Development of search method for sequencing problem in mixed-model assembly lines

Aya ISHIGAKI* and Tomoyuki MIYASHITA*
*Tokyo University of Science
2641, Yamazaki, Noda, Chiba 278-8510, Japan
E-mail: ishigaki@rs.noda.tus.ac.jp

Received 8 March 2016

Abstract
In the sequencing problem in mixed-model assembly lines, the branch-and-bound method and heuristic method have thus far been developed for minimizing total incomplete working hours. However, these are developed based on the heuristic solution method of a single-station model. Therefore, the precision of the solution considerably changes with differences in processing time for each station. In this study, an effective search method for the sequencing problem in mixed-model assembly lines is proposed. In our method, an efficient search is performed by improving the updating conditions of the solution using the simulated annealing method, and a high precision is achieved without depending on numerical examples.

Key words: Production scheduling, Sequencing problem, Mixed-model assembly lines, Branch-and-bound method, Simulated annealing

1. Introduction

This study considers an assembly line in which multiple items that closely resemble each other are present. The line is divided into stations, and goods are simultaneously assembled by workers assigned to each station. It is assumed that a worker cannot move with a product to perform work outside the boundary of an individual station. Work is assumed to be equally assigned among all stations. However, work differs depending on the item and station. If products that consume a long assembly time are consecutively processed by the same station, processing may not be completed within individual stations. Therefore, an important aspect that should be considered is the sequencing of product introduction (Kotani, 2007; Kotani and Ohno, 2004; Kotani and Suzuki, 2007; Thomopoulos, 1967).

In studies on sequencing in mixed-model assembly lines, the objective function, roughly divided into two types, has been used. One aims at minimizing line shutdown time when processing cannot be completed within a station (Boysen et al., 2009; Shimizu et al., 2009; Xiaobo and Ohno, 1997; Yoo et al., 2004), and the other aims at minimizing additional resources, assuming the use of additional power resources (i.e., total unfinished work time) to avoid shutting down the operation of a line (Bard et al., 1992; Bautista and Cano, 2011; Bautista and Cano, 2008; Bautista and Pereira, 2009; Leu, 1997; Scholl et al., 1998; Yano and Rachamadugu, 1991).

Most approaches to minimize total unfinished work time have used the branch-and-bound method or heuristic method. Yano and Rachamadugu (1991) determined the longest filing off that would not produce unfinished work time in the single-station model and proposed a heuristic method that utilizes the longest filing. Bautista and Cano (2008) proposed the up–down (Ud) heuristic method for the single-station model and introduced a method for constructing a solution using the results of the Ud method for each station as approximate values for a mixed-item assembly line comprising multiple stations. Although the heuristic method used by Bautista and Cano (2008) was capable of providing a precise solution, it was not very effective when applied to the lower bound of the branch-and-bound method, requiring a large number of solutions to be enumerated as numerical examples.

This study proposes an effective search method to minimize total unfinished work time when solving problems involving the determination of product introduction sequencing in mixed-item assembly lines comprising multiple
stations. An effective method for searching within a limited area is proposed and executed using simulated annealing (SA).

2. Mixed-item assembly line

An assembly line in which \( N \) units of \( I \) items are processed by \( K \) stations is assumed. Figure 1 shows the conceptual diagram of a mixed-item assembly line used in this study.

2.1 Configuration and assumptions

Our study assumes a mixed-item assembly line based on the same assumptions as those used by Bautista and Cano (2008).

[Assumptions]
- All work to be performed on the assembly line has previously been balanced on \( K \) assembly line stations.
- Cycle time, \( c_t \) (time between the launching of two consecutive products in the line), is smaller than the greatest processing time and is predetermined.
- Products move on a paced belt through the line.
- Processing time differs depending on the item and station.
- Processing times are assumed to be deterministic. Setup times are assumed to be included in processing time.
- Upstream and downstream station limits are closed, and workers cannot work beyond a station.
- When work is not completed in a station, work is accelerated using an additional resource.
- Time required by operators to move upstream is considered to be negligible.

