





A Flow Shop Batch Scheduling Model with Pre-Processing and Time-Changing Effects to Minimize Total Actual Flow Time

Dwi Kurniawan^{1*} , Rinto Yusriski² , Mohammad Mi'radj Isnaini¹ , Anas Ma'ruf¹ , Abdul Hakim Halim¹ 

¹Institut Teknologi Nasional Bandung (Indonesia)

²Universitas Jenderal Achmad Yani (Indonesia)

*Corresponding author: dwi_kurniawan@itenas.ac.id

rinto.yusriski@lecture.unjani.ac.id, isnaini@itb.ac.id, maruf@itb.ac.id, ahakimhalim@itb.ac.id

Received: December 2023

Accepted: May 2024

Abstract:

Purpose: This paper investigates a batch scheduling problem where pre-processing is required for parts before processing, considering time-changing effects from part deterioration and operator learning-forgetting.

Design/methodology/approach: A mathematical model was developed with the decision variables of the number of batches, the number of pre-processings, batch sizes, and the schedule of processes and pre-processings to minimize total actual flow time. Different numbers of batches were gradually tried and increased until the objective function stopped improving. The minimum number of pre-processings that resulted in a feasible solution was examined at each number of batches.

Findings: Our experiment showed that: (1) A faster operator learning led to a lower optimal number of batches and a lower total actual flow time, (2) A faster part deterioration brought a higher number of batches and a higher total actual flow time, (3) The model minimized the number of pre-processings by only scheduling pre-processings before the operations at machine 1, and (4) The model divided the parts into small batches to prevent increased processing time due to part deterioration.

Research limitations: The research did not consider multi-due date and multi-item system which require pre-processings with different times and capacities.

Practical implications: Production managers should assign fast learning operators to shorter batches and faster deteriorating parts.

Originality/value: This research was the first to consider pre-processing in batch scheduling.

Keywords: scheduling, flow shop, pre-processing, time-changing effects, actual flow time

To cite this article:

Kurniawan, D., Yusriski, R., Isnaini, M.M., Ma'ruf, A., & Halim, A.H. (2024). A flow shop batch scheduling model with pre-processing and time-changing effects to minimize total actual flow time. *Journal of Industrial Engineering and Management*, 17(2), 542-561. <https://doi.org/10.3926/jiem.7134>

1. Introduction

In manufacturing, pre-processing stands as the crucial first step, meticulously preparing raw materials or components to meet stringent final product specifications and simultaneously optimizing both efficiency and quality throughout the production process. One of the most common pre-processing techniques across diverse industries is pre-heating, employed in processes such as steel slab rolling for producing steel plates or coils (Singh, Kumar & Ghose, 2024), steel welding (Wang, Li, Huang, Wang & Zhou, 2022), steel nitro-carburization (Illing, Ren & Ernst, 2023), laser beam welding of martensitic stainless steel (Panov, Naumov, Stepanov, Sokolovsky, Volokitina, Kashaev et al., 2022), stretch forming of a fiber metal laminate (Rahiminejad & Compston, 2021), injection-mold for shoe soles production (Kuo, Tasi, Huang & Tseng, 2023), and even metal crafts (Baral, Divyadarshan & Amulya, 2017; Mardi, Syarif & Syakir, 2021). Importantly, after pre-processing, parts require further processing within a specific timeframe (Demir & Previtali, 2017). Failure to complete this subsequent processing within the designated period necessitates another round of pre-processing.

Flow shop scheduling plays a crucial role in optimizing various aspects of manufacturing and production environments. However, challenges arise when part processing times change over time due to two dynamic effects: part deterioration and operator learning-forgetting (Strusevich & Rustogi, 2017). Part deterioration results from the loss of pre-processing-induced properties during flow shop processing, leading to longer processing times if delayed (Nikolaeva & Vlasov, 2017). Conversely, operators become faster with repeated processing but experience forgetting during breaks (Kurniawan, Cakravastia, Suprayogi & Halim, 2022). This learning effect, initially observed by Wright (1936), involves processing time consistently decreasing with each doubling of repetitions. Jaber and Bonney (1996) mathematically modeled this alternating process of learning and forgetting across production lots. Notably, Nembhard and Uzumeri (2000) highlighted the importance of understanding the learning effect in various manual-operation industries for setting standard times, estimating labor costs, and creating schedules. Further, Sebrina, Diawati and Cakravastia (2011) experimentally demonstrated that product complexity and takt time influence learning rates in an automotive setting, with faster learning potentially leading to higher defect rates.

Batch production, a widely adopted technique in manufacturing (Fowler & Mönch, 2022), offers the potential to improve makespan, mean flow time, and average work-in-process levels (Kalir & Sarin, 2000). However, it presents unique scheduling challenges, including determining the optimal number of batches, batch sizes, and their scheduling sequence (Halim, Miyazaki & Ohta, 1994). Common scheduling performance metrics include makespan, flow time, lateness, tardiness, and the number of tardy jobs (Bedworth & Bailey, 1987). These metrics, however, fail to address the simultaneous demands of meeting due dates and reducing inventory levels. Actual flow time, a measure defined in Halim and Ohta (1993) as the time between production start and due date, addresses these needs. With actual flow time as the performance measure, parts do not need to arrive simultaneously at the shop floor; instead, they arrive when needed and are delivered to customers by their due dates (Halim & Ohta, 1993).

Previous studies, such as those by Yusriski, Sukoyo, Samadhi and Halim (2015), Kurniawan, Raja, Suprayogi and Halim (2020) and Kurniawan et al. (2022) focused on batch scheduling models with learning and forgetting effects, neglecting the impact of part deterioration. Conversely, Sukoyo, Samadhi, Iskandar and Halim (2010) proposed a multi-item batch scheduling model with part deterioration, but their model assumed uniform processing times within a batch and did not consider learning and forgetting effects. Building upon these works, Kurniawan, Yusriski, Isnaini, Ma'ruf and Halim (2021) incorporated operator learning-forgetting from Kurniawan et al. (2020) and part deterioration from Sukoyo et al. (2010) into their model. However, this integration did not consider the crucial requirement for pre-processing steps, despite the inherent connection between pre-processing and part deterioration where required material properties are alternately gained and lost.

In batch production, pre-processing and part deterioration can occur concurrently with operator learning, creating significant scheduling complexities. Examples include wedding ring production, where metal powder undergoes pressing, sintering, and rolling (Raw, 2000), and metal crafts, where metal ingots are heated between manual hammering to maintain malleability (Baral et al., 2017; Mardi et al., 2021). To address this challenge, integrating the learning model from Kurniawan et al. (2021) with the pre-processing requirements outlined by Zhu, Katti, Qiu, Forsmark and Easton (2023) becomes crucial. Further investigation is needed to understand the impact of varying learning, deterioration, and pre-processing parameters on batch schedules.

2. Model Formulation

The situation studied in this paper can be explained as follows. There are n parts (units of a single product) to be processed in N batches, where each batch i is to be processed in k machines with a uniform routing. Before these processes, pre-processing is required by the parts to gain specific properties, which are conducted for a pre-processing time b and preceded by a pre-process setup time u . The pre-processing is required every time setup and processing times reach a processible time c after the last pre-processing. Operation of batch i at machine m , denoted by operation $L_{m,[i]}$, requires a setup time s_m and an initial processing time t_m . Time-changing effects occur during the operations, i.e., the operator at machine m learns at a learning rate l_m , and parts deteriorate at a deterioration rate δ . Subsequently, all operations must be completed no later than the due date d .

The objective function of the model is the total actual flow time F , and the decision variables are (i) the number of batches N , (ii) the number of pre-processings V , (iii) batch sizes $Q_{[i]}$, (iv) the schedule of operation $L_{m,[i]}$ ($B_{m,[i]}$), and (v) the schedule of pre-processing p ($A_{[p]}$). Also, assumptions in this study are, (1) pre-processors and machines are always available during the scheduling horizon, (2) the pre-processing time b and pre-process setup time u are independent of batch sizes, and (3) the operators have no prior learning experience. The capacity of the pre-processor is not considered in this research. Additionally, all parameters and variables used in this model are deterministic.

Indices, parameters, variables, and notations used in this paper are shown as follows.

Indices:

- p = index of pre-processings, sequenced backward from the due date ($p = 1, \dots, V$),
- m = index of machines ($m = 1, \dots, k$),
- i = index of batches, sequenced backward from the due date ($i = 1, \dots, N$).

Parameters:

- n = number of parts to be processed,
- k = number of machines,
- d = due date, calculated from $t = 0$,
- s_m = setup time per batch at machine m ,
- t_m = part processing time at machine m before learning, forgetting and part deterioration,
- b = pre-processing time,
- c = processible time,
- u = setup time before pre-processing,
- δ = part deterioration rate,
- l_m = learning gradient of the operator at machine m .

Variables:

- F = total actual flow time of all parts,
- N = number of batches,
- $Q_{[i]}$ = number of parts in batch i ,
- $L_{m,[i]}$ = operation of batch i at machine m ,
- $A_{[p]}$ = starting time of pre-processing p ,
- $B_{m,[i]}$ = starting time of operation $L_{m,[i]}$,
- $T_{m,[i]}$ = processing time of operation $L_{m,[i]}$,
- $f_{m,[i]}$ = forgetting gradient at the beginning of operation $L_{m,[i]}$,
- $a_{m,[i]}$ = equivalent number of parts of retained learning experience at the beginning of operation $L_{m,[i]}$,
- $\beta_{m,[i]}$ = equivalent number of parts of accumulated part deterioration at the beginning of operation $L_{m,[i]}$,
- $I_{m,[i]}$ = length of process interruption between operation $L_{m,[i]}$ and operation $L_{m,[i+1]}$,
- $J_{m,[i]}$ = length of process interruption between operation $L_{m-1,[i]}$ and operation $L_{m,[i]}$,
- $X_{m,[i]}$ = binary variable that equals to 1 if pre-processing is scheduled before operation $L_{m,[i]}$, otherwise equals to 0,
- $Y_{[p],m,[i]}$ = binary variable that equals to 1 if pre-processing p is scheduled before operation $L_{m,[i]}$, otherwise equals to 0,

$G_{m,[i]}$ = time spent by batch i from the last pre-processing to operation $L_{m,[i]}$,
 V = number of pre-processings required in the schedule horizon,
 $t_{[p]}$ = processing time of the x -th part as a learning function,
 $t'_{[x]}$ = processing time of the x -th part as a forgetting function.

