Abstract
The diversification of customer needs and the shortening of product life cycle have increased the significance of small batch production. The growth of interest in environmental issues has generated a focus on methods such as recycling and reuse. Therefore, companies must pay attention to the environmental impact of their production cycle, discharge their social responsibility, and maintain their competitive advantage. In this research, we addressed the mixed-model assembly line, which is used to assemble or remanufacture multiple items simultaneously. Despite their versatility, these assembly lines carry a high risk of line stoppages, especially in dynamic environments. We describe the task assignment problem and item process sequencing problem of a mixed-model assembly line with multiple stations. A heuristic method for solving each of these problems was investigated.

Keywords: Production scheduling, Task assignment, Mixed-model assembly lines, Rebalancing, Heuristic method

1. Introduction

A mixed-model assembly line produces many types of products simultaneously on a single line divided into multiple assembly stations. The processing time at each station is determined by the nature of the item's element task. As the time required for processing each element task is different, the processing time at the station also differs. When continuously producing products whose processing time is longer than the cycle time, a line may be stopped due to work overload. However, stopping the line can be avoided by combining a product whose processing time is longer than the cycle time with one whose processing time is shorter. Therefore, to minimize the line stopping time, the item processing sequence of the products must be determined (Kotani and Ohno, 2004; Kotani and Suzuki, 2007; Kotani, 2007; Thomopoulos, 1967; Thomopoulos, 2013).

In studies of item process sequencing on a mixed-model assembly line, the objective function can be roughly divided into two types. The first aims at minimizing line stoppage time when processing cannot be completed within a station (Boysen et al., 2009; Shimizu et al., 2009; Xiao and Ohno, 1997; Yoo et al., 2004), whereas the second aims at minimizing the additional resources (i.e., total unfinished work time) needed to avoid stopping the line (Bard et al., 1992; Bautista and Cano, 2008; Bautista and Pereira, 2009; Bautista and Cano, 2011; Ishigaki and Miyashita, 2016; Leu, 1997, Scholl et al., 1998; Yano and Rachamadugu, 1991).

If the line is handling several products whose processing time exceeds the cycle time, line stoppage cannot be avoided simply by changing the item processing sequence. To prevent line stoppage on a mixed-model assembly line, both the item processing sequence and task assignment to the stations must be considered. Thomopoulos (1970) used the deviation from the average processing time for each item as a substitute evaluation value, and identified those task assignments that minimized the sum of the deviations. This method reduces the deviation in the processing time of the same item, suppresses interference between stations, and shortens the line stoppage time. However, as the evaluation is derived from the total deviations at all the stations, large divergences of individual processing time are generated at...
some stations. Link and Vaterrodt (1993) identified task assignments that minimized the maximum deviation for each item using the same substitute evaluation value as Thomopoulos (1970). Domschke et al. (1996) identified task assignments that minimized the total processing time exceeding the cycle time. However, as the processing time within the same station was combined to avoid line stoppage, since it evaluated only time to exceed cycle time, the processing time for each station became imbalanced.

In this study, we proposed a heuristic method for deciding task assignments and item processing sequences to minimize line stoppage time in a mixed-model assembly line. We also introduced a rebalancing method under dynamic constraints using these heuristic methods.

2. Mixed-item Assembly Line

Let us assume an assembly line which processes \( N \) products of \( I \) items at \( K \) stations. Figure 1 depicts a conceptual diagram of the mixed-item assembly line used in this study.

![Conceptual diagram of a mixed-model assembly line](image)

Our study made the same assumptions as those by Bautista and Cano (2008). They are as follows:
- All the work to be performed on the assembly line has previously been balanced for \( K \) assembly line stations.
- The cycle time \( c_t \) (time between the launch of two consecutive products on the line) is smaller than the greatest processing time, and it is predetermined.
- Products move through the line on a paced belt.
- The processing time depends on the item and station.
- Processing time is deterministic. Setup time is included in the processing time.
- Upstream and downstream station limits are closed, and workers cannot work beyond a station.
- When work is not completed at a station, it is accelerated using additional resources.
- The time required by operators to move upstream is negligible.