Figure 2 shows unfinished work time for sequence “ABC,” where A and B are items that consume longer time to be assembled than the cycle time, \( c_t \), followed by item C that consumes less time than \( c_t \). The vertical direction of the figure represents time, and the horizontal direction represents position. The thick line in the figure indicates that a product is still being assembled.
When products are introduced to the line at \( c_t \), item B is introduced before item A, for which work time is longer than \( c_t \), implying that a task is completed. Therefore, work on item B cannot be started from boundary \( k \), and consequently, work cannot be completed within a station. Remaining work time is unfinished work time, and unfinished work is processed using additional costs or resources. In this study, the goal is to minimize unfinished work by scheduling. For simplification, work is accelerated by adding additional resources proportional to unfinished work time to complete the work within a station without resulting in any unfinished work time. Moreover, the velocity of a paced belt and cycle time are defined as unit time, and all parameters related to time are defined based on unit time. To avoid the complexity of equations, the velocity of a paced belt is set as 1 (velocity = 1), and it is omitted in the following sections.

### 2.2 Model formalization

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

\[
X = [x_1, x_2, \ldots, x_N].
\]  

For example, when we produce in the sequence "ABC," \( X \) is shown as \([123]\) using item numbers.

Unfinished work time that results when \( x_n \) is produced in station \( k \) \((k = 1, 2, \ldots, K)\) is set to \( w_o(x_n, k) \). This may be mathematically expressed by the following equation:

\[
f_k(X) = \sum_{n=1}^{N} w_o(x_n, k). \tag{2}
\]

Moreover, total unfinished work time \( Z(X) \) is expressed as follows:

\[
Z(X) = \sum_{k=1}^{K} f_k(X). \tag{3}
\]

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

#### Case I:
\[
s_{n-1,k} + p_{x_{n-1},k} < c_t, \quad s_{n,k} = 0, \tag{4}
\]

#### Case II:
\[
s_{n-1,k} + p_{x_{n-1},k} \geq c_t, \quad s_{n,k} = \min(L_k - c_t, s_{n-1,k} + p_{x_{n-1,k}} - c_t), \tag{5}
\]

where \( s_{0,k} \) and \( p_{0,k} (x_0 = 0) \) are set to 0. Figure 3 shows an image of starting time in Case II.
Because the velocity of a belt is assumed to be unity, a work finish location in station \( k \) is calculated by adding \( p_{xn,k} \) to \( s_{n,k} \). However, each item cannot finish at the station \( k \) when the position exceeds \( L_k \). Therefore, the unfinished work time \( w_o(x_n, k) \) which occurs when \( s_{n,k} + p_{xn,k} \) exceeds \( L_k \) is as follows.

\[
\text{Case I: } s_{n,k} + p_{xn,k} \leq L_k, \\
w_o(x_n, k) = 0, \tag{6}
\]

\[
\text{Case II: } s_{n,k} + p_{xn,k} > L_k, \\
w_o(x_n, k) = s_{n,k} + p_{xn,k} - L_k. \tag{7}
\]

3. Determining product introduction sequencing in a mixed-item assembly line

This section discusses methods of conducting effective searching by limiting the area wherein a current solution can be updated by the SA method. By considering the efficacy of a single-station approximation based on the heuristic method of Bautista and Cano (2008), the area wherein a current solution can be updated is limited, and a highly localized search is then conducted in this area.


Bautista and Cano (2008) proposed the Ud heuristic method when combining items requiring long and short work times for the single-station model, in which either type of item can be consecutively introduced as many times as possible. Figures 4 and 5 show the movement locus within a worker’s station using the Ud method. If the Ud method is executed, a worker’s work position moves to the upper stream of a station (Down) or to the lower stream of a station (Up).
Here, $W$ indicates an existing schedule, and $Y$ indicates a vector formed by unscheduled products. In the Ud method, the sequence for introducing items to an assembly line of the single-station model avoids unfinished work time by first introducing items that can potentially generate unfinished work time in relation to an unscheduled product group.