Notations:

Pre-processing p = the p -th pre-processing from the due date.

2.1. Pre-processing

Some manufacturing processes require parts to have specific properties, which can be gained through pre-processing. An example of process and pre-processing in a single-machine batch schedule is shown in Figure 1, where the batch processes are scheduled backward from the due date, and batch 1 is the closest batch to the due date. The pre-processing lasts for h and is scheduled to be completed no later than the beginning of the batch processing. Furthermore, batches must be processed within a processable time c from the pre-processing completion. In Figure 1, pre-processing is scheduled before the process of each batch at the machine. The actual flow time of a batch is the time period from parts arrival, i.e., at the beginning of pre-processing, to the due date.

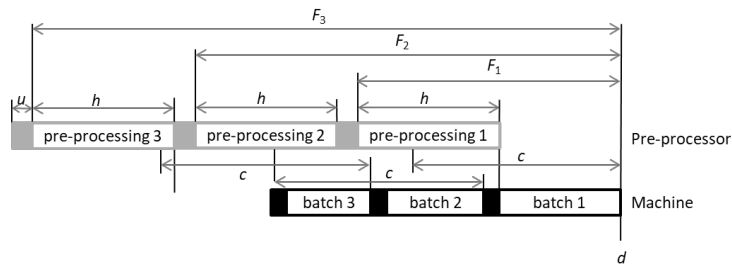


Figure 1. A single-machine backward batch schedule with pre-processing

An example of process and pre-processing in a flow shop is shown in Figure 2, where three batches are scheduled in a three-machine flow shop with a pre-processor. A pre-processing is required when the total processing time of operations, calculated backward, has exceeded c . In Figure 2, the total processing time of operations $L_{1,[1]}$, $L_{2,[1]}$ and $L_{3,[1]}$ does not exceed c , leading to only one pre-processing scheduled for batch 1 (before operation $L_{1,[1]}$). Meanwhile, the total processing time of operations $L_{1,[2]}$, $L_{2,[2]}$ and $L_{3,[2]}$ exceed c , but the total processing time of operations $L_{2,[2]}$ and $L_{3,[2]}$ does not exceed c . So, one pre-processing is scheduled before operation $L_{2,[2]}$ and another one before operation $L_{1,[2]}$. In addition, any two of operations $L_{1,[3]}$, $L_{2,[3]}$, and $L_{3,[3]}$ combined has a total processing time that is longer than c , so one pre-processing is scheduled before each operation at machines 1, 2, and 3. Bold arrows show the physical flow of each batch in the system (Figure 2).

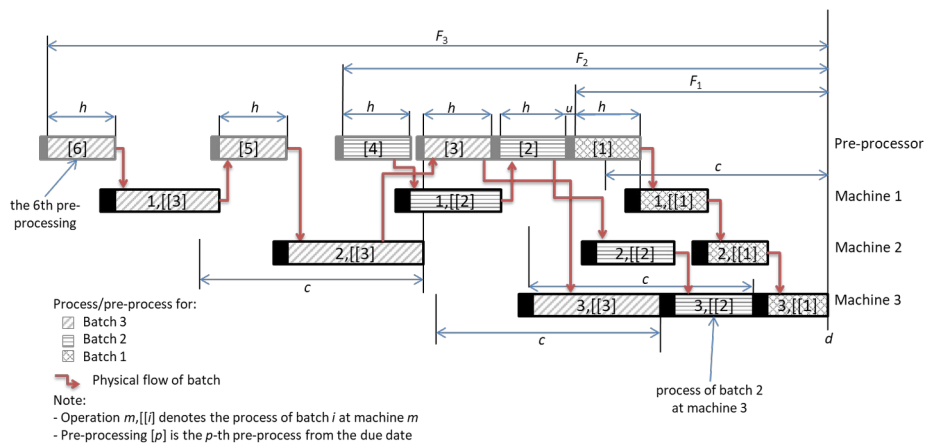


Figure 2. A flow shop batch schedule with pre-processing

The decision to determine whether a pre-processing scheduled before operation $L_{m,[i]}$ is denoted by a binary variable $X_{m,[i]}$, which equals one and zero if scheduled and not scheduled, respectively. Operations at machine 1 must be preceded by pre-processing. Meanwhile, the operations at other machines must be preceded by pre-processing when $G_{m,[i]}$ exceeds c . Variable $G_{m,[i]}$ accumulates the processing time of a batch at machines m and $m-1$, starting backward from machine k , and the accumulation is restarted when a pre-processing has been scheduled for a specific operation. The determination of a pre-processing for operation $L_{m,[i]}$ is shown in Equation (1) to (3).

$$X_{1,[i]} = 1, \quad \forall i \tag{1}$$

$$X_{m,[i]} = \begin{cases} 1, & \text{if } G_{m,[i]} / c > 1, \\ 0, & \text{elsewhere} \end{cases}, \quad m > 1, \forall i \tag{2}$$

$$G_{m,[i]} = \begin{cases} T_{m,[i]} + T_{m-1,[i]}, & m = k, i = 1 \\ B_{m,[i]} + T_{m,[i]} - B_{m-1,[i]}, & m = k, i > 1 \\ (1 - X_{m+1,[i]}) (G_{m+1,[i]} + T_{m-1,[i]}) + X_{m+1,[i]} (T_{m,[i]} + T_{m-1,[i]}), & 1 < m < k, i = 1 \\ (1 - X_{m+1,[i]}) (G_{m+1,[i]} + B_{m,[i]} - B_{m-1,[i]}) + X_{m+1,[i]} (B_{m,[i]} + T_{m,[i]} - B_{m-1,[i]}), & 1 < m < k, i > 1 \end{cases} \tag{3}$$

The required number of pre-processings in the scheduling horizon is denoted by V , where pre-processing p ($p = 1, \dots, V$) are scheduled backward. The determination whether the pre-processing before operation $L_{m,[i]}$ scheduled at the p -th pre-processing is denoted by binary variables $Y_{[p],m,[i]}$. The relationship between V , $X_{m,[i]}$ and $Y_{[p],m,[i]}$ are shown in Equation (4) and (5).

$$V = \sum_{i=1}^N \sum_{m=1}^k X_{m,[i]} \tag{4}$$

$$\sum_{p=1}^V Y_{[p],m,[i]} = X_{m,[i]} \tag{5}$$

The schedule of pre-processing p and the schedule of operation $L_{m,[i]}$ are denoted by $A_{[p]}$ and $B_{m,[i]}$, respectively. Variables $B_{m,[i]}$ are determined backward from machine k to machine 1. At machine m , $B_{m,[i]}$ are computed backward from batch 1 to batch N . At machines m ($m < k$), $B_{m,[i]}$ is scheduled based on $B_{m+1,[i]}$ while simultaneously evaluating the existence of pre-processing before operation $L_{m+1,[i]}$. Additionally, $A_{[p]}$ is determined based on the schedule of operation $L_{m,[i]}$ it precedes, and also based on $A_{[p-1]}$ for $p > 1$. The calculations of $A_{[p]}$ and $B_{m,[i]}$ are shown in Equation (6) to (11).

$$B_{k,[i]} \leq d - (i - 1)s_k + \sum_{j=1}^i T_{k,[j]}, \quad \forall i \tag{6}$$

$$B_{m,[1]} \leq B_{m+1,[1]} - T_{m,[1]} - X_{m+1,[1]} \left(B_{m+1,[1]} - \sum_{p=1}^V Y_{[p],m+1,[1]} A_{[p]} \right), \quad m < k \tag{7}$$

$$B_{m,[i]} \leq B_{m,[i-1]} - s_m - T_{m,[i]} - X_{m+1,[i]} \left(B_{m+1,[i]} - \sum_{p=1}^V Y_{[p],m+1,[i]} A_{[p]} \right), \quad m < k, i > 1 \tag{8}$$

$$B_{m,[i]} \leq B_{m+1,[i]} - T_{m,[i]} - X_{m+1,[i]} \left(B_{m+1,[i]} - \sum_{p=1}^V Y_{[p],m+1,[i]} A_{[p]} \right), \quad m < k, i > 1 \tag{9}$$

$$A_{[p]} \leq \sum_{i=1}^N \sum_{m=1}^k Y_{[p],m,[i]} B_{m,[i]} - h, \quad \forall p \tag{10}$$

$$A_{[p]} \leq A_{[p-1]} - u - h, \quad p > 1 \tag{11}$$

2.2. Time-changing Effects

According to Strusevich and Rustogi (2017), there are two types of time-changing effects, i.e., deterioration and learning, causing the processing time to increase and to decrease, respectively. Part deterioration is the decline of part condition that occurs because materials change over time. This occurs at any time after pre-processing (Demir & Previtali, 2017), causing a longer processing time when the process begins at a later time (Cheng, Kang & Ng, 2004). A linear deterioration, i.e., the increase of part processing time at a constant proportion δ of the initial processing time, was assumed in this research (Mor & Mosheiov, 2021), leading to deterioration function $D(x)$ in Equation (12),

$$D(x) = t(1 + \delta x) \tag{12}$$

where t = initial processing time (before deterioration), x = part sequence, and δ = deterioration rate of 0 or more (a higher δ means a faster deterioration, $\delta = 0$ means no deterioration).