![Visualization of unfinished work](image)
Figure 2 depicts the line stoppage time (unfinished work) for a sequence “ABC,” in which A and B are items whose assembly time is longer than cycle time $c_t$. They are followed by item C, which has an assembly time shorter than $c_t$.

The cycle time is depicted on the vertical axis and the position of the item on the horizontal axis. The thick line in the figure indicates that the product is being assembled.

It is assumed that a one-dimensional vector $X$ denotes the sequence of the item, and the item number, and factor $x_n$ ($n = 1, 2, \ldots, N$) denotes the item number which processes to the $n$th is stored in the $n$th of $X$. The sequence $X$ is then expressed as follows:

$$X = [x_1, x_2, \ldots, x_N]$$ (1)

For example, for production in the item sequence “ABC,” $X$ is depicted as [123] using item numbers.

The resulting line stoppage time when $x_n$ is produced at station $k$ ($k = 1, 2, \ldots, K$) is set to $w_o (x_n, k)$. This may be expressed mathematically as follows:

$$f_k(X) = \sum_{n=1}^{N} w_o(x_n, k)$$ (2)

The total line stoppage time $Z(X)$ is given as follows:

$$Z(X) = \sum_{k=1}^{K} f_k(X)$$ (3)

The line stoppage time depends on the starting position of the operation and the processing time of an item. Let $p_{i,k}$ denote the processing time for item $i$ ($i = 1, 2, \ldots, I$) at station $k$ ($k = 1, 2, \ldots, K$), $L_k$ denote the length of station $k$ ($c_t < L_k$), $s_{n,k}$ denote the starting time measured from the upstream station limit of the $n$th ($n = 1, 2, \ldots, N$) product in the sequence in station $k$, and the cycle time assumed to be the unit time ($c_t = 1$). Under these conditions, $s_{n,k}$ is divided into two cases that may be expressed as follows:

Case I: $s_{n-1,k} + p_{x_{n-1},k} < c_t$,

$$s_{n,k} = 0$$ (4)

Case II: $s_{n-1,k} + p_{x_{n-1},k} \geq c_t$,

$$s_{n,k} = \min(L_k - c_t, s_{n-1,k} + p_{x_{n-1},k} - c_t)$$ (5)

where $s_{0,k}$ and $p_{0,k}$ ($x_0 = 0$) are set to zero. The line stoppage time $w_o (x_n, k)$ that occurs when $s_{n,k} + p_{x_n,k}$ exceeds $L_k$ is given as follows:

Case I: $s_{n,k} + p_{x_n,k} \leq L_k$,

$$w_o(x_n, k) = 0$$ (6)

Case II: $s_{n,k} + p_{x_n,k} > L_k$,

$$w_o(x_n, k) = s_{n,k} + p_{x_n,k} - L_k$$ (7)

3. Task Assignment Using a Substitute Evaluation Value

When using task assignment to balance the lines, the maximum processing time is generally minimized. However, depending on the item processing sequence when the processing time of each station in a balanced state is used, it may not be possible to avoid line stoppages. In this section, we discuss the influence of the substitute evaluation functions used in previous studies on line stoppage time.

Thomopoulos (1970) proposed the evaluation function for determining for task assignment, in order to avoid the line stop of a mixed-model assembly line. In this paper, Thomopoulos (1970) proposed a task assignment method by which the total deviation of each station is minimized, based on the average processing time for each item. The
evaluation function used is as follows:

$$\sum_{k=1}^{K} \sum_{i=1}^{I} |\bar{p}_i - p_{i,k}|$$  \hspace{1cm} \text{(8)}$$

where $\bar{p}_i$ is the average processing time for item $i$. Minimizing of the total of the deviation for each item denoted by Eq. (8) becomes difficult to generate a long processing time. Thomopoulos assumed that the line stoppage time decreased as the optimal item sequence for each station was similar. In other words, the processing time for each item need to have a very similar structure. In other words, in order for an item sequence to be similar, the processing times for every item need to have a very similar structure. However, as the evaluation function uses the total deviation in processing time, task assignments with large deviations may arise.

