Integrated line balancing to attain Shojinka in a multiple straight line facility
Hadi Gökçen\textsuperscript{a}, Yakup Kara\textsuperscript{b}*, and Yakup Atasagun\textsuperscript{b}

\textsuperscript{a}Department of Industrial Engineering, Faculty of Engineering and Architecture, Gazi University, Maltepe, 06570, Ankara, Turkey; \textsuperscript{b}Department of Industrial Engineering, Faculty of Engineering and Architecture, Selçuk University, Campus, 42031, Konya, Turkey

(Received 3 January 2009; final version received 14 January 2010)

Traditional straight assembly lines are still one of the most important elements and an important fact of today’s production systems. If applicable, a company can combine its multiple straight assembly lines and obtain many advantages of Shojinka more or less. This paper analyses a new problem – integrated balancing of multiple straight assembly lines (MSLB) to attain Shojinka in a multiple straight assembly line facility. The MSLB problem is built on the concept that it could be possible for a company to obtain the advantages of Shojinka even if the company has not adopted the U-shaped line layout. Three connectivity types are suggested to integrate multiple assembly lines. A binary integer formulation for integrated balancing of multiple assembly lines is developed. The objective of the proposed formulation is to minimise the total number of workstations required in the assembly facility. The formulation is explained and validated using some illustrative examples. The proposed approach provides flexibility to minimise the total idle times of the lines and total number of workstations that are required in the assembly line facility.

Keywords: assembly line balancing; integer programming; Shojinka

1. Introduction
‘Shojinka’ is a Japanese word that is a combination of sho (to reduce), jin (worker) and ka (to change) (Sennott \textit{et al.} 2006). The concept of Shojinka, which was originally an important element of Toyota Production System (TPS), is easily to increase or decrease the number of workers in a production facility when the demand rate is increased or decreased. In a production facility, different types of products may be produced on different lines. The fluctuations in demands of products will probably require adding workers to some lines and removing from others. Attaining flexibility in the number of workers at a workshop to adapt to demand changes is called Shojinka, which is equivalent to increasing productivity by the adjusting and rescheduling of human resources (Monden 1993). Shojinka can be attained by changing the number of operations assigned to a worker, who is capable of performing multiple operations. On the other hand, the machinery layout must be proper to enable workers easily to walk between machines. The TPS adopts U-shaped machinery layout to form production lines. The U-shaped layout has several advantages over other layout types such as bird cages, isolated islands and linear (straight) layouts (Monden 1993). The visibility and communication between workers on U-shaped production lines are strengthened. In addition, the number of workers required on a U-shaped line will be less than or equal to the number of workers required on a comparable straight line (Miltenburg and Wijngaard 1994). For a given cycle time, the number of workers required on a U-shaped line may be fractional such as 3.4 workers. Four workers must be assigned to this U-shaped line to meet the demand. Hence, 60% of available time for one of the workers will be idle (waste) on this line. That is, \((4.0-3.4)/4.0 = 15\%\) of total available workforce is unproductive. In order to make this idle time productive, several U-shaped lines are combined into one integrated line by locating them close to each other. This way, a worker can perform operations from two or more neighbour U-shaped lines, and idle times can be eliminated or reduced. Such a production facility that consists of a group of U-shaped lines is defined as a Just-in-Time (JIT) production unit (Sparling 1998).

The problem of assigning operations to workstations on U-shaped lines, the single U-line balancing (SULB), was first studied by Miltenburg and Wijngaard (1994). Essentially, the SULB problem is the U-shaped line version of a well-known problem, the simple assembly line balancing (SALB), which was introduced by Salveson (1955). Although numerous research papers on SALB have been published, the

