

# A Two-Machine Flow Shop Batch Scheduling Model to Minimize Total Actual Flow Time

Rinto Yusriski<sup>1\*</sup>, Andri Rachmat Kumalasian Nasution<sup>1</sup>, Lukas<sup>2</sup>, Linda Wijayanti<sup>2</sup>,  
Sandra Octaviani<sup>2</sup>

<sup>1)</sup> Fakultas Teknologi Manufaktur Universitas Jenderal Achmad Yani,  
Jalan Terusan Jenderal Achmad Yani, Bandung 40285, Indonesia

<sup>2)</sup> Professional Engineer department Universitas Katolik Indonesia Atma Jaya, BSD, Lantai 6,  
Jl. Raya Cisauk Lapan, Sampora, Kec. Cisauk, Kabupaten Tangerang, Banten 15345, Indonesia  
Email: rinto.yusriski@lecture.unjani.ac.id

\*Corresponding author

---

**Abstract:** This study introduces a scheduling model for a two-machine flow shop batch system to minimize the actual flow time. In this system, two machines are responsible for processing raw materials and producing finished products, with a single bottleneck machine. The entity overseeing the manufacturing process organizes demand units into batches, ensures the accurate and timely arrival of raw materials, and delivers all finished products punctually to meet an expected due date. The study addresses crucial challenges, including determining the optimal number of batches, sizes, and sequences to achieve the specified objective. The analysis adopted an algorithm grounded in the Lagrange relaxation method to tackle these challenges. Moreover, the algorithm is operated by identifying the bottleneck machine as a scheduling reference and determining the appropriate number of batches and sizes. The analysis showed the efficacy of the developed algorithm by using Johnson's rule for making batch sequence decisions through numerical experiments conducted across 1000 cases. The results showed a 1.44% to 4.43% improvement in efficiency compared to previous research, accompanied by a 2 to 8 times reduction in computational time.

**Keywords:** Batch scheduling, two-machine flow shop, actual flow time.

---

## Introduction

Experts are deeply involved in addressing scheduling problems, as evidenced by the extensive collection of literature reviews over the past two decades [1]–[11]. Baker [12] classified scheduling problems based on machine types (single and multiple), flow patterns [Job Shop (JSS), Flow Shop (FSS), and Open Shop (OSS)], job arrival patterns (static and dynamic), and information patterns (deterministic and stochastic). Additionally, batch and job scheduling represent two distinct types of organizing problems.

The optimization of job sequences in flow shop has fascinated experts for an extended period, as evidenced by the substantial body of research over references [13]–[16]. The primary objective is to identify the most efficient sequence for executing jobs. Johnson [13] explores job problems in two and three-machines, proposing an optimal algorithm and establishing the NP-hard nature of flow shop problem with more than three-machine. Campbell *et al.* [14] presented a heuristic solution with more than three machines, while Brucker [15] stated that the optimal solution was achieved through the branch and bound method. Yusriski *et al.* [16] examined the dynamic arrival of job problems in the context of flow shops, complementing operational research with discussions on metaheuristics [17]–[19].

Beyond job organizing, batch scheduling problems extends into flow shop patterns, observed in industries such as apparel manufacturing, automotive component manufacturing, and the food industry, where identical orders are processed in batches to minimize make span [20]–[22]. Key decisions in these scenarios focus on defining the optimal number of batches, size, and the applicable batch scheduling rules. In the context of this study, particular attention is given to considering setup time when the machine initiates batch processing and ensuring the availability of raw materials at the scheduling period's outset ( $t = 0$ ). The strategy inventory minimization includes the delivering of finished items in stages, correlating with the completion of each batch on the last machine (batch completion time). However, real-world scenarios often require simultaneous delivery at the agreed-upon due date, assuming the company can manage the timely and sufficient arrival of raw materials at the first machine for inventory minimization.

Halim *et al.* [23] presented a model for single-machine batch scheduling that prioritizes minimizing the total actual flow time with respect to the due date. The total actual flow time, defined as the duration from raw material arrival to the due date, plays a crucial role in achieving on-time delivery and reducing inventory time. The research considered dividing raw materials arrival into batches, each with regulated sizes in quantity and time, based on the requirements of the production floor. Additionally, the analysis reiterated the importance of delivering finished product on time to correlate with the common due date. This metric, measuring the total actual flow time, has found widespread application in various batch scheduling problems, showing the effectiveness in not only ensuring on-time delivery but also minimizing inventory time [24]–[31].

An extension of the total actual flow time criterion is observed in batch scheduling problem within flow shop systems [32]–[34]. Halim and Ohta [32] examined the scheduling problem into two sub-problems, namely batch and scheduling. In batch sub-problem, decisions include determining the number of batches and batch size. The algorithmic commences by selecting a reference machine, using the decision solution as a guide for others. The reference machine is selected by comparing the total actual flow time performance using the single machine algorithm from research [23]. In the scheduling sub-problem, batch sequences are developed through semi-enumeration using the pairwise interchange method. While the solution in Halim and Ohta [32] does not guarantee to produce an optimal solution, it [32] has inspired and been developed for other flow shop problems, including in [33] and [34]. Suryadhini *et al.* [33] discussed batch scheduling problems assuming two product types in a two-stage flow shop, while Maulidya *et al.* [34] developed a two-stage flow shop model with continuous batch sizes [35]–[37].

The investigation into the research model by Halim and Ohta [32] specifically centers on discrete batch sizes and proposes an enhanced approach. The focus is on a two-machine flow shop, considering Johnson's [13] ability to produce optimal solutions for job scheduling in these settings. Scheduling research on two-machine flow shops has attracted attention, discussed in the context of job scheduling [38]–[41] and batch scheduling [20]–[22]. For validation in a real system, the study takes inspiration from manufacturing companies in the apparel industry. A single job with an order quantity of  $n$  units undergoes processing in the sewing and finishing stages, each assumed to be a machine. The machines process the job into batches, framing the problem as two-machine flow shop batch scheduling issue. The challenge faced by the company is to deliver products to consumers by the due date (minimize delivery time) while minimizing inventory buildup (raw materials, semi-finished goods, and finished goods). The company practice of bringing fabric materials to the production floor before the scheduling period ( $t = 0$ ) results in accumulating raw materials, semi-finished products, and finished products. This practice is modified by introducing raw materials in stages based on the production floor needs. The key decisions include determining the frequency of raw materials imports (number of batches), the quantity in each batch (batch size), and priority rules for scheduling batches to ensure timely order delivery with minimal waiting time. Consequently, the total actual flow time criterion is applied as a performance measure for scheduling.

## Methods

### Problem Formulation and Mathematical Models

Halim and Ohta [32] proposed a heuristic procedure for the single-item flow shop batch scheduling problem with a common due date to minimize total actual flow time. The algorithm was divided into two steps, namely batch and scheduling stages. In the first step, the number of batches ( $N$ ) and batch size ( $Q_{[i]}, i = 1, \dots, N$ ) were calculated using the solution for the common due date single machine problem [23]. Halim and Ohta [32] assumed that the demand for units could be divided into several continuous batch sizes, potentially resulting in  $m$  alternative solutions ( $N$ ) and  $Q_{[i]}, i = 1, \dots, N$ ). The second step included examining pairwise comparisons of the resulting batches from the first stage, and a final solution minimizing total actual flow time was selected from the  $m$  resolutions. Due to the large number of combinations, determining the effectiveness of the algorithm was challenging, necessitating the development of a comparison system to find a better solution.