Link and Vaterrodt (1993) minimized the maximum value of the deviation in processing time for each item. Their evaluation function is as follows:

$$\max_{k=1..K; i=1..I} (|\bar{p}_i - p_{i,k}|)$$  \hspace{1cm} \text{(9)}$$

By solving a minimization problem using this evaluation function, the maximum value of the deviation in processing time can be made reduction. However, despite balancing the processing time of each station, line stoppages will occur if the processing time of multiple products exceeds the cycle time. In other words, a line stop occurs cause the processing time which exceeded the cycle time.

Domschke et al. (1996) only considered processing time that exceeded the cycle time, and minimized their total. Their evaluation function is as follows:

$$\sum_{k=1}^{K} \sum_{i=1}^{I} \max(p_{i,k} - c_i, 0)$$  \hspace{1cm} \text{(10)}$$

By solving a minimization problem using this evaluation function, it becomes difficult to generate a processing time which greatly exceeded cycle time.

Numerical experiments were conducted to compare the performance of these evaluation functions. The experiments assumed an assembly line with four stations, a cycle time of $c_t = 11$, and a standard station length of $L_k = 13$. Three items (A, B, and C) were assigned to the line, and the line stoppage time was calculated by enumerating the item sequence when the demand produced 19 mix ratios. Table 1 presents the 19 mix ratios for the three items when producing nine products. Following the numerical example given by Bautista and Cano (2008), the mix ratio was created for four blocks (Block 1: the mix ratio of one item is large; Block 2: the mix ratio of two items is large; Block 3: the mix ratio is comparatively well-balanced; Block 4: the mix ratio is imbalanced).

<table>
<thead>
<tr>
<th>Block 1</th>
<th>Block 2</th>
<th>Block 3</th>
<th>Block 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A 1 1 1 1 4 4 2 2 3 3 4 4 3 1 1 3 3 5 5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B 1 7 1 1 4 4 3 2 2 4 4 2 3 3 3 5 1 5 1 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C 7 1 1 4 4 1 4 3 4 2 2 3 2 3 5 3 5 1 3 1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The task time and task precedence constraints for the element task for each item were generated using the method of Kao and Queyranne (1982) and Carraway (1989), and numerical simulations were performed using 100 examples. To evaluate the performance of these evaluation functions, numerical experiments were carried out in the following procedure:

Step 1: The task time for the element task of each item was randomly decided.
Step 2: The task assignment, which minimizes the evaluation function of Eqs (8)–(10), was obtained through the
enumeration method using the task time generated.

Step 3: By using this task assignment as the processing time for items, the line stoppage time was calculated by enumerating the item sequence.

In Step 1, the task time of the element task was determined at random using uniform random numbers. In Step 2, when element tasks were assigned to the stations, the assignment whose value of an evaluation function becomes the minimum was chosen. Finally, the sequencing problem of the items was solved for assignment of the selected element tasks. In this research, all the task assignment is considered in Step 2. Therefore, it is not dependent on an evaluation function and the computation time to Step 2 is every the same. The results show that Step 2 was calculated in all 100 trials in an average CPU time of 600 sec. on a PC machine with Intel® Core™ i5-2500 processor, 3.3GHz. The program is coded in Microsoft VBA language. Then, the sequencing problem was solved at 2.03 sec. per one task assignment. In Step 3, computation time changes according to the number of task assignment generated.