*Corresponding author. Email: ykara@selcuk.edu.tr
The literature on balancing straight assembly lines includes examples of combining multiple straight assembly lines. Gökçen et al. (2006) suggested that in a production facility, two or more straight assembly lines can be located in parallel and they can be balanced simultaneously. They developed a binary formulation and proposed a heuristic procedure for balancing of parallel straight assembly lines with the objective of minimising the number of workstations required in the system. Balancing parallel assembly lines simultaneously will constitute workstations containing tasks on two adjacent and parallel lines and these common (multi-line) workstations will provide the flexibility to minimise the total idle times of the lines and total number of workstations required in the production facility. Gökçen et al.’s (2006) study assumes that multiple assembly lines are located in parallel and parallel connections can be constructed between two lines. Essentially, Gökçen et al.’s (2006) study can be considered as a partial implementation of Shojinka in a parallel assembly line facility. Gökçen et al. (2006) provide a framework that combines two parallel assembly lines with common workstations. However, in practice, two or more assembly lines can be combined with common workstations using several connectivity opportunities. But, there is no study that combines straight assembly lines to attain Shojinka in a multiple straight line facility. This is the first study that proposes an assembly line balancing procedure to attain the benefits of Shojinka in a multiple straight line facility. This paper analyses a new problem, which is the integrated balancing of multiple straight assembly lines (MSLB). The objective of the problem is to assign tasks to a minimum number of workstations on multiple straight assembly lines. Possible location forms and connectivity opportunities among straight assembly lines are investigated and given in Section 2. Based on the connectivity types detected, a binary formulation for MSLB is presented in Section 3. The objective of the proposed formulation is to minimise the number of workstations required in a multiple straight assembly line facility. The proposed formulation is illustrated using examples in Section 4. Some concluding remarks and opportunities for further research are presented in Section 5.

2. Combining multiple straight assembly lines

Two assembly lines can be combined by connecting them with one or more common workstations. Thus, operators can work in two or more different assembly lines at the same time. These connections may provide an opportunity to reduce the total labour requirement of the system. In this section, we propose that multiple straight assembly lines can be combined using several
connectivity opportunities between these lines. Gökçen et al. (2006) suggested that two or more straight assembly lines can be located in parallel to each other and they can be balanced simultaneously. Balancing parallel assembly lines may result in common workstations that include tasks from two adjacent lines. These common workstations will provide flexibility to minimise the total idle times of lines and total number of workstations required in the production system. Gökçen et al.’s (2006) approach is based on the concept of parallel connectivity of two adjacent assembly lines located in parallel. Figure 1 shows two simultaneously balanced parallel assembly lines with parallel connectivity.

It can be shown in Figure 1 that the parallel connectivity appears in two common workstations. The first common workstation includes task 3 of line 1 and tasks 2 and 3 of line 2. The second common workstation includes tasks 4 and 6 of line 1 and task 5 of line 2. Parallel connectivity can be benefited when two or more assembly lines are located in parallel and they are sufficiently close to each other. However, it should be noted here that a common workstation includes tasks from only two adjacent lines. This is due to the difficulty and inefficiency for operators to travel from one line to another non-adjacent line.

In a production system, it may not be always possible to locate all assembly lines in parallel. Furthermore, if assembly lines are tightly related to each other and some supplier–customer relationships exist between these assembly lines then they may be located in different ways. In addition to the parallel connectivity, we suggest two new connectivity types, which are consecutive and perpendicular connectivity. It should be noted here that one or more common workstations are required to obtain a parallel connectivity between two assembly lines. However, only one common workstation can be utilised to obtain consecutive or perpendicular connectivity between two assembly lines.

The consecutive connectivity can be benefited when two assembly lines are located consecutively. This connectivity type can appear between an upstream line (supplier) and its downstream line (customer). If the output of an upstream line is the main input (part) of a downstream line, then assembly line managers may desire to locate these lines consecutively and close to each other. This means that the output of the upstream line is processed throughout most of the tasks on the downstream line. In this case, we will have the chance to obtain a consecutive connectivity between these lines. Figure 2 shows two simultaneously balanced consecutive assembly lines.

As shown in Figure 2, two consecutive assembly lines are connected with a common workstation that includes tasks 4 and 5 of line 1, and task 1 of line 2. A consecutive connection must be constructed at the end of the upstream line and the beginning of the downstream line.

The outputs of an upstream line may not always be the main part for a downstream line. In other words, the outputs of the upstream line may not be processed throughout most of the tasks on the downstream line. The outputs may be components that are attached to the main parts processed on the downstream line. In this case, assembly line managers probably desire to locate the upstream line to the nearest point of use so as to minimise material handling from the upstream line to the downstream line. If such a location exists in a production system, two lines can be connected to each other with perpendicular connectivity. Figure 3 shows two assembly lines that have a perpendicular connectivity with a common workstation.

