 approach. In Proceedings of the APMS Conference 2017, Hamburg, Germany, 3–7 September 2017.
13. Mourtzis, D.; Vlachou, E.; Xanthopoulos, N.; Givehchi, M.; Wang, L. Cloud based adaptive process planning considering availability and capabilities of machine tools. *J. Manuf. Sys.* **2016**, *39*, 1–8. [[CrossRef](#)]

14. Gupta, J.N.D.; Tunc, E.A. Minimizing tardy jobs in a two-stage hybrid flowshop. *Int. J. Prod. Res.* **1998**, *36*, 2397–2417. [[CrossRef](#)]
15. Guinet, A.G.P.; Solomon, M.M. Scheduling hybrid flowshops to minimize maximum tardiness or maximum completion time. *Int. J. Prod. Res.* **1996**, *34*, 1643–1654. [[CrossRef](#)]
16. Chen, T.C.E.; Gupta, M.C. Survey of scheduling research involving due date determination decisions. *Eur. J. Oper. Res.* **1989**, *38*, 156–166. [[CrossRef](#)]
17. Kim, Y.D. A comparison of dispatching rules for job shops with multiple identical jobs and alternative routings. *Int. J. Prod. Res.* **1990**, *28*, 953–962. [[CrossRef](#)]
18. Khouja, M.; Michalewicz, Z.; Wilmot, M. The use of genetic algorithms to solve the economic lot size scheduling problem. *Eur. J. Oper. Res.* **1998**, *110*, 509–524. [[CrossRef](#)]
19. Drexel, A.; Kimms, A. Lot sizing and scheduling survey and extensions. *Eur. J. Oper. Res.* **1997**, *99*, 221–235. [[CrossRef](#)]
20. Miranda, S.; Marcello, F.; Iannone, R.; Riemma, S. A multi-item constrained EOQ calculation algorithm with exit condition: A comparative analysis. *IFAC PapersOnLine* **2015**, *48*, 1314–1319. [[CrossRef](#)]
21. Verma, R. Management science, theory of constraints/optimized production technology and local optimization. *Omega* **1997**, *25*, 189–200. [[CrossRef](#)]
22. Lee, E.Y. Due-Date Based Scheduling Methods in a Lead Frame Manufacturing System. Master's Thesis, Department of Industrial Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Korea, 2000.
23. Yun, J.J. How do we conquer the growth limits of capitalism? Schumpeterian Dynamics of Open Innovation. *J. Open Innov. Technol. Mark. Complex.* **2015**, *1*, 17. [[CrossRef](#)]
24. Prado, A.L.; da Costa, E.M.; Furlani, T.Z.; Yigitcanlar, T. Smartness that matters: Towards a comprehensive and human-centered characterization of smart cities. *J. Open Innov. Technol. Mark. Complex.* **2016**, *2*, 8. [[CrossRef](#)]
25. Yun, J.J.; Won, D.K.; Park, K.B. Dynamics from open innovation to evolutionary change. *J. Open Innov. Technol. Mark. Complex.* **2016**, *2*, 7. [[CrossRef](#)]



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).