Simultaneously with part deterioration, a learning effect occurs when an operator repeatedly performs a specific production process, resulting in a decreasing processing time as the number of repetitions increases (Jaber & Bonney, 1996). According to Wright (1936), this effect occurs following learning function $L(x)$ shown in Equation (13),

$$L(x) = tx^{-\ell} \tag{13}$$

where t = processing time before learning, x = part sequence, and ℓ = a learning gradient of 0 or more (a higher ℓ means faster learning, $\ell = 0$ means no learning).

The simultaneous occurrence of operator learning and part deterioration also causes the processing time to behave as $L(x)$ with deterioration, $L(x)(1 + \delta x)$, or $D(x)$ with learning, $D(x)x^{-\ell}$. Both formulas led to function $t_{[x]}$ as shown in Equation (14).

$$t_{[x]} = t(1 + \delta x)x^{-\ell} \tag{14}$$

Based on Equation (14), the fluctuation of processing time is shown in Figure 3. The processing time of the x -th part decreased from t and reached a minimum value at the x_{\min} -th part. After this, the processing time subsequently increased and reached t at the r -th part. Before x_{\min} , learning was more dominant to the processing time than deterioration, while after x_{\min} , deterioration was more dominant. Propositions 1 and 2 postulated the value of x_{\min} and r .

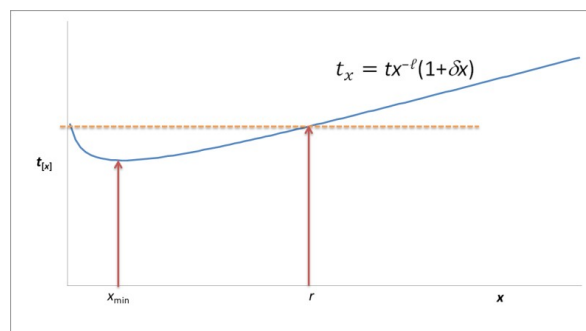


Figure 3. Fluctuation of processing time along parts in a batch

Proposition 1. Function $t_{[x]} = t(1 + \delta x)x^{-\ell}$ reaches a minimum value at the x_{\min} -th part, where $x_{\min} = \frac{\ell}{\delta(1-\ell)}$.

Proof. Function $t_{[x]} = t(1 + \delta x)x^{-\ell}$ reaches a minimum when $t'_{[x]} = t\delta x^{-\ell} - \ell t x^{-1-\ell}(1 + \delta x) = 0$, or $t\delta x^{-\ell} = \ell t x^{-1-\ell}(1 + \delta x)$, leading to $x = \frac{\ell}{\delta(1-\ell)}$, as notated in this proposition.

Proposition 2. After reaching the minimum value, function $t_{[x]} = t(1 + \delta x)x^{-\ell}$ increases and reaches t at the r -th part, where r is the bigger root of polynomial $r^\ell - \delta r = 1$.

Proof. The situation in this proposition is given by $t_{[x]} = t(1 + \delta x)x^{-\ell} = t$, which is rewritten as $r^\ell(1 + \delta x) = 1$. After some algebraic operations, $r^\ell - \delta r = 1$ is obtained and subsequently solved. The bigger root is taken as r since the smaller root is frequently negative.

Part deterioration and learning-forgetting were also explained using the batch schedule in Figure 2. Starting immediately after the completion of pre-processing 1, parts in batch 1 deteriorated continuously from operation $L_{1,[1]}$ to $L_{2,[1]}$ and $L_{3,[1]}$. Meanwhile, part deterioration in batch 2 began directly after pre-processing 4, to occur at operation $L_{1,[2]}$. This restarted after pre-processing 2, to occur subsequently at operation $L_{2,[2]}$ and continue to operation $L_{3,[2]}$. In addition, part deterioration in batch 3 started immediately after pre-processing 6, 5, and 3 to occur at each subsequent operation. Meanwhile, an operator at each machine started learning at the beginning of batch 3 and experienced forgetting during process interruptions between two consecutive batches. Forgetting started directly when the learning effect stopped and occurred during setups and machine idle times (Kurniawan et al., 2020).

Based on Jaber and Bonney (1996), Figure 4 shows a more detailed analysis of the processing time fluctuation in batch production at a machine. Suppose that the processing time before part deterioration and operator learning was t , and suppose that the process of batch i was started when the operator obtained $a_{[i]}$ unit-equivalent of the learning experience ($a_{[i+1]} = 0$ at A). Learning and deterioration occurred simultaneously along the AC curve for $T_{m,[i+1]}$, following the function in Equation (14). When the process was interrupted during the interval I , a forgetting effect occurred along CE curve, based on a forgetting function in Equation (15), which imaginarily began at B. Suppose that $Q_{[i+1]}$ parts were processed during $T_{m,[i+1]}$, and that R parts could be processed if the process continued during the interval I while learning continued to occur along CD curve based on the learning function in Equation (13).

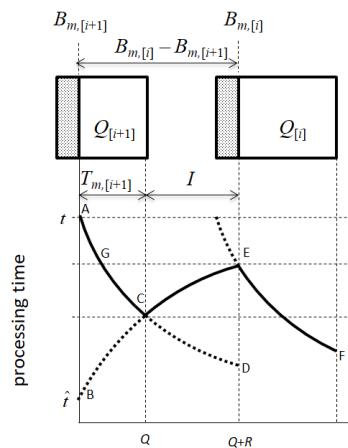


Figure 4. Learning and forgetting effects during two sequential batches

$$\hat{t}_{[x]} = \hat{t}x^f \tag{15}$$

At C, $\hat{t}_{[x]} = t_{[x]}$, or $t(1 + \delta(\alpha_{[i+1]} + Q_{[i+1]}))(\alpha_{[i+1]} + Q_{[i+1]})^{-\ell} = \hat{t}(\alpha_{[i+1]} + Q_{[i+1]})^f$, which was then used to find \hat{t} , an imaginary processing time at B where $\hat{t}_{[x]}$ started, i.e.:

$$\hat{t} = t(1 + \delta(\alpha_{[i+1]} + Q_{[i+1]}))(\alpha_{[i+1]} + Q_{[i+1]})^{-(f+\ell)} \tag{16}$$

Substituting Equation (16) to Equation (15), Equation (17) was obtained.

$$\hat{t}_{[x]} = t(1 + \delta(\alpha_{[i+1]} + Q_{[i+1]}))(\alpha_{[i+1]} + Q_{[i+1]})^{-(f+\ell)} x^f \tag{17}$$

To find I , i.e., the time required to produce R parts (if interruption did not occur), the learning function in Equation (13) was integrated along the interruption interval.

$$I = \int_{\alpha_{[i+1]} + Q_{[i+1]}}^{\alpha_{[i+1]} + Q_{[i+1]} + R} tx^{-\ell} dx = \frac{t}{1-\ell} \left[(\alpha_{[i+1]} + Q_{[i+1]} + R)^{1-\ell} - (\alpha_{[i+1]} + Q_{[i+1]})^{1-\ell} \right] \tag{18}$$

After some algebraic operations on Equation (18) to find $\alpha_{[i+1]} + Q_{[i+1]} + R$, Equation (19) was obtained.

$$\alpha_{[i+1]} + Q_{[i+1]} + R = \left[\frac{I(1-\ell)}{t} + (\alpha_{[i+1]} + Q_{[i+1]})^{1-\ell} \right]^{1/(1-\ell)} \tag{19}$$

After forgetting during CE, the operator learning experience obtained during AC was reduced from $\alpha_{[i+1]} + Q_{[i+1]}$ unit-equivalent of to $a_{[i]}$ unit-equivalent. The value of $a_{[i]}$ was found by equating $\hat{t}_{[x]}$ at E (i.e., $\hat{t}(\alpha_{[i+1]} + Q_{[i+1]} + R)$) with $\hat{t}_{[x]}$ ending at E, i.e., $t\alpha_{[i]}^{-\ell} = t(1 + \delta(\alpha_{[i+1]} + Q_{[i+1]}))(\alpha_{[i+1]} + Q_{[i+1]})^{-(\ell+f)}(\alpha_{[i+1]} + Q_{[i+1]} + R)^f$, leading to Equation (20).

$$\alpha_{[i]} = (1 + \delta(\alpha_{[i+1]} + Q_{[i+1]}))(\alpha_{[i+1]} + Q_{[i+1]})^{-(\ell+f)/\ell} (\alpha_{[i+1]} + Q_{[i+1]} + R)^{-f/\ell} \tag{20}$$

After substituting Equation (19) to Equation (20) and some algebraic operations, Equation (21) was obtained.

$$\alpha_{[i]} = (1 + \delta(\alpha_{[i+1]} + Q_{[i+1]}))(\alpha_{[i+1]} + Q_{[i+1]})^{\frac{\ell+f}{\ell}} \left[\frac{I(1-\ell)}{t} + (\alpha_{[i+1]} + Q_{[i+1]})^{1-\ell} \right]^{\frac{-f}{\ell(1-\ell)}} \tag{21}$$

According to Figure 4, the value of I was given by $I = B_{m,[i]} - B_{m,[i+1]} - T_{m,[i+1]}$. Equation (21) was applied when the operator experienced a partial forgetting after batch $i+1$ ($T_E < \ell$). When the operator experienced a total forgetting, then $T_E = t$ and $a_{[i]} = 0$, and Equation (21) was rewritten to Equation (22).

$$\alpha_{[i]} = \max \left(0, (1 + \delta(\alpha_{[i+1]} + Q_{[i+1]}))(\alpha_{[i+1]} + Q_{[i+1]})^{\frac{\ell+f}{\ell}} \left[\frac{I(1-\ell)}{t} + (\alpha_{[i+1]} + Q_{[i+1]})^{1-\ell} \right]^{\frac{-f}{\ell(1-\ell)}} \right) \tag{22}$$

The forgetting gradient f was computed at the beginning of total forgetting (when $\hat{t}_{[x]} = t$). This situation was given by $t(1 + \delta(\alpha_{[i+1]} + Q_{[i+1]}))(\alpha_{[i+1]} + Q_{[i+1]})^{-(\ell+f)}(\alpha_{[i+1]} + Q_{[i+1]} + R)^f = t$, which was used to find f as shown in Equation (23).

$$f = \frac{\ell \ln(\alpha_{[i+1]} + Q_{[i+1]}) - \ln(1 + \delta(\alpha_{[i+1]} + Q_{[i+1]}))}{\frac{1}{1-\ell} \ln \left[\frac{I}{t} (1-\ell) + (\alpha_{[i+1]} + Q_{[i+1]})^{1-\ell} \right] - \ln(\alpha_{[i+1]} + Q_{[i+1]})} \tag{23}$$

In flow shops, variables $a_{[i]}$ and f had additional indices to become $a_{m,[i]}$ and $f_{m,[i]}$, respectively.