Figure 3 shows that two lines are connected with a common workstation which includes task 3 of line 1 and task 5 of line 2. The outputs of the upstream line are required at the point of task 5 of the downstream line. That is, task 5 cannot be completed unless the outputs of line 1 feed it.

Based on the locations, an assembly line of a production system can be connected to another
assembly line with parallel, consecutive or perpendicular connectivity. Two assembly lines can be connected to each other with only one of these connectivity types at the same time. However, an assembly line can be connected to several assembly lines with different connectivity types at the same time. Such connectivity cases can be called a mixed connectivity. Figure 4 shows a mixed connectivity of four assembly lines.

Figure 4 shows that line 1 and line 2 are connected to each other with a parallel connectivity; line 2 and line 3 are connected to each other with a perpendicular connectivity; line 2 and line 4 are connected to each other with a consecutive connectivity. As shown in Figure 4, line 2 is connected to three lines with all types of connectivity at the same time. In addition, line 1 is not allowed to connect to line 3 and line 4.

Balancing several assembly lines simultaneously using their connectivity opportunities will provide flexibility to minimise the total idle times of the lines and total number of workstations required in the production system. However, there is an important problem when connecting multiple assembly lines with common workstations. If the cycle times of assembly lines vary, the cycle time for a common workstation is the minimum of the cycle times of multiple assembly lines that it spans (Miltenburg 1998). For example, consider a common workstation that spans line a, where the cycle time is 5 min and that spans line b, where the cycle time is 10 min. If the workload of this workstation exceeds 5 min, the operator will be unable to complete tasks on line a in its cycle time.

3. Mathematical formulation

The most important management problem in assembly lines is the Assembly Line Balancing (ALB). The simplest version of ALB problems is the SALB problem mentioned above. ALB is the problem of assigning assembly tasks to successive workstations by satisfying some constraints and optimising a performance measure. This performance measure is usually the minimisation of the number of workstations utilised in the assembly line. For a given cycle time, minimising the sum of station idle times is equal to minimising the number of opened stations. SALB problems under this objective are called Type-1. Conversely, if the number of stations is given, then minimising the cycle time guarantees minimum idle times, which is known as Type-2. If both number of stations and the cycle time can be altered, the line efficiency E is used to determine the quality of a
balance. The corresponding SALB problem was hence labeled Type-E (Boysen et al. 2007). It is usually assumed in ALB literature that each workstation requires one worker. Hence, if the number of workstations utilised in the line is minimised then the number of workers required is also minimised and productivity is maximised. There are three main constraints of ALB problems. The first set of constraints is called assignment constraints that ensure each task is assigned to at least and at most one workstation. Cycle time constraints are the second set of constraints that guarantee the workload of a workstation will not exceed the cycle time. Cycle time is the time interval between two successive completed products and represents the output rate of the line. In other words, an assembly line produces one product for each cycle time unit. Therefore, the workloads of workstations should not exceed cycle time. The precedence relationships among tasks are satisfied by precedence constraints. The precedence relationships among assembly tasks are illustrated using precedence diagrams. Figure 5 shows an example precedence diagram with six tasks.

In this section, we propose a binary formulation for MSLB. The proposed formulation attempts to balance an entire production system which consists of combined assembly lines. The proposed formulation is aiming at integrated balancing of multiple assembly lines to attain Shojinka in a multiple straight line facility using the connectivity types mentioned in Section 2. Therefore, we need an integrated precedence diagram that represents all precedence diagrams of multiple assembly lines. We suggest that the integrated precedence diagram of a production system can be obtained by merging individual precedence diagrams into a huge diagram. Multiple precedence diagrams can be integrated based on the connectivity opportunities between multiple assembly lines. For example, if a consecutive connectivity exists between two assembly lines, the precedence diagram of the downstream assembly line can be appended at the end of the upstream assembly line. Thus, these assembly lines can be connected with a common workstation which includes at least the last task of upstream line and the first task of downstream line. Based on the same connectivity opportunities given in Section 2, Figure 6 shows the integrated precedence diagram of multiple assembly lines shown in Figure 4.