For multiple stations, Bautista and Cano (2008) proposed a method for building a solution using the heuristic method in the single-station model to estimate unfinished work time for each individual station. In other words, the Ud method is independently applied to each station with an assessment value for the group of unscheduled products ($Y$). An approximation value $LB$ for total unfinished work time is estimated by determining the sum of unfinished work time for each station, $LB_1$. Considering $LB_2$ as unfinished work time in an existing schedule $W$, $LB$ is derived as follows:

$$LB = LB_1 + LB_2,$$

$$LB_1 = f_1(Y_{1}^{Ud}) + f_2(Y_{2}^{Ud}) + \cdots + f_K(Y_{K}^{Ud}),$$

$$LB_2 = \sum_{k=1}^{K} f_k(W).$$

Here, $Y_k^{Ud}$ is a sequence of unscheduled products when the Ud method is applied to station $k$.

Figure 6 shows the architecture of the solution using the branch-and-bound method of Bautista and Cano (2008). First, unfinished work time is calculated for all items ($LB_2$). Next, unfinished work time for remaining units is calculated using the Ud method ($LB_1$). Bautista and Cano (2008) have proposed a method for building a solution by selecting only the node with the smallest $LB$ (node shown by shading in the figure) at various depths, whereby the sum of the number of squares ($LB_1$) and circles ($LB_2$) provides an approximate value for $LB$. In the depth 1 layer of Fig. 6, the first item is selected, and this item is processed first ($x_1 = 1$).
Test results when the method of Bautista and Cano (2008) is applied under the abovementioned conditions are shown, and issues that arise when they are applied to the search method are discussed below.

(1) Test conditions

Assuming an assembly line comprising four stations, 45 blend ratios of five blocks of four items (A, B, C, D) were studied. The study considered five work time patterns, and in total, 225 numerical examples were used. Cycle time was maintained at 100 s.

(2) Assessment scale

Because the optimal solution was known for the problem of Bautista and Cano (2008), the ratio (%) determined from the rate of divergence $\sigma$ (%) and the obtained optimal solution $X^*$ were used for the assessment scale.

$$\sigma = \frac{(Z(X) - Z(X^*))}{Z(X^*)} \times 100 \%.$$  \hspace{1cm} \text{(11)}

Table 1 shows the divergence ($\sigma$) from the optimal solution for the Ud method. Table 2 presents the percentage of obtaining the optimal solution ($O_p$) using the Ud method. Thus, it is shown that the solution method is excellent when the divergence from the optimal solution is closer to 0 and the percentage of obtaining the optimal solution is close to 100. The results presented in Table 1 show that the heuristic method of Bautista and Cano (2008) can produce a highly accurate solution. Some optimal solution can be obtained with work time patterns 3 and 4 from Table 2, an optimal solution almost cannot be obtained with patterns 1, 2, and 5.

On the other hand, although an optimal solution can be determined for all numerical examples, in comparison with the heuristic method, calculated time dramatically increases for patterns 1, 2, and 5 when the branch-and-bound method is applied.

Processing times for patterns 1 and 4 are presented in Tables 3 and 4, respectively. Bautista and Cano (2008) derived an approximation of total unfinished work time by determining the optimal introduction sequence in each station. In numerical examples wherein the introduction sequence was similar for each station (Table 4), the precision of approximate values was higher, and an optimal solution could be obtained. However, in numerical examples wherein the introduction sequence significantly varied between each station (Table 3), the precision of approximate values was relatively lower, resulting in a small percentage of cases where an optimal solution can be obtained. This is believed to be the reason why the lower bound is ineffective when the branch-and-bound method is applied to patterns 1, 2, and 5, resulting in larger enumerations than patterns 3 and 4. In this study, the development of a search method that is not dependent on the difference of the pattern was studied using numerical examples: pattern $P_1$ (patterns 3 and 4), wherein the optimal introduction sequence was similar for each station, and pattern $P_2$, wherein they significantly varied (patterns 1, 2, and 5).

Table 1  Divergence from optimal solution using Ud method [Bautista and Cano (2008)].