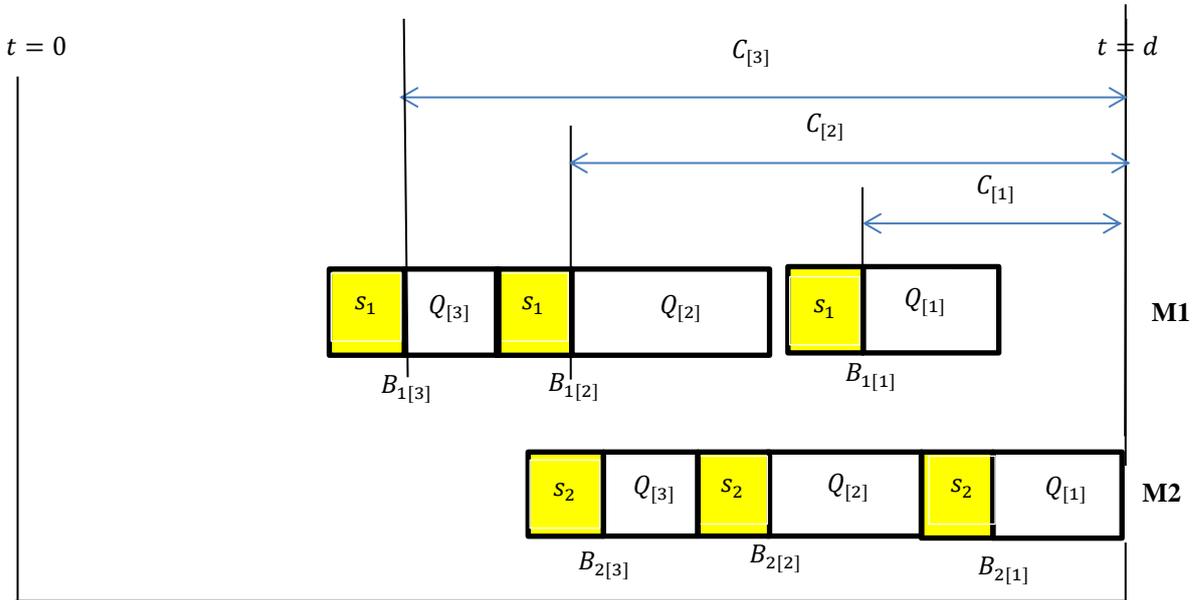
The analysis built on the research of Halim and Ohta [32], considering that batches consisted of discrete parts and reduced the number of machines to two ( $m = 1, 2$ ). The algorithm solution from [32] was enhanced by identifying the bottleneck machine and determining batch solution. Specifically, the research considered a single job with a certain number of units ( $n$ ) and a common due date ( $d$ ). It assumed the company could organize the arrival of raw materials in several batches ( $N$ ) with batch sizes ( $Q_{[i]}, i = 1, \dots, N$ ) according to the research needs. Backward scheduling method was adopted, with the first batch index being the closest to the common due date. Batches were processed in flow shop, starting on Machine 1 at time ( $t_1$ ) and continuing

Machine 2 with time ( $t_2$ ). The starting times of two-machine to process batches were denoted by  $B_{m[i]}$ ,  $i = 1, \dots, N$ . Since each batch consisted of multiple parts, a setup time ( $s_m$ ) was needed before the machine was processed.

The objective function was the total actual flow time of batches ( $F^a$ ). The trade-off in this study was evident by the maximizing number of batches ( $N = n$ ,  $Q_{[i]} = 1$ ) and the total setup time, leading to maximum total actual flow time. Meanwhile, minimizing the number of batches ( $N = 1$ ,  $Q_{[i]} = n$ ) increased the duration between the starting time to process batch with the due date, also resulting in maximum total actual flow time. The actual flow time of batch was calculated by finding the duration between the starting time to process batch on Machine 1 with the due date ( $C_{[i]}$ ), performed by batch sizes ( $F_{[i]}^a = C_{1[i]}Q_{[i]}$ ). The formula for the total actual flow time was as follows:

$$F^a = \sum_{i=1}^N (C_{[i]})Q_{[i]} \tag{1}$$

There were two alternatives for calculating  $C_{[i]}$  in two-machine flow shop scheduling problem. The selected method depended on the bottleneck machine, as shown in Figure 1.



**Figure 1.** A Gantt chart for two-machine flow shop problem with three batches

Figure 1 shows the processing of the number of units into three batches ( $N = 3$ ) in a two-machine flow shop. The assumed sequence of the three batches was  $Q_{[1]}$ ,  $Q_{[2]}$ , and  $Q_{[3]}$  consecutively. Batches were scheduled using backward scheduling method and the  $C_{[i]}$ ,  $i = 1, \dots, 3$ , was calculated as follows:

$$\begin{aligned} C_{[1]} &= t_2 Q_{[1]} + t_1 Q_{[1]} \\ C_{[2]} &= t_2 Q_{[1]} + \max\{(t_1 Q_{[1]} + s_1), (s_2 + t_2 Q_{[2]})\} + t_1 Q_{[2]} \\ C_{[3]} &= Q_{[1]} t_2 + \max\{(t_1 Q_{[1]} + s_1 + t_1 Q_{[2]} + s_1), (s_2 + t_2 Q_{[2]} + s_2 + t_2 Q_{[3]})\} + t_1 Q_{[3]} \\ C_{[i]} &= t_2 Q_{[1]} + \max\{(i - 1)s_1 + t_1 \sum_{j=1}^i Q_{[j]}, (i - 1)s_2 + t_2 \sum_{j=1}^i Q_{[j]}\} + t_1 Q_{[i]} \\ i &= 1, \dots, N \end{aligned} \tag{2}$$

The formula for  $\max\{(i - 1)s_1 + t_1 \sum_{j=1}^i Q_{[j]}, (i - 1)s_2 + t_2 \sum_{j=1}^i Q_{[j]}\}$  showed the bottleneck machine. When the value of  $(i - 1)s_1 + t_1 \sum_{j=1}^i Q_{[j]}$  was the maximum, then Machine 1 became the bottleneck machine and vice versa, Machine 2 was identified as the bottleneck machine.

The notation and mathematical models of the problem were as follows:

index

- $i$  : the batch index,  $i = 1, \dots, N$ .
- $m$  : the machine index,  $m = 1, 2$ .

parameter

- $n$  : the number of parts demanded during a scheduled period.
- $d$  : the common due date.
- $t_m$  : the processing time of part on *machines*.
- $s_m$  : the setup time required before processing any batch on *machine*.

variable

- $B_{m[i]}$  : the starting time of  $m$ -th machine to process  $i$ -th batch
- $F_{[i]}$  : the flow time for  $i$ -th batch.
- $C_{[i]}$  : the duration between the starting time to process  $i$ -th batch with the due date
- $N$  : the number of batches (the decision variable).
- $Q_{[i]}$  : the batch size, which consisted of the number of parts (the decision variable).

objective function

- $F^a$  : the total actual flow time

A mathematical model is shown as follows:

$$\text{Min } F^a = \sum_{i=1}^N (t_2 Q_{[1]} + \max\{[(i-1)s_1 + t_1 \sum_{j=1}^i Q_{[j]}], [(i-1)s_2 + t_2 \sum_{j=1}^i Q_{[j]}\} + t_1 Q_{[i]}) Q_{[i]} \quad (3)$$

Subject to:

$$\sum_{i=1}^N Q_{[i]} = n; \quad (4)$$

$$B_{1[1]} + C_{[1]} = d; \quad (5)$$

$$d - s_1 - C_{[N]} \geq 0; \quad (6)$$

$$d - t_2 n - N s_2 \geq 0; \quad (7)$$

$$N \geq 1; Q_{[i]} \geq 1; i = 1, \dots, N; Q_{[i]} \in \mathbb{N}; \quad (8)$$

Equation (3) represented the objective function, aiming to minimize the total actual flow time. This time was defined as a total interval for all components in batches ( $Q_{[i]}, i = 1, \dots, N$ ) from the arrival at the shop until the common due date ( $d$ ). Constraint (4) showed the material balance, asserting that the total number of unit components in all batches should be equal to the total unit components of demand. Constraint (5) stipulated that the first batch in backward scheduling would be completed exactly at the common due date ( $d$ ). Constraints (6) and (7) declared that all batches were processed during the scheduling period, ranging from  $t = 0$  to the due date. Constraint (8) specified that the minimum batch number and size were 1, and the element was a natural figure.

### Problem Solution

In this section, the decision variables were determined along with the formula for calculating the minimum total actual flow time.