Table 2 presents the ratio of divergence $\sigma (%)$ between each solution and the optimal solution ($X^*$). The ratio was derived as follows:

$$
\sigma = \frac{(Z(X) - Z(X^*))}{Z(X^*)} \times 100
$$

(11)

The 1st row of Table 2 (Thomopoulos, Link, and Domschke) shows that the value in the table was calculated using Eqs (8)–(10), respectively. When assigning tasks using an evaluation function, many task assignments undergo the same evaluation. Table 2 presents the rate of deviation of the minimum and the average of the line stoppage time when the item processing sequence is determined using these task assignments. It also presents the number of task assignments for which each evaluation value becomes the smallest by using each evaluation function.

Table 2  The ratio of divergence $\sigma (%)$ between derived solution and optimal solution, and the number of optimal solutions for three items

<table>
<thead>
<tr>
<th></th>
<th>$\sigma$ of the minimum solution</th>
<th>$\sigma$ of the average solution</th>
<th>Number of task assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thomopoulos</td>
<td>46.94</td>
<td>79.90</td>
<td>689</td>
</tr>
<tr>
<td>Link</td>
<td>26.30</td>
<td>167.55</td>
<td>4,088</td>
</tr>
<tr>
<td>Domschke</td>
<td>35.32</td>
<td>183.08</td>
<td>8,318</td>
</tr>
</tbody>
</table>

These evaluation functions focus on the balance of processing time for each station and processing time that exceeds the cycle time, which are the causes of line stoppages. As the element task time is different for each item, it is difficult to balance the processing time for all the items. However, line stoppages can be avoided by combining items whose processing time is longer than the cycle time and those whose processing time is shorter, by adjusting the processing sequence of items whose processing time is imbalanced. This implies that balancing processing time between stations is not always effective.

4. Task Assignment Considering Item Sequence

In this section, we introduce a new task assignment that considers the processing sequence of items. We consider allocations in which the total processing time for two items is no more than twice the cycle time at the same station. Our method allows avoiding line stoppages by mixing two imbalanced items. Our novel evaluation function is as follows:

$$
\max_{k=1..K} \left[ \sum_{i=1}^{I} \sum_{j=i+1}^{I} \max\left(p_{i,k} + p_{j,k} - 2 \times c_i,0\right) \right]
$$

(12)

In this study, tasks are assigned to the stations in such a way that the value derived from Eq. (12) is minimized. The evaluation function which we propose aims at that the sum of two processing time does not exceed the twice of cycle time. We think that the combination of what has long processing time and
short processing becomes easy to avoid a line stop. However, we minimize the maximum of the value exceeding cycle time, since it is difficult for all the processing time to become below cycle time.

To investigate the performance of the proposed evaluation function, we applied the numerical experiments introduced in Section 3. Table 3 presents the ratio of divergence $\sigma$ (%) between the derived solution and the optimal solution, and the number of task assignments. The proposed evaluation function was indicated to reduce the line stoppage time. The number of task assignments receiving the same evaluation also decreased, as compared to the methods of Link and Vaterrodt (1993) and Domschke et al. (1996). With the decrease in the number of task assignments, the solutions became more stable.

To investigate the influence on the difference in mix ratios, the result for each block was obtained. Figures 3 and 4 depict the minimum divergence and the average divergence from the optimal solution for each block. The proposed evaluation function reduced the minimum and average values of divergence in all blocks. As line stoppages are easy to avoid even when the processing time is imbalanced, the proposed method was effective in reducing the line stoppage time in Block 3.

Tables 4 and 5 present the result of increasing the number of items to four. Table 4 presents the 21 mix ratios of four items for producing 16 products. As an increase in the number of items significantly increases the number of item processing sequences, the metaheuristic method of Ishigaki and Miyashita (2016) was used to determine the item processing sequence. Table 5 indicates that the task assignment number was larger when using the proposed method than when using the method of Domschke et al. (1996). However, the average and minimum values of the ratio of divergence from the optimum were smaller, because the increase in the number of combinations of items improved their robustness against the differences in the mix ratio.