Although the production system has a SALB problem for each line, these SALB problems are now transformed into a single MSLB problem using the integrated precedence diagram. It should be noted in Figure 6 that task numbers are sequentially modified in the integrated diagram to prevent any inconvenience. Since line 1 and line 2 are located in parallel and they can be combined with parallel connectivity opportunities, no precedence relationships between any tasks of these lines are established in the integrated diagram. If a supplier–customer relationship exists between an upstream and a downstream assembly line, a
preference relationship between two assembly lines should be established. There is no supplier–customer relationship between line 1 and line 2. Once the integrated precedence diagram is established and connectivity options between assembly lines are identified, the following formulation can be used for simultaneous balancing of multiple assembly lines:

Indices:
- \( i, r, s, k, l \): task
- \( j \): workstation
- \( h, g, a, b, p, q \): assembly line

Parameters and sets:
- \( I_h \): set of tasks on line \( h \)
- \( J \): set of workstations; \( j = 1, \ldots, K_{max} \)
- \( A \): set of assembly lines
- \( n_h \): number of tasks on line \( h \)
- \( H \): total number of assembly lines in the production system
- \( N \): total number of tasks in the production system; \( N = \sum_{h \in A} n_h \)
- \( K_{max} \): maximum number of workstations
- \( C_b \): cycle time of line \( h \)
- \( t_i \): completion time of task \( i \)
- \( M \): a big number
- \( S \): set of all precedence relationships in the production system
- \( (r, s) \in S \): a precedence relationship; task \( r \) is an immediate predecessor of task \( s \)
- \( R \): set of disconnection relationships
- \( (p, q) \in R \): a disconnection relationship; line \( p \) cannot be connected with line \( q \)
- \( F \): set of perpendicular connectivity relationships
- \( (a, b) \in F \): a pair of lines; upstream line \( a \) can be connected to downstream line \( b \) with perpendicular connectivity
- \( k \): the last task in the precedence diagram of upstream line \( a \)
- \( l \): the task of downstream line \( b \) to which the outputs of upstream line \( a \) are input

Variables:
- \( x_{hij} \): 1, if task \( i \) of line \( h \) is assigned to workstation \( j \); 0, otherwise
- \( U_{hj} \): 1, if workstation \( j \) is utilised on line \( h \); 0, otherwise
- \( z_j \): 1, if workstation \( j \) is utilised; 0, otherwise
- \( W_{(a,b)j} \): 1, if workstation \( j \) includes tasks from both line \( a \) and line \( b \). That is, it is a common workstation; 0, otherwise
- \( V_{(a,b)j} \): 1, if workstation \( j \) includes tasks from only line \( a \) or line \( b \). That is, it is not a common workstation; 0, otherwise

We assume in our formulation that completion times of tasks are deterministic and the longest task time is not greater than cycle time. The mathematical formulation is presented below:

Minimise \[ \sum_{j=1}^{K_{max}} z_j \] (1)

Subject to:
- \[ \sum_{j=1}^{K_{max}} x_{hij} = 1 \quad \forall h \in A \quad \forall i \in I_h \] (2)
- \[ \sum_{j=1}^{K_{max}} (K_{max} - j + 1)(x_{hij} - x_{hji}) \geq 0 \quad \forall (r, s) \in S \] (3)
- \[ \sum_{i \in I_h} t_i x_{hij} = M - (M - C_p)U_{gj} \quad \forall j \in J \quad \forall g \in A \] (4)
- \[ \sum_{i \in I_h} x_{hij} - n_h U_{hj} \leq 0 \quad \forall j \in J \quad \forall h \in A \] (5)
- \[ \sum_{i \in I_h} x_{hij} - n_h U_{hj} \geq 1 - n_h \quad \forall j \in J \quad \forall h \in A \] (6)
- \[ U_{pj} + U_{qj} \leq 1 \quad \forall j \in J \quad \forall (p, q) \in R \] (7)
- \[ U_{aj} + U_{bj} - 2W_{(a,b)j} - V_{(a,b)j} = 0 \quad \forall j \in J \quad \forall (a, b) \in F \] (8)
- \[ \sum_{j=1}^{K_{max}} W_{(a,b)j} \leq 1 \quad \forall (a, b) \in F \] (9)
- \[ x_{akj} + x_{hbj} - 2W_{(a,b)j} \geq 0 \quad \forall j \in J \quad \forall (a, b) \in F \] (10)
- \[ \sum_{h=1}^{H} U_{hj} - H z_j \leq 0 \quad \forall j \in J \] (11)