#### 1) Determining the Bottleneck Machine

In two-machine flow shop problem, it was essential to identify the machine to serve as a reference in scheduling. The solution for calculating the number of batches and batch sizes on the machine would subsequently be applied to other devices. This determination of the reference machine was derived from the bottleneck machine, which was the device with the smallest capacity in comparison to others. The capacity calculation was obtained by substituting Equation (2) into (6).

$$d - s_1 - \max\{Q_{[1]}t_2 + [(i-1)s_1 + t_1 \sum_{j=1}^i Q_{[j]}], [(i-1)s_2 + t_2 \sum_{j=1}^i Q_{[j]}] + Q_{[i]}t_1\} = 0; \quad (9)$$

Equation (9) was dissected into two separate equations to show the bottleneck condition on Machine 1 and 2.

$$\text{Machine 1: } d - s_1 - (Q_{[1]}t_2 + [(i - 1)s_1 + t_1 \sum_{j=1}^i Q_{[j]}]) = 0; \tag{10}$$

$$\text{Machine 2: } d - s_1 - ((i - 1)s_2 + t_2 \sum_{j=1}^i Q_{[j]}) + Q_{[N]}t_1 = 0; \tag{11}$$

The calculation for the maximum capacity was derived when  $i=N$ , and it was obtained as follows:

$$\text{Machine 1: } d - s_1 - (Q_{[N]}t_2 + [(N - 1)s_1 + t_1 \sum_{j=1}^N Q_{[j]}]) = 0; \tag{12}$$

$$\text{Machine 2: } d - s_1 - ((N - 1)s_2 + t_2 \sum_{j=1}^N Q_{[j]}) + Q_{[N]}t_1 = 0; \tag{13}$$

By substituting Equation (4) into (12) and (13), Equations (14) and (15) were generated, as presented below:

$$\text{Machine 1: } d - s_1 - (Q_{[1]}t_2 + [(N - 1)s_1 + t_1n]) = 0; \tag{14}$$

$$\text{Machine 2: } d - s_1 - ((N - 1)s_2 + t_2n) + Q_{[N]}t_1 = 0; \tag{15}$$

The subsequent step included the current calculation of the maximum capacity of flow shop. This was achieved by computing the maximum capacity on two-machine consecutively. The capacity of Machine 1 reached the maximum when the distance between the  $B_{2[1]}$  and the due date was significantly close. This study assumed that batch size was a discrete unit, hence, the minimum distance was found when determining  $Q_{[1]} = 1$ . The capacity of Machine 2 attained the maximum when  $Q_{[N]} = 1$ , as the distance between time zero and the  $B_{2[N]}$  was very close. Consequently, Equations (14) and (15) were rephrased as follows:

$$\text{Machine 1: } d - s_1 - (t_2 + [(N - 1)s_1 + t_1(n - 1)]) = 0; \tag{16}$$

$$\text{Machine 2: } d - s_1 - ((N - 1)s_2 + t_2(n - 1)) + t_1 = 0; \tag{17}$$

Simplifying Equation (16) and concurrently solving Equations (7) and (17) simultaneously produced Equations (18) and (19) as follows:

$$\text{Machine 1: } d - t_2 - Ns_1 - t_1(n - 1) = 0; \tag{18}$$

$$\text{Machine 2: } d - \max\{s_1; (s_2 - t_1)\} - t_1 - (N-1)s_2 - t_2(n - 1) = 0; \tag{19}$$

The capacity of the machine represented the number of requested units that the device could process during the scheduling period, formulated as follows:

$$\text{Cap } M_1 = \frac{d - Ns_1 - t_2}{t_1} - 1; \tag{20}$$

$$\text{Cap } M_2 = \frac{d - \max\{s_1; (s_2 - t_1)\} - (N-1)s_2 - t_1}{t_2} - 1; \tag{21}$$

The maximum capacity of each machine was determined by assuming  $N = 1$ , resulting in the rewriting of Equations (20) and (21) as follows:

$$\text{Cap } M_1 = \frac{d - s_1 - t_2}{t_1} - 1; \tag{22}$$

$$\text{Cap } M_2 = \frac{d - \max\{s_1; (s_2 - t_1)\} - t_1}{t_2} - 1; \tag{23}$$

The capacity of the entire flow shop system was determined by the bottleneck machine, which was the device with the smallest capacity per period. The formula for calculating flow shop system capacity was formulated as follows:

$$\text{Cap} = \min\{\text{Cap } M_1, \text{Cap } M_2\} \tag{24}$$

The decision solution for the bottleneck machine was subsequently used as the fundamental solution for other devices.

2) Determining Batch Size

In this section, the formula for calculating batch size was determined. Substituting Equations (2) and (3) into Equation (1), assuming  $K$  as the notation for the bottleneck machine, produced the following equation:

$$F^a = \sum_{i=1}^N (Q_{[1]}t_2 + (i-1)s_K + t_K \sum_{j=1}^i Q_{[j]} + Q_{[i]}t_1)Q_{[i]} \quad (25)$$

$K \in m; i = 1, \dots, N$

The formula for the total actual flow time in Equation (25) was rewritten as follows:

$$F^a = s_K \sum_{i=1}^N iQ_{[i]} + \frac{1}{2}(t_K + 2 \sum_{u=1}^{K-1} t_u) \sum_{i=1}^N Q_{[i]}^2 + \frac{1}{2}t_K (\sum_{i=1}^N Q_{[i]})^2 - [s_K - t_2Q_{[1]}] \sum_{i=1}^N Q_{[i]}; \quad (26)$$

with  $\sum_{u=1}^{K-1} t_u = 0$ , if  $m = 1$ ;  
 $K \in m; i = 1, \dots, N$

Solving Equation (26) using the Lagrange relaxation technique produced the following stationary points:

$$s_K i + (t_K + 2 \sum_{u=1}^{K-1} t_u)Q_{[i]} + t_K \sum_{i=1}^N Q_{[i]} - s_K + t_2Q_{[1]} - \lambda_1 - t_K \lambda_2 = 0; \quad (27)$$

with  $\sum_{u=1}^{K-1} t_u = 0$ , if  $m = 1$ ;  
 $K \in m; i = 1, \dots, N$

Substituting Equation (4) into (27) resulted in:

$$s_2 i + (t_2 + 2t_1)Q_{[i]} + t_2 n - s_2 + t_2Q_{[1]} - \lambda_1 - t_2 \lambda_2 = 0; \quad (28)$$

with  $\sum_{u=1}^{K-1} t_u = 0$ , if  $m = 1$ ;  
 $K \in m; i = 1, \dots, N$

The solution (28) to find  $Q_{[i]}$  was obtained as follows:

$$Q_{[i]} = \frac{-(s_K i + t_K n - s_K + t_2Q_{[1]} - \lambda_1 - t_K \lambda_2)}{(t_K + 2 \sum_{u=1}^{K-1} t_u)} \quad (29)$$

with  $\sum_{u=1}^{K-1} t_u = 0$ , if  $m = 1$ ;  
 $K \in m; i = 1, \dots, N$

Substituting Equation (29) into (4) resulted in:

$$\lambda_1 = (t_K + 2 \sum_{u=1}^{K-1} t_u)/N + t_K n + \frac{1}{2}s_K(N+1) - s_K + t_2Q_{[1]} - t_c \lambda_2, \quad (30)$$

with  $\sum_{u=1}^{K-1} t_u = 0$ , jika  $m = 1$ ;  
 $K \in m; i = 1, \dots, N$

Substituting Equation (30) into (29) resulted in:

$$Q_{[i]} = \frac{n}{N} + \frac{s_c}{2(t_K + 2 \sum_{u=1}^{K-1} t_u)} + \frac{Ns_c}{2(t_K + 2 \sum_{u=1}^{K-1} t_u)} - \frac{is_c}{(t_K + 2 \sum_{u=1}^{K-1} t_u)} \quad (31)$$