Similar to learning, the unit-equivalent was also used to quantify accumulated part deterioration at a time point. Suppose that batch i was pre-processed and further processed at machine m and $m+1$, as shown in Figure 5. When batch i completed pre-processing, $\beta = 0$ at A. Suppose that β equals $\beta_{m,[i]}$, $\beta_{m,[i]} + Q_{[i]}$ and $\beta_{m,[i]} + Q_{[i]} + U_{m,[i]}$ at B, C and D, respectively, where $\beta_{m,[i]}$, $Q_{[i]}$ and $U_{m,[i]}$ were unit-equivalent of accumulated part deterioration. Along AB, the interval between pre-processing completion and process commencement, parts deteriorated from zero to $\beta_{m,[i]}$. The

length of interval AB equals the integration of function $D(x)$ in Equation (12) from zero to $\beta_{m,[i]}$ as shown in Equation (24).

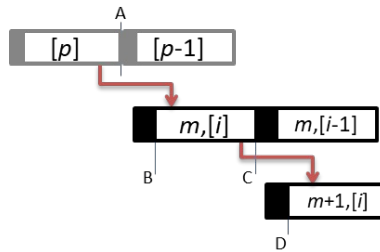


Figure 5. Part deterioration in sequential operations

$$B_{m,[i]} - \left(\sum_{p=1}^V Y_{[p],m,[i]} A_{[p]} + h \right) = \int_0^{\beta_{m,[i]}} t_m (1 + \delta_m x) dx \tag{24}$$

Substituting both sides of Equation (24), we obtained Equation (25).

$$B_{m,[i]} - \sum_{p=1}^V Y_{[p],m,[i]} A_{[p]} - h = t_m \left(\beta_{m,[i]} + \delta \frac{\beta_{m,[i]}^2}{2} \right) \tag{25}$$

After some algebraic operations, Equation (25) was rewritten as in Equation (26), a quadratic function of $\beta_{m,[i]}$, which was solved to find $\beta_{m,[i]}$ as shown in Equation (27).

$$\frac{\delta}{2} \beta_{m,[i]}^2 + \beta_{m,[i]} - \frac{B_{m,[i]} - \sum_{p=1}^V Y_{[p],m,[i]} A_{[p]} - h}{t_m} = 0 \tag{26}$$

$$\beta_{m,[i]} = \frac{-1 + \sqrt{1 + 2\delta \frac{B_{m,[i]} - \sum_{p=1}^V Y_{[p],m,[i]} A_{[p]} - h}{t_m}}}{\delta} \tag{27}$$

Similar to AB, the length of interval CD, or $B_{m+1,[i]} - B_{m,[i]} - T_{m,[i]}$, equals the integration of function $D(x)$ in Equation (12) from $\beta_{m,[i]} + Q_{[i]}$ to $\beta_{m,[i]} + Q_{[i]} + U_{m,[i]}$, as shown in Equation (28).

$$B_{m+1,[i]} - B_{m,[i]} - T_{m,[i]} = \int_{\beta_{m,[i]} + Q_{[i]}}^{\beta_{m,[i]} + Q_{[i]} + U_{m,[i]}} t_m (1 + \delta_m x) dx \tag{28}$$

$$B_{m+1,[i]} - B_{m,[i]} - T_{m,[i]} = t_m \left(U_{m,[i]} + \delta \left(\frac{(\beta_{m,[i]} + Q_{[i]} + U_{m,[i]})^2}{2} - \frac{(\beta_{m,[i]} + Q_{[i]})^2}{2} \right) \right) \tag{29}$$

which was used to obtain $\beta_{m,[i]} + Q_{[i]} + U_{m,[i]}$, the accumulated deterioration at D, as shown in Equation (30).

$$\beta_{m,[i]} + Q_{[i]} + U_{m,[i]} = \frac{-t_m + \sqrt{t_m^2 + 2t_m \delta \left(t_m (\beta_{m,[i]} + Q_{[i]}) + \frac{t_m \delta (\beta_{m,[i]} + Q_{[i]})^2}{2} + B_{m+1,[i]} - B_{m,[i]} - T_{m,[i]} \right)}}{t_m \delta} \tag{30}$$

The accumulated deterioration at D, or $\beta_{m,[i]} + Q_{[i]} + U_{m,[i]}$, was converted to $B_{m+1,[i]}$ by multiplying it with t_m/t_{m+1} . Therefore, $B_{m+1,[i]} = t_m/t_{m+1}(B_{m,[i]} + Q_{[i]} + U_{m,[i]})$, or,

$$\beta_{m+1,[i]} = \frac{-t_m + \sqrt{t_m^2 + 2t_m \delta \left(t_m(\beta_{m,[i]} + Q_{[i]}) + \frac{t_m \delta (\beta_{m,[i]} + Q_{[i]})^2}{2} + B_{m+1,[i]} - B_{m,[i]} - T_{m,[i]} \right)}}{t_{m+1} \delta} \quad (31)$$

Different from AB and CD, the length of interval BC ($T_{m,[i]}$) equals the integration of $t_{[x]}$ in Equation (14) with its deterioration and learning bounds. Since deterioration and learning occurred with different bounds, the part sequences in Equation (14) were separated into x and y for deterioration and learning, respectively. Therefore, $t_{[x]}$ was rewritten into $t_{[x,y]} = t(1 + \delta x)y^{-\ell}$, and integrated $t_{[x,y]}$ along with its deterioration and learning bounds as shown in Equation (32), leading to Equation (33).

$$T_{m,[i]} = \int_{\alpha_{[i]}}^{\alpha_{[i]} + Q_{[i]}} \int_{\beta_{m,[i]}}^{\beta_{m,[i]} + Q_{[i]}} t(1 + \delta x)y^{-\ell} dx dy \quad (32)$$

$$T_{m,[i]} = t \frac{(\alpha_{[i]} + Q_{[i]})^{1-\ell} - \alpha_{[i]}^{1-\ell}}{1-\ell} \left(Q_{[i]} + \frac{\delta(Q_{[i]}^2 + 2Q_{[i]}\beta_{m,[i]})}{2} \right) \quad (33)$$

2.3. Conceptual Framework

The relationships between parameters c, u, b, δ and ℓ with decision variables $N, Q_{[i]}$, and F were hypothesized in a model conceptual framework (Figure 6). Pre-process setup time and pre-processing time were hypothesized to behave like setup time and processing time, for instance, having a positive and negative relationship with total actual flow time and the number of batches, respectively (Mekler, 1993). However, the opposite properties with pre-processing time were predicted as processable time limits the length of processing, such as a negative and positive relationship with total actual flow time and the number of batches, respectively. Furthermore, a higher deterioration rate was supposed to have the same effects with pre-processing time since it led to a longer operation. According to Propositions 3 and 4 in Kurniawan et al. (2020), the learning rate was hypothesized to have a negative relationship with total actual flow time and the number of batches. Finally, more batches reduced the size of batches and vice versa.

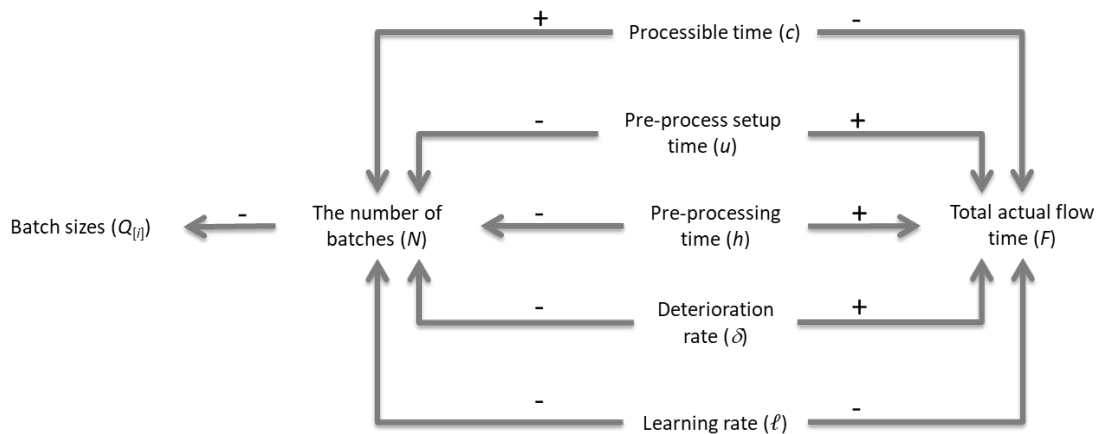


Figure 6. Conceptual framework of the model

2.4. The Proposed Model

Sections 2.2 and 2.3 explained the basic concept of pre-processing, part deterioration, and operator learning, as well as their application in batch scheduling. These led to the flow shop batch scheduling model with pre-processing, part deterioration, and operator learning-forgetting to minimize total actual flow time as formulated in Model 1.