The objective of the proposed formulation is to minimise the number of workstations utilised in the entire production system. Equation (2) ensures that each task is assigned to at least and at most one workstation. Equation (3) ensures that task \( s \) cannot be assigned until its predecessor task \( r \) is assigned to an earlier or the same workstation that task \( s \) is assigned.
Equation (4) is the cycle time constraints of the model. This set of constraints ensures that the work content of a workstation does not exceed cycle time. These constraints are designed for the situation that each line runs at different cycle times. Therefore, these constraints also ensure that the cycle time for a common workstation is the minimum of the cycle times of multiple assembly lines that it spans. Equations (5) and (6) determine whether workstation \( j \) is utilised on line \( h \) or not. If workstation \( j \) is utilised on line \( h \) then \( U_{jk} \) will be 1, otherwise it will be 0. Connecting two assembly lines with any connectivity type may not always be possible. Therefore, Equation (7) is used to disconnect two assembly lines that cannot be connected with one or more common workstations. Equations (8), (9) and (10) are added to the model to establish perpendicular connections. For a given \((a, b)\) perpendicular connectivity relationship, Equation (8) determines whether any common workstation \( j \) between lines \( a \) and \( b \) is utilised or not. Equation (9) ensures that at most one common workstation can be utilised between lines \( a \) and \( b \). If a common workstation between lines \( a \) and \( b \) is utilised, Equation (10) guarantees this workstation contains the last task \( k \) of the upstream line \( a \) and the task \( l \) of the downstream line \( b \) to which the outputs of upstream line \( a \) are the input. Equation (11) determines whether workstation \( j \) is utilised or not.

4. Illustrative examples

The proposed model is illustrated using a simple multiple straight assembly line facility. The example facility consists of four assembly lines with a total of 15 tasks. Task completion times and cycle times of assembly lines are given in Table 1.

Suppose assembly line managers conveyed that there are several connectivity opportunities between four assembly lines. These opportunities are given in Table 2.

The set of disconnection relationships can be easily obtained using Table 2. Using these connectivity opportunities, the integrated precedence diagram is constructed and presented in Figure 7.

At the first stage, each assembly line is balanced independently supposing no connectivity between assembly lines is allowed. The results for independent line balances will show us the effects of common workstations on total number of workstations utilised in the production facility. The maximum number of workstations \( K_{\text{max}} \) is selected as 6 and the problems are solved using LINGO (Schrage 2002) on an Intel Core 2, 2.00 GHz, 1022 MB Ram computer. Independent line balances are given in Table 3.

Table 3 shows that a total of seven workstations are required for the optimal solution. For workstation 1, \( \frac{2}{5} = 40\% \) of total available workforce is unproductive. The ratios of unproductive times for other workstations are 40%, 20%, 50%, 30%, 40% and 25% respectively with an average of 35%. This can be considered as an important waste of workforce resources. According to the assignments given in

### Table 1. Task data and cycle times of assembly lines.

<table>
<thead>
<tr>
<th>Lines (h)</th>
<th>Tasks (i)</th>
<th>Task Times (t_i)</th>
<th>Cycle Times (C_h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>11</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

### Table 2. Connectivity opportunities between assembly lines.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>None</td>
<td>Parallel</td>
<td>None</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>None</td>
<td>None</td>
<td>Perpendicular</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>None</td>
<td>Consecutive</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>None</td>
<td>None</td>
<td></td>
</tr>
</tbody>
</table>

![Figure 7. The integrated precedence diagram of the example.](image-url)
Table 3. Independent line balances.