With  $\sum_{u=1}^{K-1} t_u = 0$ , if  $m = 1$ ;  
 $K \in m; i = 1, \dots, N$

The formula in Equation (31) produced a continuous batch size (a real number). However, the objective in this investigation aimed to achieve results represented by distinct batch sizes (integers), leading to the modification of Equation (31) in the subsequent manner.

$$Q_{[N+1-i]} = \left\lceil \frac{n - \sum_{j=1}^i Q_j}{N + 1 - i} + \frac{s_K}{2(t_K + 2t_1)} + \frac{(N + 1 - i)s_K}{2(t_K + 2 \sum_{u=1}^{K-1} t_u)} - \frac{(N + 1 - i)s_K}{(t_K + 2 \sum_{u=1}^{K-1} t_u)} \right\rceil \quad (32)$$

with  $\sum_{u=1}^{K-1} t_u = 0$ , if  $m = 1$ ;

$K \in m$ ;  $i = 1, \dots, N$

$\lceil x \rceil$  was rounding up when decimal of  $x \geq 0.5$

### 3) Determining the Number of Batches

After determining the formula for calculating batch size, the subsequent step was to establish the optimal batch number. In Constraint (8), it was established that all batch sizes should be positive, thereby  $Q_{[N]} \geq 1$ . Substitution  $i = N$  in Equation (31) produced the following outcome.

$$Q_{[N]} = \frac{n}{N} + \frac{s_K}{2Y} + \frac{Ns_K}{2Y} - \frac{Ns_K}{Y} \geq 0 \quad (33)$$

$Y = t_K + 2 \sum_{u=1}^{K-1} t_u$

$K \in m$ ;  $m = 1, 2$ ;  $i = 1, \dots, N$

The completion of Equation (32) to obtain  $N_{max}$  was accomplished.

$$N_{max} = \left\lfloor \frac{A}{2s_K} \right\rfloor - 1 \quad (34)$$

$A = s_K + \sqrt{(s_K)^2 + 16ns_KY + 8ns_Kt_K}$

$Y = t_K + 2 \sum_{u=1}^{K-1} t_u$

$\lfloor X \rfloor$  is the largest integer less than  $X$ .

$K \in m$ ;  $m = 1, 2$ ;  $i = 1, \dots, N$

### 4) Determine the Initial Batch Quantity

The initial batch number was calculated by solving Equations (20) and (21), resulting in the following formula:

$$N^0 = \begin{cases} \left\lfloor \frac{d - t_2 - t_1n}{s_1} \right\rfloor; & \text{if } K = 1; \\ \left\lfloor \frac{d - \max\{s_1; (s_2 - t_1)\} - t_1 - t_2n}{s_2} \right\rfloor; & \text{if } K = 2 \end{cases} \quad (35)$$

$\lfloor X \rfloor$  was the largest integer less than  $X$ .

### 5) Determining the Number of Batches

The number of batches was determined by Lemma as follows:

**Lemma 1.** The objective function of total actual flow time was a convex function on the closed interval  $[1, N_{max}]$ . The minimum total actual flow time was obtained for the number of batches in this interval.

**Proof.** The objective function of total actual flow time was a convex function on the closed interval  $i = [1, N_{max}]$  when the Hessian matrix satisfied the positive definite condition. Each  $i$  in Equation (26) was subsequently written as:

$$f(i) = \{s_K i - [s_K - t_2 Q_{[1]}]\} \sum_{j=1}^i Q_{[j]} + \frac{1}{2} (t_K + 2t_1) \sum_{i=1}^N Q_{[i]}^2 + \frac{1}{2} t_K (\sum_{i=1}^N Q_{[i]})^2 \quad (36)$$

The first Derivative ( $\nabla^2 f(i)$ ) is obtained.

$$\nabla^2 f(i) = \begin{cases} 2t_2 Q_{[1]}; & i = 1; \\ \{t_K + 2t_1\} Q_{[i]} + (i - 1)s_K + t_K Q_{[i]}; & i = 2, \dots, N_{max}; \end{cases} \quad (37)$$

Hessian Matrix:

$$H(i) = \nabla^2 f(i) = \begin{pmatrix} 2t_2 & 0 & 0 & 0 \\ 0 & t_K + 2t_1 & 0 & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & t_K + 2t_1 & 0 \\ 0 & 0 & 0 & t_K + 2t_1 \end{pmatrix} \quad (38)$$

$i = 1, 2, \dots, N_{max}, a, K \in m$

The condition for a convex function was positive definite, namely  $detH(i) > 0$  and  $detH(i) = 2^{N_{max}}t_2(t_K + 2t_1)^{(N_{max}-1)} > 0$ , (positive definite).

Therefore, the objective function was a convex function on the closed interval  $[1, N_{max}]$ .  
 Proved ■

6) Define Batch Scheduling Rules

In this study, batch scheduling adopted the rule of Johnson and was subsequently developed for the problems. The algorithm for determining batch size sequence was as follows:

Batch Scheduling Sub-Algorithm

- Step 0. Set the initial schedule sequence by considering batch index ( $Q_{[i]}$ ), where  $i = 1, \dots, N$ .
- Step 1. Create a matrix A with dimensions  $M \times N$ , where  $M$  represented the number of machines ( $M=2$ ), and  $N$  signified the number of batches.
- Step 2. Calculate the elements of matrix A, where  $A_{m,i} = t_m Q_{[i]} + s_m$ , with  $m = 1, 2; i = 1, \dots, N$  and continue to Step 3.
- Step 3. The element in matrix A with the smallest value should be selected.
- Step 3.1. When the smallest element value occurred on Machine 1, batch index ( $i$ ) of that element was positioned at the end of the scheduled series.
- Step 3.2. When the smallest element value resided on Machine 2, batch index ( $i$ ) of the element was placed at the beginning of the scheduled series.
- Step 3.3. When the value of the smallest element was the same for Machines 1 and 2, batch index ( $i$ ) of the element was positioned at the beginning of the scheduled series, then continued to step 4.
- Step 4. The column where the smallest matrix element in Step 3 was eliminated.
- Step 5. When the number of columns in matrix A was no longer greater than 1, proceed to establish a new schedule order and conclude the process. ■

**Proposed Scheduling Algorithm**

PBFS-2M Algorithm (2 Machine Flow shop Batch Scheduling)

- Step 0. Input parameters  $n, d, t_m, s_m$  and continue to Step 1.
- Step 1. The bottleneck machine ( $K$ ) was selected.
- Step 1.1.  $Cap M_1$  and  $Cap M_2$  with Equations (22) and (23) was computed.
- Step 1.2. The bottleneck machine was selected and moved to Step 2.
- Step 2.  $N_{max}$  with Equation (34) was calculated.
- Step 3. Equation (35) was subsequently used to calculate  $N^0$  and to verify  $N_{max}$
- Step 3.1. When  $N^0 \leq 1$ , then  $N^0 \leq 1$ .
- Step 3.2. When  $N^0 \geq N_{max}$ , then  $N_{max} = N_{max}$
- Step 3.3. When  $1 < N^0 < N_{max}$ , then  $N_{max} = N^0$
- Step 4. Start from  $N=1$ .
- Step 5.  $N$  was used as input for Equation (32) in calculating  $Q_{[i]}$ .
- Step 6. Batch Scheduling Sub Algorithm was adopted to find batch order index  $i = 1, \dots, N$ .
- Step 7. Batch was scheduled according to the parcel index order  $Q_{[i]}, i = 1, \dots, N$ .
- Step 8.  $F^a$  was calculated with Equation (1). When  $N = 1$ , move to Step 9 but when  $N > 1$  then skip Stage 9 and move to 10.
- Step 9.  $F^a$  was set as  $F^{a'}$  and moved to Step 11.
- Step 10.  $F^a$  was compared with  $F^{a'}$ . When  $F^a < F^{a'}$  then  $F^a = F^{a'}$  subsequently continuing with Step 11.
- Step 11.  $N = N + 1$  was set. When  $N < N_{max}$ , return to Step 5, otherwise continue to Step 12.
- Step 12. The total actual flow time was set at  $F^{a'}$  then finally stopped the procedure.