Model 1

$$\text{Minimize } F = \sum_{i=1}^N \left(d - \sum_{p=1}^V Y_{[p],1,[i]} A_{[p]} \right) Q_{[i]} \quad (34)$$

subject to

$$B_{k,[i]} \leq d - (i - 1)s_k + \sum_{j=1}^i T_{k,[j]}, \quad \forall i \quad (35)$$

$$B_{m,[1]} \leq B_{m+1,[1]} - T_{m,[1]} - X_{m+1,[1]} \left(B_{m+1,[1]} - \sum_{p=1}^V Y_{[p],m+1,[1]} A_{[p]} \right), \quad m < k \quad (36)$$

$$B_{m,[i]} \leq B_{m,[i-1]} - s_m - T_{m,[i]} - X_{m+1,[i]} \left(B_{m+1,[i]} - \sum_{p=1}^V Y_{[p],m+1,[i]} A_{[p]} \right), \quad m < k, i > 1 \quad (37)$$

$$B_{m,[i]} \leq B_{m+1,[i]} - T_{m,[i]} - X_{m+1,[i]} \left(B_{m+1,[i]} - \sum_{p=1}^V Y_{[p],m+1,[i]} A_{[p]} \right), \quad m < k, i > 1 \quad (38)$$

$$A_{[p]} \leq \sum_{i=1}^N \sum_{m=1}^k Y_{[p],m,[i]} B_{m,[i]} - h, \quad \forall p \quad (39)$$

$$A_{[p]} \leq A_{[p-1]} - u - h, \quad p > 1 \quad (40)$$

$$A_{[V]} \geq 0 \quad (41)$$

$$\sum_{p=1}^V Y_{[p],m,[i]} = X_{m,[i]}, \quad \forall m, \forall i \quad (42)$$

$$V = \sum_{i=1}^N \sum_{m=1}^k X_{m,[i]} \quad (43)$$

$$X_{1,[i]} = 1, \quad \forall i \quad (44)$$

$$X_{m,[i]} = \begin{cases} 1, & \text{if } G_{m,[i]} / c > 1 \\ 0, & \text{elsewhere} \end{cases}, \quad m > 1, \forall i \quad (45)$$

$$G_{m,[i]} = \begin{cases} T_{m,[i]} + T_{m-1,[i]}, & m = k, i = 1 \\ B_{m,[i]} + T_{m,[i]} - B_{m-1,[i]}, & m = k, i > 1 \\ (1 - X_{m+1,[i]}) (G_{m+1,[i]} + T_{m-1,[i]}) + X_{m+1,[i]} (T_{m,[i]} + T_{m-1,[i]}), & 1 < m < k, i = 1 \\ (1 - X_{m+1,[i]}) (G_{m+1,[i]} + B_{m,[i]} - B_{m-1,[i]}) + X_{m+1,[i]} (B_{m,[i]} + T_{m,[i]} - B_{m-1,[i]}), & 1 < m < k, i > 1 \end{cases} \quad (46)$$

$$T_{m,[i]} = t_m \frac{(\alpha_{m,[i]} + Q_{[i]})^{1-\ell_m} - \alpha_{m,[i]}^{1-\ell_m}}{1-\ell_m} \left(Q_{[i]} + \frac{\delta(Q_{[i]}^2 + 2Q_{[i]}\beta_{m,[i]})}{2} \right), \quad \forall m, \forall i \quad (47)$$

$$T_{m,[i]} \leq c, \quad \forall m, \forall i \quad (48)$$

$$\alpha_{m,[i]} = \begin{cases} 0, & \forall m, i = N \\ \lceil \ell_m \rceil \max \left(0, (1 + \delta(\alpha_{m,[i+1]} + Q_{[i+1]})) [\alpha_{m,[i+1]} + Q_{[i+1]}]^{\frac{\ell_m + \ell_m(i)}{\ell_m}} \left[\frac{I_{m,[i]}}{t_m} (1 - \ell_m) + (\alpha_{m,[i+1]} + Q_{[i+1]})^{1-\ell_m} \right]^{\frac{-\ell_m(i)}{\ell_m(1-\ell_m)}} \right), & \forall m, i < N \end{cases} \quad (49)$$

$$\beta_{m,[i]} = \lceil \delta \rceil \cdot \left[X_{m,[i]} \frac{-1 + \sqrt{1 + 2\delta \frac{B_{m,[i]} - \sum_{p=1}^V Y_{[p],m,[i]} A_{[p]} - h}{t_m}}}{\delta} + \right. \\ \left. (1 - X_{m,[i]}) \frac{-t_{m-1} + \sqrt{t_{m-1}^2 + 2t_{m-1}\delta \left(t_{m-1}(\beta_{m-1,[i]} + Q_{[i]}) + \frac{t_{m-1}\delta(\beta_{m-1,[i]} + Q_{[i]})^2}{2} + J_{m,[i]} \right)}}{t_m\delta} \right], \quad \forall m, \forall i \quad (50)$$

$$f_{m,[i]} = \frac{\ell_m \ln(\alpha_{m,[i+1]} + Q_{[i+1]}) - \ln(1 + \delta(\alpha_{m,[i+1]} + Q_{[i+1]}))}{\frac{1}{1-\ell_m} \ln \left[\frac{I_{m,[i]}}{t_m} (1 - \ell_m) + (\alpha_{m,[i+1]} + Q_{[i+1]})^{1-\ell_m} \right] - \ln(\alpha_{m,[i+1]} + Q_{[i+1]})}, \quad \forall m, i < N, \ell_m > 0 \quad (51)$$

$$I_{m,[i]} = B_{m,[i]} - B_{m,[i+1]} - T_{m,[i+1]}, \quad \forall m, i < N \quad (52)$$

$$J_{m,[i]} = B_{m,[i]} - B_{m-1,[i]} - T_{m-1,[i]}, \quad m > 1, \forall i \quad (53)$$

$$\sum_{i=1}^N Q_{[i]} = n \quad (54)$$

$$Q_{[i]} > 0, \quad 1 \leq N \leq n, \quad i = 1, \dots, N. \quad (55)$$

The objective function (34) was minimizing total actual flow time of all parts, i.e., the total time spent by all parts in the shop floor, from the batch arrival for pre-processing to the due date. The schedule of operation $L_{m,[i]}$ and the schedule of pre-processing p were determined in constraints (35) to (38) and in constraints (39) to (41), respectively. Furthermore, constraints (35) and (36) computed the schedule of operation $L_{m,[i]}$ for $m = k$ and $m < k$, respectively. Meanwhile, constraints (37) and (38) determined the schedule of operation $L_{m,[i]}$ for $i > 1$ and $m < k$. Constraint (39) scheduled pre-processing p based on operation $L_{m,[i]}$ following it, while constraint (40) ensured that pre-processing p ($p > 1$) did not overlap with pre-processing $p-1$. In addition, constraint (41) scheduled pre-processing V not to be earlier than time zero.

Based on Section 2.1, mathematical models for pre-processing were applied in constraints (42) to (46). Constraint (42) explained the relationship between $X_{m,[i]}$ and $Y_{[p],m,[i]}$, i.e., if a pre-processing was scheduled before operation $L_{m,[i]}$, the pre-processing must be scheduled to exactly one pre-processing p . Constraint (43) defined the number of pre-processings along the scheduling horizon. Constraint (44) explained that operations at machine 1 must be

preceded by pre-processing, while pre-processing determination for operations at other machines were based on the cumulative processing time defined in constraint (46).

Meanwhile, mathematical models for part deterioration and operator learning developed in Section 4 were used in constraints (47) to (53). Constraint (47) defined the processing time of operation $L_{m,[j]}$ used in constraint (35) to (38), while constraint (48) limited the processing time of operation $L_{m,[j]}$ not to exceed c . Constraints (49) and (50) determined the accumulated unit-equivalent of learning experience and part deterioration at the beginning of operation $L_{m,[j]}$, respectively, while constraint (51) computed the forgetting gradient used in constraint (49). The length of interruption for learning and deterioration were computed in constraints (52) and (53), respectively. Additionally, constraint (54) stated that the number of parts in all batches must equal the total number of parts, and constraint (55) stated that batch sizes must be positive and that the number of batches must be from one to the number of parts.

The existence of processable time c limiting operation's processing time may lead to the absence of a feasible solution at low numbers of batches due to the long processing time. Based on parameters, the existence of a feasible solution can be determined at $N = 1$, as postulated in Proposition 3. If $N > 1$, the existence of a feasible solution will not be able to be identified from the parameters.

Proposition 3: When $t_m \frac{n^{1-\ell_m}}{1-\ell_m} \left(n + \frac{\delta n^2}{2} \right) > c$ for all m , there is no feasible solution for the problem when $N = 1$.

Proof: Consider the operation's processing time defined in constraint (47). If $N = 1$, then $Q_{[1]} = n$, $\alpha_{m,[1]} = 0$ and $\beta_{m,[1]} = 0$. Substituting these to constraint (47) we obtain:

$$T_{m,[1]} = t_m \frac{n^{1-\ell_m}}{1-\ell_m} \left(n + \frac{\delta n^2}{2} \right).$$

Therefore, when $t_m \frac{n^{1-\ell_m}}{1-\ell_m} \left(n + \frac{\delta n^2}{2} \right) > c$, there is no feasible solution to the problem, as stated in this proposition. □

The required number of pre-processings during the scheduling horizon ranges from N to Nk . Scheduling N pre-processings means that only operations at machine 1 are preceded by pre-processing, while scheduling Nk pre-processings means that all operations are preceded by pre-processing. To minimize the objective function, the model schedules as few pre-processings as possible, as stated in Proposition 4.

Proposition 4: If a problem has a solution F at N and V ($N \leq V \leq Nk$), and its solution at N and V' ($V' > V$ and $N \leq V' \leq Nk$) is F' , then $F' \geq F$.