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>Time Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Block</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.742</td>
</tr>
<tr>
<td>2</td>
<td>0.578</td>
</tr>
<tr>
<td>3</td>
<td>0.349</td>
</tr>
<tr>
<td>4</td>
<td>0.467</td>
</tr>
<tr>
<td>5</td>
<td>0.557</td>
</tr>
<tr>
<td>average</td>
<td>0.539</td>
</tr>
</tbody>
</table>
3.2 Method of determining introduction sequence using SA

In this study, our search is primarily based on the SA method (Kirkpatrick et al., 1983; Van Laarhoven and Aarts, 1987). A solution $X'$ is generated from a current solution $X$ according to the neighborhood structure $N_b$. The probability of accepting an inferior solution $X'$ is determined by the distance (separation) between the two solutions $X$ and $X'$ and temperature. In the search process, temperature $T_j$ is gradually decreased using function $H$, $T_{j+1} = H(T_j)$, representing the physical equivalent of slow and gradual cooling to reach a global optimum energy state for matter. An SA algorithm for a minimization problem can now be summarized as follows.

\[
\begin{align*}
(1) & \quad \text{Set an initial temperature } T_0 \text{ and an initial solution } X. \text{ And define as best } = Z(X), \text{ best solution } X^* = X, \text{ and } j = 0. \\
(2) & \quad \text{Select a solution } X' \in N_b(X). \\
(3) & \quad \text{If } \frac{Z(X') - Z(X)}{T_j} \leq \log(Y), \quad \text{set } X = X' \quad (Y \text{ is a random number with uniform distribution within the interval [0,1]).} \\
(4) & \quad \text{If } Z(X') < \text{best, set } X^* = X' \text{ and alter the best current solution to be } Z(X'). \\
(5) & \quad \text{Revise (anneal) the temperature: set } T_{j+1} = H(T_j) \text{ and } j = j + 1. \\
(6) & \quad \text{Repeat steps (2)-(5) until the terminal condition is satisfied.}
\end{align*}
\]

As reported in Section 3.1, simulation was conducted using the same numerical examples as those used by Bautista and Cano (2008) to assess the performance of the search within a selected area in a conventional search space using the SA method. For SA, an initial solution is randomly generated, enabling an insertion method to be used to generate a neighborhood solution. The initial temperature, $T_{in}$ of the SA method was determined from preliminary tests for each pattern. Moreover, the effect of changing the temperature on search efficiency was investigated by following the renewal function:

\[
H(T_j) = T_j + \alpha.
\]

Preliminary experiments are performed at each setting of $\alpha = -0.05, 0.00, \text{ and } 0.05$. In this experiment, the constant temperature case ($\alpha = 0.00$) has the highest efficiency, followed by the increasing temperature ($\alpha = 0.05$) case, and contrary to popular belief, the decreasing temperature ($\alpha = -0.05$) case happened to be the worst. Therefore, the
percentage of obtaining an optimal solution is improved while the temperature is in the special range. If the temperature transited to a very high or very low value, the optimization routine did not function very well. Thus, we found the best initial temperature for our study and fixed the temperature throughout \((T_{j+1} = T_j)\).

The terminal condition for the SA method is a search length of 50,000 cycles. Assessment is based on an average of the tests repeated 10 times. Table 5 shows the initial temperature for the SA method, as determined by preliminary tests.

Tables 6 and 7 show the rate of divergence from the optimal solution and the respective percentage values that determined the optimal solution when the SA method was implemented. As a result of comparing the Table 2 and 6, it is shown that a solution with a higher precision than the heuristic method of Bautista and Cano (2008) is obtained using the SA method in terms of the rate of divergence from the optimal solution. Moreover, Table 2 and 7 show that the optimal solution was able to be found by SA method in many numerical examples compared with the heuristic method of Bautista and Cano (2008). Moreover, Table 7 shows that a significant difference between the patterns \(P_1\) and \(P_2\) was observed in percentage values when seeking an optimal solution up to the terminal condition. Dependency on the numerical examples was also observed for the search method. The results differed from those of the heuristic method of Bautista and Cano (2008), which was effective for pattern \(P_1\) and showed a very high precision for pattern \(P_2\).

Figure 7 shows the search process results for patterns \(P_1\) and \(P_2\) using the SA method. The horizontal axis of the figure indicates the number of iterations in the search process, and the vertical axis indicates the rate of divergence (%) from the optimal solution.