## Results and Discussions

In this section, the outcomes of the implemented algorithm in straightforward scenarios were examined, alongside a discussion of the results derived from numerical experiments conducted on more intricate cases that comprised combinations of diverse parameters. Applying the algorithm to simple cases served the purpose of presenting a comprehensive understanding of the functionality and the outcomes achieved. Meanwhile, the objective of the numerical experiments was to obtain insights into algorithmic solutions for addressing complex problem scenarios.

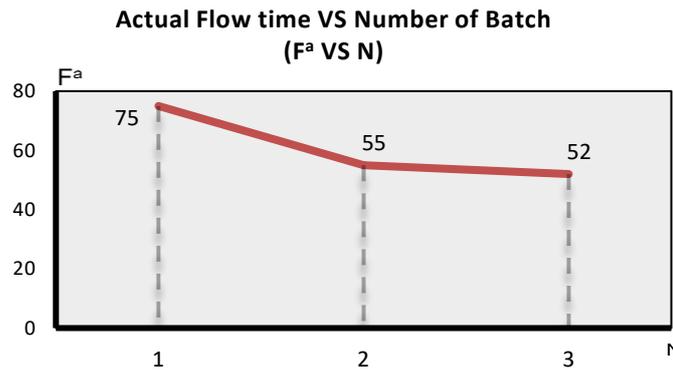
### Results

The application of the proposed algorithm was shown through two simple case examples. In the first case, the bottleneck machine was assigned to the other device, while in the second scenario, it was assigned to the first machine.

**First Case.** Given the following parameters,  $n = 5$  unit,  $d = 25$  hours,  $t_1 = 1$  hours/unit,  $t_2 = 2$  hours/unit,  $s_1 = 3$  hours/batch,  $s_2 = 2$  hours/batch. The number of batches, integer batch size, and batch scheduling were determined and the total actual flow time  $F^a$  was subsequently calculated.

**Completion.** The application of the proposed algorithm resulted in the following outcomes. In the first step, the capacity of Machine 1 was determined to be fifteen units per scheduling period (25 hours), while the efficiency of Machine 2 was 7.5 units per 25 hours. Consequently, flow shop capacity was 7.5 units per 25 hours, identified on Machine 2 as the bottleneck machine ( $K=2$ ). In the second step,  $N_{max}$  was found to be five, and in the third step,  $N^0$  was 3, thereby confirming that the previous  $N_{max}$  value was indeed 3.

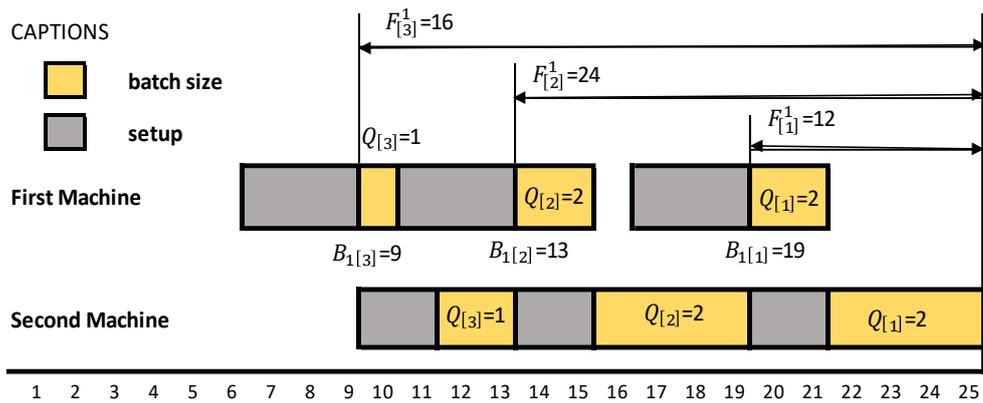
The subsequent step included iteratively searching for the minimum total actual flow time in the range of  $N = 1$  to  $N = N_{max}$ . Following the guidelines of Lemma 1, the algorithm halted when the actual flow time value for  $N$  in an iteration surpassed the value for  $N$  in the preceding recursion, or  $N$  had reached the value  $N_{max}$ . In the initial iteration,  $N = 1$  yielded  $Q_{[1]}=5$ , resulting in  $F^a = 75$ . Subsequent calculations for  $N=2$  yielded  $Q_{[1]}=3$  and  $Q_{[2]}=2$ , with an associated  $F^a$  value of 55. Recognizing this improvement,  $F^a = 55$  was set as  $F^{a'}$ . In the third iteration  $N=3$ , the values  $Q_{[1]}=2$ ,  $Q_{[2]}=2$ , and  $Q_{[3]}=1$  were obtained, yielding  $F^a = 52$ . Given the further enhancement in  $F^a$ , this value was designated as  $F^{a'}$ . The third iteration marked the final recursion since  $N=N_{max}$ , prompting the algorithm to cease and establish the decision as  $N=3$ ,  $Q_{[1]}=2$ ,  $Q_{[2]}=2$ , and  $Q_{[3]}=1$ , with  $F^a = 52$ . Figure 2 shows the solution for each algorithm iteration ( $N$  versus  $F^a$ ).



**Figure 2.** Actual flow time and number of batches graph for the first case

The Gantt chart of the selected solution with  $F^a = 52$  was shown in Figure 3.

In Figure 3, the identification of the bottleneck machine was evident on the second machine. The outcome of the algorithm included the segmentation of a request for five units into three batches ( $N = 3$ ) with batch sizes ( $Q_{[i]}, i = 1,2,3$ ) of 2 units, 2 units, and 1 unit, respectively. Raw materials were introduced to the production section at the 9<sup>th</sup>, 13<sup>th</sup>, and 19<sup>th</sup> hours to facilitate the simultaneous delivery of finished products on the due date (25<sup>th</sup> hour), minimizing the time inventory on the shop floor. The total actual flow time, representing the duration of inventory presence on the production floor, amounts to 52 hours. This duration consisted of 12, 24, and 16 hours for the 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> batches, respectively. The results showed that the proposed algorithm signified efficacy in addressing the problem.



**Figure 3.** Gantt Chart batch scheduling solution for parameters:

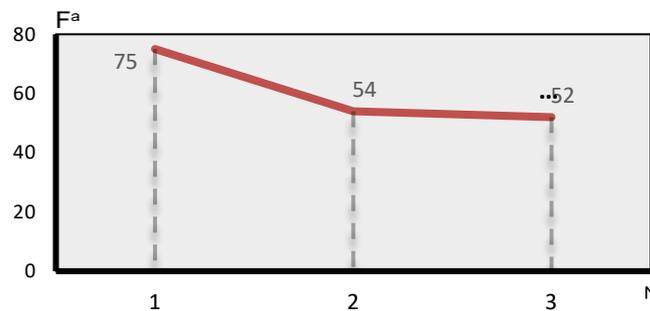
$n = 5$  unit,  $d = 25$  hours,  $t_1 = 1$  hours/unit,  $t_2 = 2$  hours/unit,  $s_1 = 3$  hours/batch,  $s_2 = 2$  hours/batch

**Second Case.** Given the following parameters,  $n = 5$  unit,  $d = 25$  hours,  $t_1 = 2$  hours/unit,  $t_2 = 1$  hours/unit,  $s_1 = 2$  hours/batch,  $s_2 = 3$  hours/batch. The optimal batch count, integer batch size, batch schedule, and total actual flow time  $F^a$  were determined.

**Completion.** The proposed algorithm yielded the following outcomes upon completion. In the initial step, the capacity of Machine 1 was obtained as 10 units per scheduling period (25 hours), while the output of Machine 2 was found to be 20 units per 25 hours. Consequently, flow shop capacity was established as 10 units per 25 hours, with Machine 2 identified as the bottleneck machine ( $K = 1$ ). In the second step,  $N_{max}$  was calculated to be 3, and in the third stage,  $N^0$  was determined to be 7, confirming the consistency of the previous  $N_{max}$  value as 3.