<table>
<thead>
<tr>
<th>Workstation/Line</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>Workloads</th>
<th>Idle Times</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1,2</td>
<td></td>
<td></td>
<td>3</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td>3</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>7,8,9,10</td>
<td>12</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>5</td>
<td></td>
<td>5</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>4,6</td>
<td></td>
<td>7</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td>11,12</td>
<td>12</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td>13,14,15</td>
<td>15</td>
<td></td>
<td>5</td>
</tr>
</tbody>
</table>

Table 3, the layout of the facility can be illustrated as shown in Figure 8.

The example facility is then balanced considering the connectivity opportunities given in Table 2. Since the cycle times of assembly lines vary, the cycle time for a common workstation will be the minimum of the cycle times of multiple assembly lines that it spans. In addition to the precedence relationships for each individual precedence diagram, (6, 14) and (10, 11) precedence relationships are added to S for reflecting the integration of precedence diagrams. The results are given in Table 4.

Table 4 shows that total number of workstations is reduced to six when assembly lines are connected to each other with common workstations. Workstations 1, 3 and 5 are common for two different assembly lines. Workstation 1 is common for lines 1 and 3, workstation 3 is common for lines 3 and 4, and workstation 5 is common for lines 2 and 4. According to the assignments given in Table 4, the layout of the facility can be illustrated in Figure 9.

The cycle times for these common workstations are the minimum of the cycle times that they span. For example, the cycle time for line 2 is 10 min and the cycle time for line 4 is 20 min. However, the cycle time for workstation 5 that is common for these lines is 10 min. Owing to the timing difficulties and operator/machine interference, the multi-line (common) workstations that were utilised on these lines are more difficult to operate when the cycle times of the lines are varied (Miltenburg 1998). Consider workstation 5 in Figure 10 as an example. Task 6 must be completed once every 10 min, while task 14 must be completed once every 20 min. Therefore, a task sequence that the operator will follow must be generated to operate this workstation synchronously. Three Gantt charts for task sequences of common workstations 1, 3, and 5 are generated and given in Figure 10.

The Gantt charts show when the operator must start and finish the tasks and when the operator is idle. The task sequences in Figure 10 are generated for a limited duration. This duration is the least common multiple of the cycle times of associated assembly lines. For instance, the task sequence of workstation 3 is generated for 60 min because the least common multiple of 15 and 20 is 60. After 60 min, the operator will follow the same task sequence. Therefore, the magnitude of workloads and idle times for common workstations must be calculated by considering least
common multiple of cycle times. Figure 10 shows that 6 min of 15 min is idle for workstation 1, 6 min of 60 min is idle for workstation 3 and 9 min of 20 min is idle for workstation 5. This means that 40%, 10% and 45% of available time of workstations are unproductive respectively. The workloads of regular (single-line) workstations 2, 4 and 6 are 8, 4 and 18 min respectively with 20%, 20% and 10% unproductive idle time ratios. These values mean that the average rate of unproductive times for the example multiple assembly line facility is 24.16%. That is, unproductive times for workstations are reduced significantly compared with independent line balances.

5. Conclusions

The main objective of the TPS is to increase productivity by eliminating or reducing all kinds of wastes in a production system. Shojinka is an important TPS technique that aims to eliminate or reduce operators’ idle times. If the cycle time is sufficiently large, an operator can perform operations on two or more different production lines at the same time. The problem of assigning operations to operators is important while implementing Shojinka. In this paper, we proposed that if practical, Shojinka can be implemented in a multiple straight assembly line
facility. The problem of assigning assembly tasks of multiple assembly lines to minimum number of workstations is called MSLB. The MSLB problem is built on the concept that it should be possible for a company to obtain the advantages of Shojinka even if the company has not adopted the U-shaped line layout. Three connectivity types to combine multiple straight assembly lines are investigated and explained. A binary formulation for MSLB is developed based on the connectivity types investigated. The proposed formulation assigns some tasks to common workstations which include tasks on two or more neighbour assembly lines. The utilisation of common workstations minimises total idle times of assembly lines and total number of workstations required in the facility. As with other types of ALB problems, the MSLB problem is also NP-Hard. Therefore development of effective heuristics to solve an MSLB problem is an important topic for future researches. In addition, the proposed binary formulation is expected to help researchers to develop formulations for combining multiple U-lines.

References