Proof: Suppose that there is a schedule as shown in Figure 7, with the solution $F = Q_a F_a + Q_b F_b$. In the schedule, operation $L_{p,[a]}$ is preceded by pre-processing r , while operation $L_{p,[b]}$ is preceded by pre-processing s . Suppose also that the pre-processing time is b . If we schedule additional pre-processing before operation $L_{q,[a]}$, operation $L_{p,[a]}$ will shift left, which will cause pre-processing r to shift left, causing F_a to increase. Similarly, if we schedule additional pre-processing before operation $L_{q,[b]}$, operation $L_{p,[b]}$ will shift left, which will cause pre-processing s to shift left, causing F_b to increase. If two additional pre-processings are scheduled before operation $L_{q,[a]}$ and $L_{q,[b]}$, F_a and F_b will increase further. It is clear that any additional pre-processing will increase F_a or F_b or both, meaning that F will also increase.

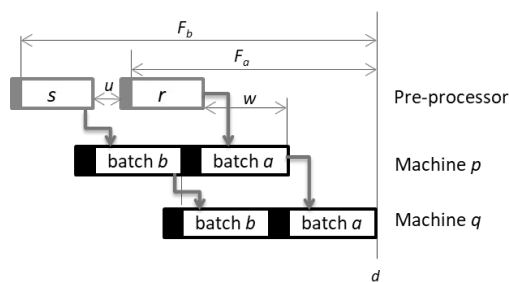


Figure 7. An example of schedule for Proposition 4

3. Solution Method

The problem formulated in Model 1 was classified as a mixed integer non linear programming (MINLP), and was unsolvable when N or V is unknown. We needed to relax N and V from decision variables to parameters, so the problem can be solved in $O(bc \cdot 2^{n/\varepsilon-1})$ time, where ε is the desired accuracy of $Q_{[t]}$ value ($\varepsilon \in \{1, 0.1, 0.001, \dots\}$) (Kurniawan et al., 2022). The problem formulated in Model 1 is NP-hard since it can be reduced to a flow shop batch scheduling problem without operator assignment and without learning and forgetting effects which is classified as NP-hard in Mortezaei and Zulkifli (2013).

It was necessary to try all possible N and V values ($1 \leq N \leq n$ and $N \leq V \leq Nk$), and find the best objective function value to find the optimal solution for the problem. However, it was timely-inefficient as the computation time increased rapidly with N . Therefore, a solution procedure adapted from Kurniawan et al. (2020) was applied to solve Model 1. Using Proposition 3, several N values were analyzed, starting from $N = 2$ if Proposition 3 was satisfied or from $N = 1$ otherwise. Then, N was increased one by one until F stopped improving. Starting from $V = N$, Proposition 4 was also applied at each N by not increasing V after a feasible solution was available at the current V . The last improving F was set as the optimal solution. The complete solution method for Model 1 is explained in Algorithm 1 and shown in Figure 8.

Algorithm 1

- Step 1. Set parameters $n, k, s_m, t_m, \ell_m, \delta, h, c, u$ and d . Go to Step 2.
- Step 2. If Proposition 3 is fulfilled, set $N = 2$. Otherwise, set $N = 1$. Go to Step 3.
- Step 3. Set $V = N$, go to Step 4.
- Step 4. Solve Model 1, find F , the best solution for the current N and V . Go to Step 5.
- Step 5. If a feasible solution exists for the problem, go to Step 6. Otherwise, go to Step 7.
- Step 6. If $F < F^*$ or F^* has not been set, set F as F^* , go to Step 8. Otherwise, go to Step 9.
- Step 7. If $V = Nk$, go to Step 8. Otherwise, set $V = V + 1$, return to Step 4.
- Step 8. If $N = n$, go to Step 9. Otherwise, set $N = N + 1$, return to Step 3.
- Step 9. If F^* has a value, set F^* as the optimal solution. Otherwise, no feasible solution is available for the problem.

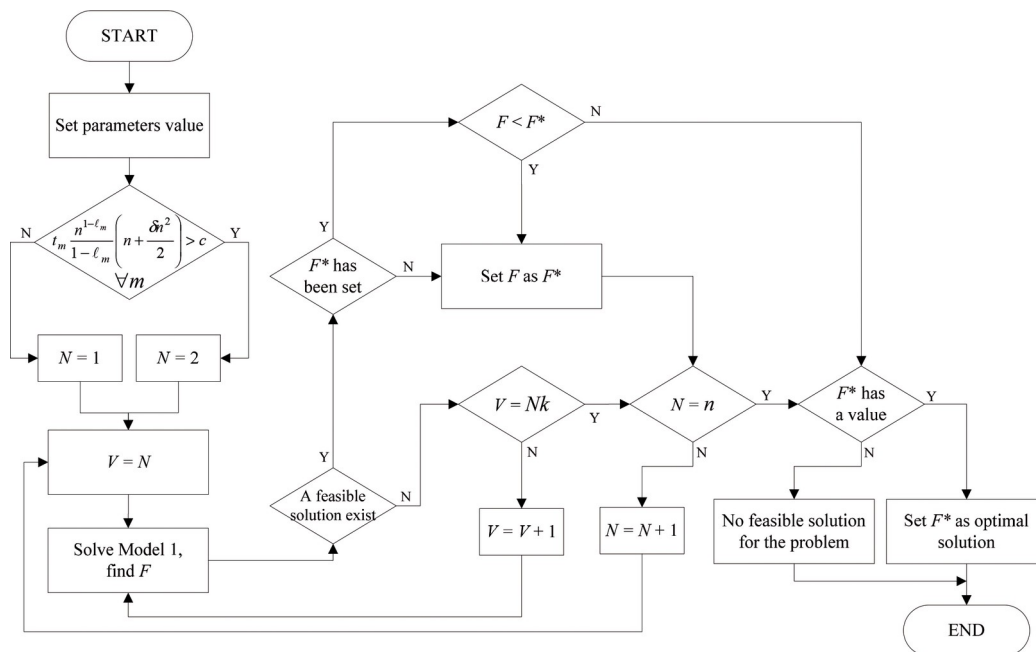


Figure 8. A solution method for solving Model 1

Step 4 in Algorithm 1 was performed using Lingo, as it is a comprehensive optimization tool, capable to solve linear or non-linear models efficiently (Goodarzi, Ziaei & Hosseinipour, 2014).

4. Numerical Experiences

Model 1 and Algorithm 1 were applied in 5 experimental scenarios, utilizing 16 datasets with varying parameter values. Each scenario aimed to perform a simultaneous sensitivity analysis during the experiment. This involved changing one parameter at a time while keeping all others constant, and observing the effects on decision variables. The parameters varied across the five scenarios: c , h , u , δ , and ℓ_m (note that ℓ is a vector containing ℓ_m). All other parameters remained consistent across all datasets: $k = 3$, $n = 10$, $s = (22\ 20\ 27)$, and $t = (3\ 5\ 4)$. Additionally, a high due date of $d = 10,000$ was chosen to prevent data infeasibility at low N values, as makespan values tended to be high at low N due to the absence of operation overlap.

Table 1 and 2 show the value of parameters and the optimal solution in the datasets, respectively.

Scenario 1: changing c			
Dataset 1 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.01$; $b = 20$; $c = 100$;	Dataset 2 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.01$; $b = 20$; $c = 70$;	Dataset 3 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.01$; $b = 20$; $c = 50$;	Dataset 4 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.01$; $b = 20$; $c = 40$;
Scenario 2: changing h			
Dataset 1 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.01$; $b = 20$; $c = 100$;	Dataset 5 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.01$; $b = 30$; $c = 100$;	Dataset 6 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.01$; $b = 40$; $c = 100$;	Dataset 7 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.01$; $b = 50$; $c = 100$;
Scenario 3: changing u			
Dataset 1 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.01$; $b = 20$; $c = 100$;	Dataset 8 $\ell = (0.33\ 0.31\ 0.36)$; $u = 6$; $\delta = 0.01$; $b = 20$; $c = 100$;	Dataset 9 $\ell = (0.33\ 0.31\ 0.36)$; $u = 10$; $\delta = 0.01$; $b = 20$; $c = 100$;	Dataset 10 $\ell = (0.33\ 0.31\ 0.36)$; $u = 15$; $\delta = 0.01$; $b = 20$; $c = 100$;
Scenario 4: changing δ			
Dataset 11 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0$; $b = 20$; $c = 100$;	Dataset 12 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.005$; $b = 20$; $c = 100$;	Dataset 1 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.010$; $b = 20$; $c = 100$;	Dataset 13 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.015$; $b = 20$; $c = 100$;
Scenario 5: changing ℓ_m			
Dataset 1 $\ell = (0.33\ 0.31\ 0.36)$; $u = 2$; $\delta = 0.010$; $b = 20$; $c = 100$;	Dataset 14 $\ell = (0.24\ 0.22\ 0.24)$; $u = 2$; $\delta = 0.010$; $b = 20$; $c = 100$;	Dataset 15 $\ell = (0.11\ 0.13\ 0.18)$; $u = 2$; $\delta = 0.010$; $b = 20$; $c = 100$;	Dataset 16 $\ell = (0\ 0\ 3)$; $u = 2$; $\delta = 0.010$; $b = 20$; $c = 100$;

Table 1. Parameters of datasets

Based on Table 2, several results were observed. First, the number of batches decreased with an increase in pre-processing time, pre-process setup time, deterioration rate or learning rate. However, the number of batches increased when the processable time increased. Second, total actual flow time increased with the increase of pre-processing time, pre-process setup time or deterioration rate. Conversely, total actual flow time decreased when the processable time or learning rate increased. These two results were consistent with Proposition 1 and 2 in Kurniawan et al. (2020), which stated that the faster operators learn, the lower the optimal number of batches and the lower the total actual flow time. However, the situation in these propositions did not appear consistently in Table 2 since Kurniawan et al. (2020) did not consider part deterioration in their model.

Third, the model resulted in small batches, with 5 to 7 batches, sized 1 to 3 in the optimal solutions. This was because the model prevented the processing time from increasing after the x_{\min} -th part as indicated in Figure 3 by forming small batches. Fourth, in all optimal solutions, pre-processings were only scheduled before operations at machine 1. This finding was consistent with Proposition 4, which indicated that scheduling additional pre-processing when a feasible solution already existed at a specific number of pre-processings increased the

objective function value. In addition, batch sizes in Table 2 did not consistently follow shortest processing time (SPT) nor longest processing time (LPT) rules. This was because SPT and LPT usually appear in single-machine systems, not in flow shops. All findings in Table 2 confirmed the conceptual framework hypothesized in Figure 6.