The subsequent step included determining the minimum actual flow time iteratively for values  $N$  ranging from 1 to  $N_{max}$ . Following the criteria outlined in Lemma 1, the algorithm will terminate when the actual flow time value for  $N$  in the iteration exceeded the figure for  $N$  in the previous recursion or  $N$  reaches  $N_{max}$ . In the initial iteration, specifically for  $N = 1$ ,  $Q_{[1]}=5$  was obtained with  $F^a = 75$ . The calculation continued for  $N = 2$ , resulting in  $Q_{[1]}=2$  and  $Q_{[2]}=3$ , yielding an  $F^a$  value of 54. As this  $F^a$  value was an improvement from the previous iteration,  $F^a=54$  was designated as  $F^{a'}$ . The third iteration corresponded to  $N = 3$ , with  $Q_{[1]}=2$ ,  $Q_{[2]}=2$ , and  $Q_{[3]}=1$  being obtained and  $F^a$  value was equal to 52. This  $F^a$  value was better than the previous iteration, leading to the designation as  $F^{a'}$ . The third iteration marked the conclusion of the algorithm as  $N = N_{max}$ , prompting the procedure to halt and finalize the decision as  $N = 3$ ,  $Q_{[1]} = 2$ ,  $Q_{[2]} = 2$ , and  $Q_{[3]} = 1$ , with  $F^a = 52$ . The progression of the algorithm iteration ( $N$  versus  $F^a$ ) is depicted in Figure 4.

**Actual Flow time VS Number of Batch ( $F^a$  VS  $N$ )**

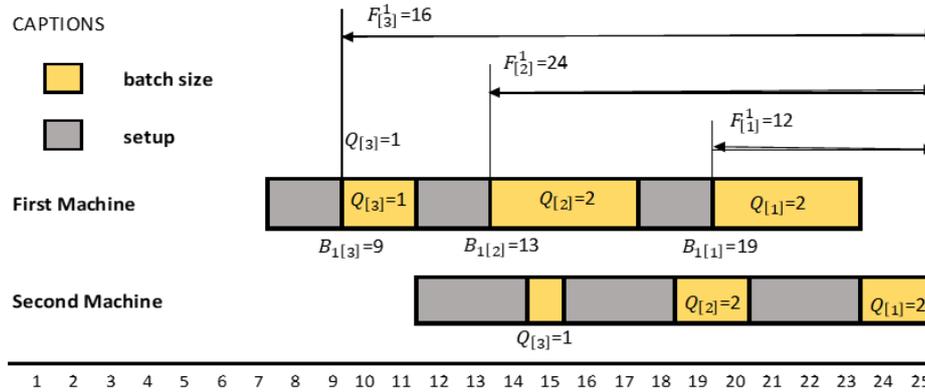


**Figure 4.** Actual flow time and number of batches graph for the second case

The Gantt chart of the selected solution with  $F^a = 52$  can be seen in Figure 5.

In the experimental example, both Cases 1 and 2 yielded the same solution, specifically  $N = 2$ , with  $Q_{[1]} = 2$ ,  $Q_{[2]} = 2$ , and  $Q_{[3]} = 1$ , resulting in an  $F^a$  value of 52. This similarity in solutions occurred despite the bottleneck machine being on different devices. The identical batch size solutions, influenced by rounding, and

the consistent schedule order contributed to this outcome.



**Figure 5.** Gantt chart of batch scheduling solutions for parameters:  $n = 5$  unit,  $d = 25$  hours,  $t_1 = 2$  hours/unit,  $t_2 = 1$  hours/unit,  $s_1 = 2$  hours/batch,  $s_2 = 3$  hours/batch

Both Case 1 and Case 2 allowed for solutions with continuous batch sizes. In Case 1, where the bottleneck machine was on Machine 2 ( $K = 2$ ), the algorithm produced solutions with  $N = 3$ ,  $Q_{[1]} = 2.1667$ ,  $Q_{[2]} = 1.6667$ , and  $Q_{[3]} = 1.1667$ , resulting in an actual flow time of  $F^a = 51.06$ . Meanwhile, in Case 2, where the bottleneck machine was on Machine 1 ( $K = 1$ ), the algorithm generated solutions with  $N=3$ ,  $Q_{[1]} = 1.6667$ ,  $Q_{[2]} = 2.6667$ , and  $Q_{[3]} = 0.1667$ , leading to an  $F^a$  value of 52.78. The processing results from the proposed algorithm for both cases, including integer and continuous batch size solutions, showed the sensitivity of actual flow time to changes in one or more decision variable values. This sensitivity was evident when transitioning from continuous solutions to integer solutions (as observed in Case 1 and Case 2), showing that alterations in batch size and sequence could impact actual flow time, even when the number of batches remained constant.

### Discussions

In this section, a comparative analysis is conducted between the performance of the proposed algorithm and the procedure outlined by Halim [23]. The objective is to assess how the proposed algorithm performs in comparison to existing procedures. The distinctions between the Halim procedure and the proposed algorithm are as follows.

Halim and Ohta [32] introduced a procedure to address flow shop problem to minimize the total actual flow time. Halim procedure operates by determining the optimal solution through a comparison of solutions obtained from each flow shop machine ( $m = 1$  and  $m = 2$ ). The search for a solution on a machine includes iteratively obtaining values for  $N$  and  $Q_{[i]}$ ,  $i = 1, \dots, N$ . It starts at  $N = 1$  and concludes when  $N$  reaches  $N_{max}$  or when the solution in an iteration is not superior to the previous recursion. In each  $N$  iteration, the best sequence is determined by interchanging batch sizes using the pairwise interchange (PI) method. The outcome on a machine is then compared with the remedy on the other device, and the optimal resolution is identified as the algorithmic solution.

**Proposed Algorithm.** The algorithm is devised by identifying the bottleneck machine on flow shop device, and the solution from the number of batches and batch size serves as the scheduling basis for others. Similar to Halim procedure, the search for the bottleneck machine solution commences iteratively from  $N = 1$  to  $N = N_{max}$ , or stops when the resolution in the iteration is not superior to the previous recursion. Batch order at each iteration is determined based on Lemma 2 and 3, eventually establishing the best resolution as the algorithmic solution. Consequently, the proposed algorithm features more concise steps compared to the procedure of Halim *et al.* [23].

Two algorithms are executed using MATLAB R2014A application, and the outcomes are being reported using an Intel® i7® 860 processor at 2.80 GHz (4 cores, eight threads) and 8 GB of RAM. Testing is conducted for two categories of problems, namely (1) where the machine processing time is greater than the setup time, and (2) where the machine processing time is smaller than the setup time. For the first category, the machine processing time data is randomly generated between the numbers 3 and 5, and the setup duration is produced between 0.5 and 1. In the second category, the machine processing time data is randomly formulated between 2 and 4, while the setup duration is divided between 6 and 8. The number of requests is generated between 12

and 60, and the deadline is produced between 600 and 900 to ensure optimal algorithm performance.

**Table 1.** Example of algorithm comparison test results for input data ( $n = 13, d = 606, t_1 = 2, t_2 = 3, s_1 = 6.8764, t_2 = 7.4702$ )

Algorithm	$K$	$N$	$Q_{[i]}, i = 1, \dots, N$	$F^a$	Time lapsed
Proposed algorithm	2	5	4.7343, 3.6672, 2.6000, 1.5328, 0.4657	526.68	0.158967
Halim Procedure	2	3	6.8234, 4.3333, 1.8433	558.99	0.239596
Proposed algorithm					
Winner percentage	=100% x 1/1 = 100%				
Effectiveness (%)	=100% x {(558.99 - 526.68)/526.68} = 6%				
Time-saving ratio	=0.158967: 0.239596 (1: 2)				

The simulation is carried out using 1000 cases for each category. The test results are evaluated by examining the number of winners, the percentage increase in effectiveness, and the ratio of computing time savings. Examples of comparison results are shown in Table 1, while Table 2 provides a summary of the comparison results for all cases.