In the SA method, the solution was assessed using Eq. (12), i.e., total unfinished work time \(Z(X)\). For numerical examples with different assessment values for different stations, the solution was often updated as the search progressed because the total value for \(Z(X)\) frequently changed, as in pattern \(P_2\). In numerical examples of pattern \(P_1\), \(Z(X)\) barely changed because of the similarity in the assessment values for each station, and there was a general tendency to reach a local optimal solution. In general, when using the SA method, the stronger the local tendency for an optimal solution, the higher was the initial temperature to be set to avoid a quick convergence toward a local optimal solution. Thus, the initial temperature, \(T_0\), of the SA method in Table 5 would be set to a higher value for pattern \(P_1\) compared with other patterns. However, if the initial temperature is set very high to escape from the local optimal solution, it might take extremely long time to converge to the global optimum solution, thus making it impossible to search in an efficient manner. Therefore, limiting the area in which current solutions can be updated should be considered to improve the search efficiency.

Table 5 Initial temperature, \(T_0\), for SA method.

<table>
<thead>
<tr>
<th>Time Pattern</th>
<th>Initial Temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 6 Divergence from optimal solution of SA method.

<table>
<thead>
<tr>
<th>Block</th>
<th>Time Pattern</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.000</td>
<td>0.000</td>
<td><strong>0.003</strong></td>
<td><strong>0.003</strong></td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.000</td>
<td>0.000</td>
<td><strong>0.029</strong></td>
<td><strong>0.049</strong></td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.000</td>
<td>0.000</td>
<td><strong>0.001</strong></td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>0.000</td>
<td><strong>0.001</strong></td>
<td><strong>0.003</strong></td>
<td><strong>0.006</strong></td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>average</td>
<td>0.000</td>
<td>0.000</td>
<td>0.007</td>
<td>0.012</td>
<td>0.000</td>
</tr>
</tbody>
</table>
3.3 Improvement of search space for SA method

This study used not only total unfinished work time $Z(X)$ [Eq. (3)] but also unfinished work time $f_k(X)$ [Eq. (2)] for each station to determine the area in which a current solution can be updated. Figure 8 shows the area wherein a current solution can be updated using Eq. (12) for a two-station problem [Condition (1)]. The circles (○) in the figure indicate assessment values for the current solution of each station, and the points on the line indicate point sets that form the current solution and redundant assessment values. The shading in the figure indicates a probability that neighborhood solutions will be updated to become the “next” current solution; the darkest area has a 100% probability of being updated.

Table 7 Percentage of successful attempts seeking optimal solution using SA method.

<table>
<thead>
<tr>
<th>$O_p$</th>
<th>Time Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1  2  3  4  5</td>
</tr>
<tr>
<td></td>
<td>100 100 100 100 100</td>
</tr>
<tr>
<td></td>
<td>100 100 83 50 100</td>
</tr>
<tr>
<td>Block</td>
<td>100 100 43 29 100</td>
</tr>
<tr>
<td></td>
<td>100 100 75 100 100</td>
</tr>
<tr>
<td></td>
<td>100 96 71 71 100</td>
</tr>
<tr>
<td>average</td>
<td>100 99 74 70 100</td>
</tr>
</tbody>
</table>

Fig. 7 Search process for patterns $P_1$ and $P_2$ using SA method.

Fig. 8 Area wherein current solution can be updated using Eq. (12) [Condition (1)].
The SA method escapes from local optimal solutions by stochastically accepting worsening solutions and adjusting the worsening solution acceptance percentage by temperature. In other words, the temperature is set for numerical examples where there is a high possibility of very quickly reaching a local optimal solution. However, if the temperature is set to a very high value, the possibility of shifting to worse-case solutions increases, resulting in the deterioration of the optimal point search efficiency. Numerical examples with similar variations in assessment values for each station such as those in pattern $P_1$ tend toward local optimal solutions, and therefore, the temperature should be set to a higher value than usual. Even if assessment values for all stations do not improve, improvements can be achieved by arriving at a solution that is improved for some stations. Therefore, limiting the updating process of neighborhood solutions belonging to the first quadrant relative to a current solution was considered. Because both assessment values $f_1(X)$ and $f_2(X)$ are inferior in the first quadrant, it may be argued that the search can be relatively more efficiently conducted by limiting this area. Even if this does not improve total unfinished work time, solutions get updated when unfinished work time is improved for each individual station.