Scenario	Data Set	Parameter changed	Decision variables						
			N	V	$Q_{[i]} (i = 1, \dots, N)$	$X_{m,[i]}$			F
						m=1	m=2	m=3	
1	1	$c = 100$	6	6	1.97; 1.90; 1.82; 1.71; 1.56; 1.03	111111	000000	000000	1297.4
	2	$c = 70$	6	6	2.01; 1.96; 1.89; 1.80; 1.23; 1.10	111111	000000	000000	1372.3
	3	$c = 50$	6	6	2.01; 1.96; 1.89; 1.80; 1.23; 1.10	111111	000000	000000	1372.3
	4	$c = 40$	5	5	2.14; 2.12; 2.10; 2.06; 1.57	11111	00000	00000	1352.1
2	1	$b = 20$	6	6	1.97; 1.90; 1.82; 1.71; 1.56; 1.03	111111	000000	000000	1297.4
	5	$b = 30$	5	5	2.14; 2.12; 2.10; 2.06; 1.57	11111	00000	00000	1452.1
	6	$b = 40$	5	5	1.82; 2.02; 2.14; 2.21; 1.81	11111	00000	00000	1573.8
	7	$b = 50$	5	5	1.65; 2.19; 2.49; 2.64; 1.03	11111	00000	00000	1771.0
3	1	$\mu = 2$	6	6	1.97; 1.90; 1.82; 1.71; 1.56; 1.03	111111	000000	000000	1297.4
	8	$\mu = 6$	5	5	2.14; 2.12; 2.10; 2.06; 1.57	11111	00000	00000	1352.1
	9	$\mu = 10$	5	5	2.14; 2.12; 2.10; 2.06; 1.57	11111	00000	00000	1352.1
	10	$\mu = 15$	5	5	2.14; 2.12; 2.10; 2.06; 1.57	11111	00000	00000	1352.1
4	11	$\delta = 0.000$	6	6	1.93; 1.88; 1.81; 1.72; 1.60; 1.07	111111	000000	000000	1291.3
	12	$\delta = 0.005$	6	6	1.95; 1.89; 1.82; 1.72; 1.58; 1.05	111111	000000	000000	1294.4
	1	$\delta = 0.010$	6	6	1.97; 1.90; 1.82; 1.71; 1.56; 1.03	111111	000000	000000	1297.4
	13	$\delta = 0.015$	5	5	2.16; 2.14; 2.11; 2.05; 1.54	11111	00000	00000	1356.5
5	16	$\ell = (0.00 \ 0.00 \ 0.03)$	7	7	1.73; 1.67; 1.59; 1.49; 1.34; 1.17; 1.00	1111111	0000000	0000000	1397.3
	15	$\ell = (0.11 \ 0.13 \ 0.18)$	7	7	1.77; 1.70; 1.60; 1.46; 1.28; 1.09; 1.11	1111111	0000000	0000000	1351.0
	14	$\ell = (0.24 \ 0.22 \ 0.24)$	7	7	1.81; 1.73; 1.62; 1.46; 1.26; 1.06; 1.05	1111111	0000000	0000000	1330.9
	1	$\ell = (0.33 \ 0.31 \ 0.36)$	6	6	1.97; 1.90; 1.82; 1.71; 1.56; 1.03	111111	000000	000000	1297.4

Table 2. The optimal solution for each datasets

To better understand the proposed model and algorithm, the implementation of Algorithm 1 to Dataset 5 is shown in Figure 9, where Box A shows the value of the objective function F at various N and V values. After evaluating some N values, starting from $N = 1$, the model started to have a feasible solution at $N = 2$ and $V = 4$. There was no feasible solution when $N = 1$ because constraint (48), $T_{m,[i]} \leq c (\mathbf{V}m, \mathbf{V}i)$, was not satisfied at all V values. From $N = 2$, the objective function improved (decreased) until $N = 5$, and subsequently increased at $N = 6$. At this point, the calculation was stopped, and the objective function value at $N = 5$ was set as the optimal solution.

To explain the absence of a feasible solution of Dataset 5 at $N = 1$, constraint (48) was temporarily relaxed at $N = 1$, and the values of $T_{m,[i]}$ (while constraint (48) relaxed) at $V = 3$ (the maximum V at $N = 1$) were shown in Box B of Figure 9 (since processable time c limited accumulated processing time after one pre-processing, a feasible solution existed most likely at the maximum V of each N). Box B showed that all $T_{m,[i]}$ values at $N = 1$ exceed c , therefore, breaching constraint (48).

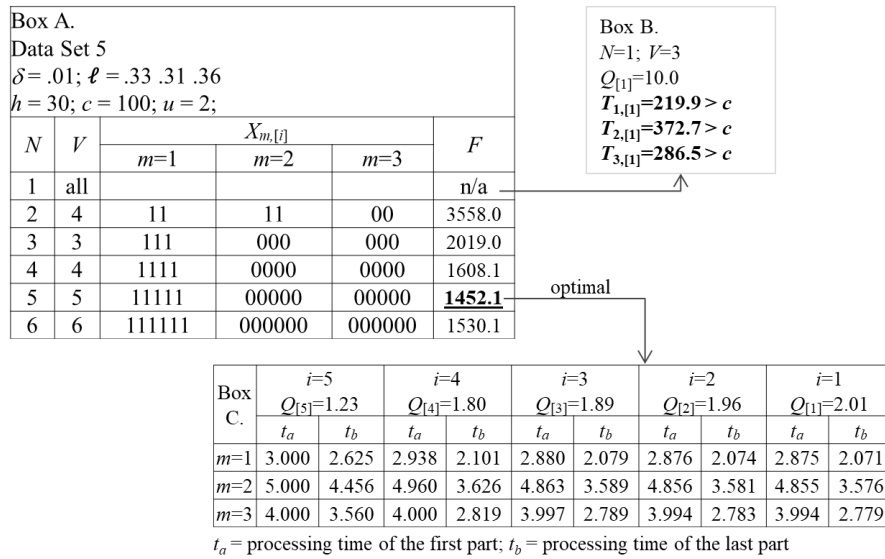


Figure 9. Detailed solution of Dataset 5

Box C (Figure 9) shows the fluctuation of processing time within each operation $L_{m,[i]}$ at the optimal solution of Dataset 5. It is shown that, first, in all operations, the processing times decreased, such as from 3.000 to 2.625 in operation $L_{1,[1]}$. This explained the reason the model formed small batches, i.e., to prevent the processing time from increasing after the x_{min} -th part, as shown in Figure 3. Second, the phenomenon of operator forgetting during process interruption was observable in the increasing processing time, such as from 3.560 to 4.000 (total forgetting) and from 2.819 to 3.997 (partial forgetting) at machine 3. Meanwhile, parts continued to deteriorate during batch transfers from a machine to the next. The accumulated deterioration influenced the processing time of the first part of operations at machines 2 and 3.

Figure 10 shows the resulting Gantt-chart for Dataset 2, where small batches resulted in short processes in the schedule. These operations were quite short, as the setups were longer than the processes.

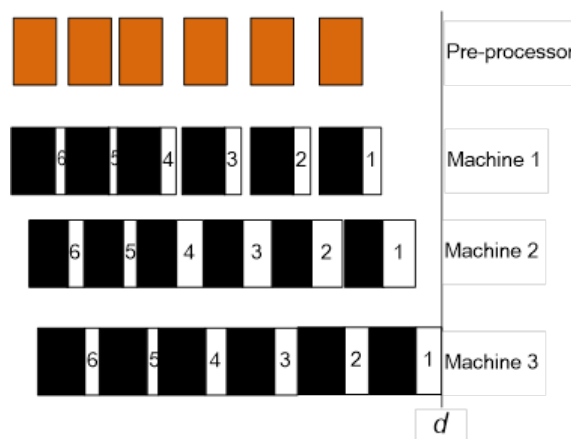


Figure 10. Gantt-chart for the optimal solution of Dataset 2

Since the heuristic procedure used in this paper stopped at the first N where F did not improve, and further improvement of F was still possible at higher N values, the resulting solution was not guaranteed to be globally optimal. However, the procedure was argued to be a good choice to save computation time while achieving a reasonably good solution. We also argue that our experiment (as shown in Tables 1 and 2) was sufficient to explore the effect of each parameter on decision variables, as additional datasets would not bring different results.

Findings in this study have two managerial implications. First, as postulated in Nembhard and Uzumeri (2000), production managers should assign fast learning operators to shorter batches and avoid assigning them highly various tasks. Second, due to the increasing F when δ increases, fast learning operators are better to handle fast deteriorating parts. Nevertheless, slow learning operators might be better in handling highly customized tasks.

This study's findings have two key managerial implications. First, supporting Nembhard and Uzumeri (2000), production managers should assign fast-learning operators to shorter batches and avoid highly varied tasks. Second, due to the positive correlation between F and δ , fast-learning operators are better suited for tasks involving rapidly deteriorating parts. However, slow-learning operators may be more effective in handling highly customized tasks. Matching operators' learning speed to task characteristics enhances production efficiency, product quality, cost savings, and empowers managers with data-driven decisions for a more effective workforce.

5. Concluding Remarks

This research developed a batch scheduling model for flow shops with pre-processing, operator learning-forgetting, and part deterioration, aiming to minimize the total actual flow time. An algorithm was subsequently developed to solve the model. This algorithm operates by analyzing different numbers of batches, starting with one or two based on parameter values. The number of batches is then incrementally increased until the objective function value plateaus. At each number of batches, the minimum number of pre-processings that leads to a feasible solution is determined. The results demonstrate that faster operator learning leads to both a lower number of batches and a lower total actual flow time. Conversely, faster part deterioration results in a higher number of batches and a higher total actual flow time. Additionally, the model identifies the least number of pre-processings necessary in the optimal solution. Building upon this work, future studies should consider extending the model to multi-due date and multi-item systems where pre-processing times and capacities vary, investigate advanced algorithms for more complex scenarios, and validate the model in real-world settings while accounting for uncertainty and dynamic environments.