**Table 2.** Recapitulation of comparative testing results of the Proposed Algorithm versus Halim Procedure (Testing data of 1000 cases for each category)

	Proposed algorithm	Halim procedure
<b>First category (<math>t_1, t_2 &gt; s_1, s_2</math>)</b>		
Total actual flow time		
Mean	4035.231	4097.270
Standard deviation	2635.345	2663.331
Max	10496.51	10741.89
Min	377.8	384.6
Winner percentage (1000 cases)	67.2 %	9%
Draw percentage	23.8%	23.8%
Effectiveness (%)	1.44%	
Time computation		
Mean	0.045	0.334
Standard deviation	0.027209	0.227814
Max	0.216892	1.677439
Min	0.007265	0.030755
Time-saving ratio	1	8
<b>Second category (<math>t_1, t_2 \leq s_1, s_2</math>)</b>		
Total actual flow time		
Mean	4332.411	4520.584
Standard deviation	2767.301	2868.815
Max	11415.72	12364.33
Min	501.88	538.72
Winner percentage	69.6%	0.2%
Draw percentage	30.2%	30.2%
Effectiveness (%)	4.34%	
Time computation		
Mean	0.009	0.013
Standard deviation	0.006203	0.009355
Max	0.160918	0.229302
Min	0.001159	0.003007
Time-saving ratio	1	2

Table 1 shows the outcomes of both the proposed and Halim procedure, along with examples implying the comparative analysis calculations between the two procedures. The proposed algorithm in this case identifies the bottleneck machine ( $K$ ) as Machine 2, with ( $N$ ) equal to 5 and a total actual flow time ( $F^a$ ) of 526.68, achieved in 0.158967 seconds. Meanwhile, Halim procedure designates  $K$  as 2,  $N$  as 3,  $F^a$  as 558.99, and a computing time of 0.239596 seconds. A comparative analysis shows the proposed algorithm as the winner in this instance, showing 6% effectiveness and a computing time efficiency ratio twice as short.

Table 2 presents the results of the comparative analysis across distinct categories. Examining the experimental outcomes for both categories shows that the proposed algorithm consistently yields a shorter total actual flow

time compared to Halim procedure. In the first category, the proposed algorithm prevailed in 672 instances (67.2%), drew 238 times (23.8%), and lost nine times (2%) against Halim procedure. It showed a 1.44% enhancement in solution effectiveness with a computing time efficiency ratio eight times shorter. However, in the second category, the proposed algorithm secured victory 699 times (69.9%), achieved a draw in 302 instances (30.2%), and encountered defeat twice (2%) when compared to Halim procedure [22]. The proposed algorithm showed a 4.34% improvement in solution effectiveness with a computing time efficiency ratio twice as short. The results affirm the efficacy of the proposed algorithm in satisfactorily addressing two-machine flow shop problem.

## Conclusion

In conclusion, this study explored batch scheduling model in a two-machine flow shop, where the process flow moved from Machine 1 to Machine 2, with the aim of minimizing the total actual flow time. In comparison to previous research [23], the analysis expanded on existing knowledge by, (1) establishing rules for using the bottleneck machine as reference points to determine the number and size of batches, (2) devising significant methods to ascertain the number of batches and sizes, (3) implementing the Johnson algorithm for batch scheduling, and (4) proposing an algorithm to address batch scheduling problems. Further development was needed to improve performance and expand the applicability of this study to  $M$  machine flow shop problems and more complex scheduling scenarios, including considerations of machine availability.

## Acknowledgment

The authors are grateful to the editors and referees for the valuable comments and suggestions. This study received support from the Industrial Engineering Department of Jenderal Achmad Yani University (UNJANI) Bandung and the Professional Education of Engineer's Department of Atma Jaya Indonesian Catholic University (UNIKA) Jakarta.

## References

- [1] A. Allahverdi, "A survey of scheduling problems with uncertain interval/bounded processing/setup times," *Journal of Project Management*, vol. 7, no. 4, pp. 255–264, 2022, doi: 10.5267/j.jpm.2022.3.003.
- [2] A. Allahverdi, "A survey of scheduling problems with no-wait in process," *European Journal of Operational Research*, vol. 255, no. 3, pp. 665–686, 2016, doi: 10.1016/j.ejor.2016.05.036.
- [3] A. Allahverdi, C. T. Ng, T. C. E. Cheng, and M. Y. Kovalyov, "A survey of scheduling problems with setup times or costs," *European Journal of Operational Research*, vol. 187, no. 3, pp. 985–1032, 2008, doi: 10.1016/j.ejor.2006.06.060.
- [4] B. Çalış and S. Bulkan, "A research survey: review of AI solution strategies of job shop scheduling problem," *Journal of Intelligent Manufacturing*, vol. 26, no. 5, pp. 961–973, 2015, doi: 10.1007/s10845-013-0837-8.
- [5] J. W. Fowler, and L. Mönch, "A survey of scheduling with parallel batch (p-batch) processing," *European Journal of Operational Research*, vol. 298, no. 1, pp. 1–24, 2022, doi: 10.1016/j.ejor.2021.06.012.
- [6] S. Hartmann, and D. Briskorn, "An updated survey of variants and extensions of the resource-constrained project scheduling problem," *European Journal of Operational Research*, vol. 297, no. 1, pp. 1–14, 2022, doi: 10.1016/j.ejor.2021.05.004.
- [7] B. M. T. Lin, "Fabrication scheduling on a single machine with due date constraints," *European Journal of Operational Research*, vol. 136, no. 1, pp. 95–105, 2002, doi: 10.1016/S0377-2217(01)00036-4.
- [8] S. Reza Hejazi, and S. Saghafian, "Flowshop-scheduling problems with makespan criterion: A review," *International Journal of Production Research*, vol. 43, no. 14, pp. 2895–2929, 2005, doi: 10.1080/0020754050056417.
- [9] D. Shabtay, and G. Steiner, "A survey of scheduling with controllable processing times," *Discrete Applied Mathematics*, vol. 155, no. 13, pp. 1643–1666, 2007, doi: 10.1016/j.dam.2007.02.003.
- [10] M. Sterna, "Late and early work scheduling: A survey," *Omega (United Kingdom)*, vol. 104, no. xxxx, 2021, doi: 10.1016/j.omega.2021.102453.
- [11] S. Suprayogi, and V. Valentine, "Teknik Relaksasi Lagrange untuk Penjadwalan Pekerjaan Majemuk dengan Penggunaan Sumberdaya Simultan," *Jurnal Teknik Industri: Jurnal Keilmuan dan Terapan Teknik Industri, UK. Petra*, vol. 17, no. 2, pp. 71–80, Dec. 2015, doi: 10.9744/jti.17.2.71-80.
- [12] K. R. Baker, *Introduction to Sequencing and Scheduling*. New York: JOHN WILEY & SONS, INC, 1974.
- [13] S. M. Johnson, "Optimal two-and three-stage production schedules with setup times included," *Naval Research Logistics Quarterly*, vol. 1, no. 1, pp. 61–68, 1954.