In the SA method, the search space is controlled by adding the following conditions (2) to solution update conditions (1) in Eq. (12).

\[
\frac{Z(X') - Z(X)}{T} \leq \log(Y), \quad (T \leq T'),
\]

Condition (2-2)

\[
f_k(X') \leq f_k(X).
\]

Figure 9 shows the area in which a current solution can be updated using Conditions (1) and (2). Referring to Fig. 9, the area defined by Conditions (1) and (2) that control the update area for neighborhood solutions belonging to the first quadrant can also be said to be the sum of the area wherein unfinished work time for each individual station is improved and the area normally applicable to the SA method.

A simulation test was conducted using numerical examples from Bautista and Cano (2008) reported in Section 3.1 to study the effects of differences in solution update conditions on the search efficiency. Table 8 provides the percentage of optimal solutions derived when update Conditions (1) and (2) were applied to the SA method. The proposed method was able to determine the optimal solution in various numerical examples, in comparison to the conventional SA method using update Condition (1) alone, by adding update Conditions (2) (see Table 8).

Table 9 provides the initial temperature when using update Conditions (1) and (2). It can be ascertained from Table 9 that compared with conventional SA methods, the initial temperature, $T_0$, was lower for the SA method using update Conditions (1) and (2). This suggests that the update of neighborhood solutions belonging to the first quadrant is limited. On the other hand, because the initial temperature $T_0'$ is higher than $T_0$, it can be shown that the addition of an area wherein unfinished work time is improved for each station is an effective approach to the search process.

Figure 10 shows the search process using various methods. The vertical axis of the figure indicates the rate of
divergence from the optimal solution, and the horizontal axis indicates the number of iterations in the search process. Figure 10 shows that the SA method using update Conditions (1) and (2) offers faster solution convergence and better search efficiency compared with conventional SA methods.

Table 10 provides calculation times (s) and the rates of divergence from optimal solutions when employing the branch-and-bound method, the heuristic method used by Bautista and Cano (2008), and the SA method using update Conditions (1) and (2). Calculation times and the rates of divergence from optimal solutions presented in the table are an average of 225 combinations. The table assumes that the terminal condition of the SA method is 50,000 iterations. Table 10 shows that the search method studied here yielded a larger number of optimal solutions at lower calculation time compared with the branch-and-bound method and enabled calculation without dependency on numerical examples with lower calculation time than that required for the heuristic method.

Table 8 Percentage of successful transitions seeking optimal solution using Conditions (1) and (2).

<table>
<thead>
<tr>
<th>Block</th>
<th>Time Pattern</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100 100 100 100 100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>100 100 100 100 100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>100 100 100</td>
<td>71</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>100 100 100 100 100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>100 100 100 100 100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>average</td>
<td>100 100 100 94 100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 9 Initial temperature using Conditions (1) and (2).

<table>
<thead>
<tr>
<th>Time Pattern</th>
<th>Initial Temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( T_0' )</td>
</tr>
<tr>
<td>1</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>3.0</td>
</tr>
<tr>
<td>3</td>
<td>9.0</td>
</tr>
<tr>
<td>4</td>
<td>12.0</td>
</tr>
<tr>
<td>5</td>
<td>1.0</td>
</tr>
</tbody>
</table>

![Fig. 10 Search process for Conditions (1) and (2) using SA method.](image-url)
4. Conclusions

This study developed an efficient search method that does not depend on numerical examples for mixed-item assembly lines with multiple stations. The search method enabled a solution of high precision to be determined at limited calculation time by balancing the update area for neighborhood solutions belonging to the first quadrant. In the SA method, the update area was controlled by using two temperatures and used for updating of solutions.

Our basic idea is to search a better solution using a limited solution space. In this study, although we introduced the SA method, the search algorithm is not limited to SA. In future, we will consider applications of other search algorithms. Although areas wherein unfinished work time was improved for individual stations by updating additional conditions were added to all stations, it was still possible that unnecessary areas with biased numerical examples were added to these stations. It is believed that by limiting areas and considering bias for each station, a more efficient search algorithm than this method can be realized in future.

Acknowledgments

This research was partially supported by the Japan Society for Promotion of Science (JSPS), KAKENHI, Grant-in-Aid for Scientific Research (C) 16K01262.

References

Kotani, S. and Ohno, K., Problem of minimizing the total work overload with the constraints of keeping a constant rate of usage of every part used by an assembly line, Journal of Japan Industrial Management Association, Vol.55, No.1