Acknowledgement

This article builds upon our previous work published in the Proceedings of the 2nd Asia Pacific Conference on Industrial Engineering and Operations Management (<http://ieomsociety.org/proceedings/2021indonesia/97.pdf>). The key difference in this submission compared to the published conference paper is that we introduce a scenario where some parts require pre-processing to acquire specific properties before being processed in a flow shop.

Declaration of Conflicting Interests

The authors of this paper declare no conflict of interest related to the research and the publication of the paper.

Funding

This research was funded by Manufacturing System Research Group (KKSM) 2021 of Bandung Institute of Technology (ITB), Indonesia.

References

- Baral, B., Divyadarshan, C.S., & Amulya, S. (2017). Metal Crafts - Balakati, Orissa. Available at: <http://www.dsource.in/resource/metal-crafts-balakati-orissa>
- Bedworth, D.D., & Bailey, J.E. (1987). *Integrated production control system: Management, analysis, design*. John Wiley & Sons.
- Cheng, T., Kang, L., & Ng, C. T. (2004). Due-date assignment and single machine scheduling with deteriorating jobs. *Journal of the Operational Research Society*, 55(2), 198-203. <https://doi.org/10.1057/palgrave.jors.2601681>
- Demir, A.G., & Previtali, B. (2017). Investigation of remelting and preheating in SLM of 18Ni300 maraging steel as corrective and preventive measures for porosity reduction. *The International Journal of Advanced Manufacturing Technology*, 93(5), 2697-2709. <https://doi.org/10.1007/s00170-017-0697-z>

- Fowler, J.W., & Mönch, L. (2022). A survey of scheduling with parallel batch (p-batch) processing. *European Journal of Operational Research*, 298(1), 1-24. <https://doi.org/10.1016/j.ejor.2021.06.012>
- Goodarzi, E., Ziaei, M., & Hosseinipour, E.Z. (2014). Optimization analysis using LINGO and MATLAB. In *Introduction to Optimization Analysis in Hydrosystem Engineering* (149-193). Springer. https://doi.org/10.1007/978-3-319-04400-2_5
- Halim, A.H., Miyazaki, S., & Ohta, H. (1994). Batch-scheduling problems to minimize actual flow times of parts through the shop under JIT environment. *European Journal of Operational Research*, 72, 529-544. [https://doi.org/10.1016/0377-2217\(94\)90421-9](https://doi.org/10.1016/0377-2217(94)90421-9)
- Halim, A.H., & Ohta, H. (1993). Batch-scheduling problems through the flowshop with both receiving and delivery just in time. *The International Journal Of Production Research*, 31(8), 1943-1955. <https://doi.org/10.1080/00207549308956833>
- Illing, C., Ren, Z., & Ernst, F. (2023). Low-Temperature Carburization: Ex Situ Activation of Austenitic Stainless Steel. *Metals*, 13(2), 335. <https://doi.org/10.3390/met13020335>
- Jaber, M.Y., & Bonney, M. (1996). Production breaks and the learning curve: the forgetting phenomenon. *Applied Mathematical Modelling*, 2(20), 162-169. [https://doi.org/10.1016/0307-904X\(95\)00157-F](https://doi.org/10.1016/0307-904X(95)00157-F)
- Kalir, A.A., & Sarin, S.C. (2000). Evaluation of the potential benefits of lot streaming in flow-shop systems. *International Journal of Production Economics*, 66(2), 131-142. [https://doi.org/10.1016/S0925-5273\(99\)00115-2](https://doi.org/10.1016/S0925-5273(99)00115-2)
- Kuo, C.C., Tasi, Q.Z., Huang, S.H., & Tseng, S.F. (2023). Enhancing surface temperature uniformity in a liquid silicone rubber injection mold with conformal heating channels. *Materials*, 16(17), 5739. <https://doi.org/10.3390/ma16175739>
- Kurniawan, D., Cakravastia, A., Suprayogi, S., & Halim, A.H. (2022). A batch scheduling and operator assignment model with time-changing effects to minimise total actual flow time. *International Journal of Manufacturing Research*, 17(4), 422-451. <https://doi.org/10.1504/IJMR.2022.127091>
- Kurniawan, D., Raja, A.C., Suprayogi, S., & Halim, A.H. (2020). A flow shop batch scheduling and operator assignment model with time-changing effects of learning and forgetting to minimize total actual flow time. *Journal of Industrial Engineering and Management*, 13(3), 546-564. <https://doi.org/10.3926/jiem.3153>
- Kurniawan, D., Yusriski, R., Isnaini, M.M., Ma'ruf, A., & Halim, A.H. (2021). A Flow Shop Batch Scheduling Model with Part Deterioration and Operator Learning-Forgetting Effects to Minimise Total Actual Flow Time. *2nd Asia Pacific Conference on Industrial Engineering and Operations Management*. Surakarta, Indonesia.
- Mardi, M., Syarif, M.I., & Syakir, S. (2021). Characteristics and Praxis of Tumang Metalcraft Cultural Production. *Catharsis*, 10(2), 106-115.
- Mekler, V.A. (1993). Setup cost reduction in the dynamic lot-size model. *Journal of operations management*, 11(1), 35-43. [https://doi.org/10.1016/0272-6963\(93\)90031-J](https://doi.org/10.1016/0272-6963(93)90031-J)
- Mor, B., & Mosheiov, G. (2021). A note: flowshop scheduling with linear deterioration and job-rejection. *4OR*, 19(1), 103-111. <https://doi.org/10.1007/s10288-020-00436-z>
- Mortezaei, N., & Zulkifli, N. (2013). Integration of lot sizing and flow shop scheduling with lot streaming. *Journal of Applied Mathematics*, 2013. <https://doi.org/10.1155/2013/216595>
- Nembhard, D.A., & Uzumeri, M.V. (2000). Experiential learning and forgetting for manual and cognitive tasks. *International Journal of Industrial Ergonomics*, 25(4), 315-326. [https://doi.org/10.1016/S0169-8141\(99\)00021-9](https://doi.org/10.1016/S0169-8141(99)00021-9)
- Nikolaeva, E., & Vlasov, D. (2017). Effect of heat treatment conditions on structure and properties of high-speed steel. *IOP Conference Series: Materials Science and Engineering*. <https://doi.org/10.1088/1757-899X/177/1/012113>
- Panov, D., Naumov, S., Stepanov, N., Sokolovsky, V., Volokitina, E., Kashayev, N. et al. (2022). Effect of pre-heating and post-weld heat treatment on structure and mechanical properties of laser beam-welded Ti2AlNb-based joints. *Intermetallics*, 143, 107466. <https://doi.org/10.1016/j.intermet.2022.107466>

- Rahiminejad, D., & Compston, P. (2021). The effect of pre-heat temperature on the formability of a glass-fibre/polypropylene and steel-based fibre-metal laminate. *International Journal of Material Forming*, 14, 715-727. <https://doi.org/10.1007/s12289-020-01566-9>
- Raw, P.M. (2000). Development of a powder metallurgical technique for the mass production of carat gold wedding rings. *Gold Bulletin*, 33(3), 79-88. <https://doi.org/10.1007/BF03215482>
- Sebrina, Diawati, L., & Cakravastia, A. (2011). Learning curves in automobile assembly line. *International Conference on Industrial Engineering*. Stuttgart, Germany.
- Singh, S., Kumar, V., & Ghose, P. (2024). Computational Modeling of Walking Beam Type Reheat Furnace for the Prediction of Slab Heating and Scale Formation. *ASME Journal of Heat and Mass Transfer*, 146(1). <https://doi.org/10.1115/1.4063643>
- Strusevich, V.A., & Rustogi, K. (2017). Scheduling with time-changing effects and rate-modifying activities. Springer. <https://doi.org/10.1007/978-3-319-39574-6>
- Sukoyo, S., Samadhi, T.A., Iskandar, B.P., & Halim, A.H. (2010). Model penjadwalan batch multi item dengan dependent processing time. *Jurnal Teknik Industri*, 12(2), 69-80. <https://doi.org/10.9744/jti.12.2.69-80>
- Wang, L., Li, H., Huang, Y., Wang, K., & Zhou, M. (2022). Effect of preheating on martensitic transformation in the laser beam welded AH36 steel joint: a numerical study. *Metals*, 12(1), 127. <https://doi.org/10.3390/met12010127>
- Wright, T.P. (1936). Factors affecting the cost of airplanes. *Journal of the Aeronautical Sciences*, 3(4), 122-128. <https://doi.org/10.2514/8.155>
- Yusriski, R., Sukoyo, S., Samadhi, T.A., & Halim, A.H. (2015). 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*, 6(3), 365. <https://doi.org/10.5267/j.ijiec.2015.2.005>
- Zhu, S., Katti, I., Qiu, D., Forsmark, J.H., & Easton, M.A. (2023). Microstructural analysis of the influences of platform preheating and post-build heat treatment on mechanical properties of laser powder bed fusion manufactured AlSi10Mg alloy. *Materials Science and Engineering: A*, 882, 145486. <https://doi.org/10.1016/j.msea.2023.145486>

Journal of Industrial Engineering and Management, 2024 (www.jiem.org)



Article's contents are provided on an Attribution-Non Commercial 4.0 Creative commons International License. Readers are allowed to copy, distribute and communicate article's contents, provided the author's and Journal of Industrial Engineering and Management's names are included. It must not be used for commercial purposes. To see the complete license contents, please visit <https://creativecommons.org/licenses/by-nc/4.0/>.