- [14] H. G. Campbell, R. A. Dudek, and M. L. Smith, "A heuristic algorithm for the n job, m machine sequencing problem," *Management Science*, vol. 16, no. 10, pp. B-630, 1970.
- [15] P. Brucker, B. Jurisch, and B. Sievers, "A branch and bound algorithm for the job-shop scheduling problem," *Discrete Applied Mathematics*, vol. 49, no. 1–3, pp. 107–127, 1994, doi: 10.1016/0166-218X(94)90204-6.
- [16] R. Yusriski, R. Pardiyono, Zahedi, S. Rahmawati, and A. Ramdhani, "Dynamic flexible flow shop scheduling with theory of constraint and time window approach to minimize mean tardiness," *AIP Conference Proceedings*, vol. 2510, no. 1, pp. 40026, Oct. 2023, doi: 10.1063/5.0128591.
- [17] C. Blum, and M. Sampels, "An ant colony optimization algorithm for shop scheduling problems," *Journal of Mathematical Modelling and Algorithms*, vol. 3, no. 3, pp. 285–308, 2004, doi: 10.1023/B:JMMA.0000038614.39977.6f.
- [18] T. Murata, H. Ishibuchi, and H. Tanaka, "Multiobjective genetic algorithm and its application to flowshop scheduling," *Computers and Industrial Engineering*, vol. 30, pp. 957–968, 1996, doi: [http://dx.doi.org/10.1016/0360-8352\(96\)00045-9](http://dx.doi.org/10.1016/0360-8352(96)00045-9).
- [19] E. Nowicki, and C. Smutnicki, "A fast tabu search algorithm for the permutation flow-shop problem," *European Journal of Operational Research*, vol. 91, no. 1, pp. 160–175, 1996, doi: 10.1016/0377-2217(95)00037-2.
- [20] B. M. T. Lin, and T. C. E. Cheng, "Batch scheduling in the no-wait two-machine flowshop to minimize the makespan," *Computers & Operations Research*, vol. 28, no. 7, pp. 613–624, 2001, doi: [https://doi.org/10.1016/S0305-0548\(99\)00138-0](https://doi.org/10.1016/S0305-0548(99)00138-0).
- [21] M. Pranzo, "Batch scheduling in a two-machine flow shop with limited buffer and sequence independent setup times and removal times," *European Journal of Operational Research*, vol. 153, no. 3, pp. 581–592, 2004, doi: [https://doi.org/10.1016/S0377-2217\(03\)00264-9](https://doi.org/10.1016/S0377-2217(03)00264-9).
- [22] L. Ben-Dati, G. Mosheiov, and D. Oron, "Batch scheduling on two-machine flowshop with machine-dependent setup times," in *Advances in Operations Research*, 2009, pp. 10. doi: 10.1155/2009/153910.
- [23] A. H. Halim, S. Miyazaki, and H. Ohta, "Batch-scheduling problems to minimize actual flow times of parts through the shop under JIT environment," *European Journal of Operational Research*, vol. 72, no. 3, pp. 529–544, 1994, doi: 10.1016/0377-2217(94)90421-9.
- [24] D. Kurniawan, R. Yusriski, M. M. Isnaini, M. Anas, and A. H. Halim, "A flow shop batch scheduling model with part deterioration and operator learning-forgetting effects to minimize total actual flow time," in *Proceedings of the Second Asia Pacific International Conference on Industrial Engineering and Operations Management*, 2021, pp. 644–655.
- [25] A. S. Indrapriyatna, S. Suprayogi, B. P. Iskandar, and A. H. Halim, "Model penjadwalan batch pada satu mesin yang mengalami deteriorasi untuk minimasi total biaya simpan dan biaya kualitas," *Jurnal Teknik Industri: Jurnal Keilmuan dan Terapan Teknik Industri*, vol. 10, no. 1, pp. 26–37, Jul. 2008, doi: 10.9744/jti.10.1.26-37.
- [26] R. Yusriski, Sukoyo, T. M. A. A. Samadhi, and A. H. Halim, "An integer batch scheduling model for a single machine with simultaneous learning and deterioration effects to minimize total actual flow time," *IOP Conference Series: Materials Science and Engineering*, vol. 114, no. 1, 2016, doi: 10.1088/1757-899X/114/1/012073.
- [27] R. Yusriski, B. Astuti, D. Biksono, and T. A. Wardani, "A single machine multi-job integer batch scheduling problem with multi due date to minimize total actual flow time," *Decision Science Letters*, vol. 10, no. 3, pp. 231–240, 2021, doi: 10.5267/j.dsl.2021.4.002.
- [28] R. Yusriski, S. Sukoyo, T. M. A. A. Samadhi, and A. H. Halim, "Integer batch scheduling problems for a single-machine with simultaneous effects of learning and forgetting to minimize total actual flow time," *International Journal of Industrial Engineering Computations*, vol. 6, no. 3, pp. 365–378, 2015, doi: 10.5267/j.ijiec.2015.2.005.
- [29] Zahedi, Rojali, and R. Yusriski, "Stepwise optimization for model of integrated batch production and maintenance scheduling for single item processed on flow shop with two machines in jit environment," *Procedia Computer Science*, vol. 116, pp. 408–420, 2017, doi: 10.1016/j.procs.2017.10.081.
- [30] Zahedi, T. A. Samadhi, Suprayogi, and A. H. Halim, "Model integrasi penjadwalan produksi batch dan penjadwalan perawatan dengan kendala due date," *Jurnal Teknik Industri: Jurnal Keilmuan dan Terapan Teknik Industri, UK. Petra*, vol. 16, no. 2, pp. 75–83, 2014, doi: 10.9744/jti.16.2.73-82.
- [31] Z. Zahedi, A. Salim, R. Yusriski, and H. Haris, "Optimization of an integrated batch production and maintenance scheduling on flow shop with two machines," *International Journal of Industrial Engineering Computations*, vol. 10, no. 2, pp. 225–238, 2019, doi: 10.5267/j.ijiec.2018.7.001.
- [32] A. Hakim Halim, and H. Ohta, "Batch-scheduling problems through the flowshop with both receiving and delivery just in time," *International Journal of Production Research*, vol. 31, no. 8, pp. 1943–1955, 1993, doi: 10.1080/00207549308956833.
- [33] R. Maulidya, Suprayogi, R. Wangsaputra, and A. H. Halim, "Batch scheduling for hybrid assembly

- differentiation flow shop to minimize total actual flow time,” *IOP Conference Series: Materials Science and Engineering*, vol. 319, no. 1, 2018, doi: 10.1088/1757-899X/319/1/012042.
- [34] P. P. Suryadhini, S. Sukoyo, S. Suprayogi, and A. H. Halim, “A batch scheduling model for a three-stage flow shop with job and batch processors considering a sampling inspection to minimize expected total actual flow time,” *Journal of Industrial Engineering and Management*, vol. 14, no. 3, pp. 520–537, 2021, doi: 10.3926/JIEM.3438.
- [35] H. Triha, A. S. Indrapriyatna, J. Jonrinaldi, and B. Yuliandra, “Algoritma penentuan ukuran batch integer pada penjadwalan flowshop satu mesin,” *Jurnal Optimasi Sistem Industri*, vol. 15, no. 1, p. 1, 2016, doi: 10.25077/josi.v15.n1.p1-15.2016.
- [36] R. Yusriski, B. Astuti, M. Ilham, and Zahedi, “Integrated batch production and multiple preventive maintenance scheduling on a single machine to minimize total actual flow time,” *IOP Conference Series: Materials Science and Engineering*, vol. 598, no. 1, 2019, doi: 10.1088/1757-899X/598/1/012083.
- [37] R. Yusriski, B. Astuti, Sukoyo, T. M. A. A. Samadhi, and A. H. Halim, “Integer batch scheduling problems for a single-machine to minimize total actual flow time,” *Procedia Manufacturing*, vol. 2, no. February, pp. 118–123, 2015, doi: 10.1016/j.promfg.2015.07.021.
- [38] A. Ageev, “Approximating the 2-machine flow shop problem with exact delays taking two values,” *Journal of Global Optimization*, vol. 76, no. 3, pp. 491–497, 2020, doi: 10.1007/s10898-019-00775-0.
- [39] B. C. Choi, and M. J. Park, “Correction to: Two-machine flow shop scheduling with convex resource consumption functions,” *Optimization Letters*, vol. 17, no. 5, pp. 1261–1262, 2023, doi: 10.1007/s11590-023-01982-1.
- [40] S. A. Mansouri, E. Aktas, and U. Besikci, “Green scheduling of a two-machine flowshop: Trade-off between makespan and energy consumption,” *European Journal of Operational Research*, vol. 248, no. 3, pp. 772–788, 2016, doi: <https://doi.org/10.1016/j.ejor.2015.08.064>.
- [41] M.-J. Park, B.-C. Choi, Y. Min, and K. M. Kim, “Two-machine ordered flow shop scheduling with generalized due dates,” *Asia-Pacific Journal of Operational Research*, vol. 37, no. 1, pp. 1–16, 2020, [Online]. Available: <https://api.semanticscholar.org/CorpusID:202126506>