Table 3 The ratio of divergence $\sigma$ (%) between new solution and optimal solution, and the number of optimal solutions for three items

<table>
<thead>
<tr>
<th></th>
<th>$\sigma$ of the minimum solution</th>
<th>$\sigma$ of the average solution</th>
<th>number of task assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>0.55</td>
<td>43.35</td>
<td>2,826</td>
</tr>
</tbody>
</table>

Fig. 3 Minimum divergence from optimal solution for each block
5. Mixed-model Assembly Line Re-balancing

Yang et al. (2013) investigated the relationship between the degree of demand fluctuation and the cost of task assignment, in response to changes in the mix ratio of products. In their model, rebalancing was performed when a line stop occurred, but the timing and degree of rebalancing were not taken into consideration. As the rebalancing method, their search algorithm used the exchange of element tasks for the entire production line (Fig. 5). However, this meant that many irrelevant operations were included. In this research, we investigated a method for determining the efficient timing and degree of rebalancing.

In the method of Yang et al. (2013) a minimum reassignment is performed when a line stop occurs. However, such minimal assignment changes do not always behave robustly in a dynamic environment. Small changes in task assignment increase the number of rebalancings and the line stop time needed to reconstruct the line.

In this study, we focused on the number of rebalancings within the planning period and determined the appropriate timing from the trade-off between the degree of rebalancing and the number of times of rebalancing. In this experiment, not all tasks were exchanged, but only those on stations affecting the line stop considering the item sequence. At these stations, the element tasks were exchanged so that a combination of processing times was generated by item sequencing such that line stops were avoided.
In this experiment, we first investigated the relationship between the number of rebalancings and line stop time. Figure 6 shows the average line stop time and the number of rebalancing times when rebalancing was performed at all possible timing in the initial 5 plan periods. In this numerical example, it has been found that 2 rebalancing during the 5 planning period reduces the line stop time. In addition, as for the timing of rebalancing, the initial 2 periods were the most reduced line stop time. However, this result is limited only when the demand is constantly changing. If the planning period is long, we will need a criterion to determine the change in demand.

6. Conclusions

In this study, we proposed a method of task assignment that shortened line stoppage time, using an item processing sequence for a mixed-model assembly line. In previous approaches, when tasks were assigned to stations, either a method of balancing the processing time between stations or a method of reducing processing time that exceeded the cycle time were used. However, it is difficult to achieve these goals using the difference in processing time of different items on a mixed-model assembly line. In this study, task assignment was designed to minimize the total processing time of two items that were greater than twice the cycle time. This made it easier to combine products with a long
processing time and products with a short processing time, while robustly avoiding line stoppages caused by differences in mix rates.

We also addressed the rebalancing problem in a mixed-model assembly line. The timing and method of rebalancing considered the knowledge obtained in the determination of the item processing sequence. Using numerical experiments with a benchmark, we confirmed that the timing and method of rebalancing was robust against line stoppages.

Mixed-model assembly line and rebalancing are indispensable elements of flexible production under a dynamic environment. In further studies, the number of experiments should be increased. In this study, we designed an evaluation function for robust task assignment to fluctuations and an indicator for determining the timing of appropriate rebalancing. It is necessary to develop an algorithm to optimize these objective functions, in future studies.

Acknowledgment
The authors would like to thank Mr. Tomoyuki Miyashita, Mr. Yuta Hamada, and Mr. Kentarou Hara for providing ample information for this research. This research was partially supported by the Japan Society for the Promotion of Science (JSPS), KAKENHI, Grant-in-Aid for Scientific Research (C): JP16K01262.

References


Shimizu, Y., Waki, T. and Yoo, J. K., Multi-objective analysis for a sequencing planning of mixed-model assembly line,


Thomopoulos, N. T., Assembly Line Planning and Control (2013), Springer.


