Simultaneous Multi-skilled Worker Assignment and Mixed-model Two-sided Assembly Line Balancing

P. Fattahi, P. Samouei, M. Zandieh

1. INTRODUCTION

An assembly line is a production process which usually has several work centers (stations) connected together with a material handling system such as a conveyor belt. In this line, the unfinished products are launched down through the stations and a set of tasks with certain operation times and precedence relationships among tasks are done with robots or human workers in each station.

In the assembly line, worker assignment and line balancing have important effects on the performance of the line. Assembly line balancing is to assign the tasks to the stations in such a way that satisfy all constraints, and the objectives are optimized.

The first scientific article in the assembly line balancing problems (ALBP) was published in 1950s. After that, because of the important role of balancing on productivity, many researchers have studied this problem with different constraints, objectives and solving methods to make better decisions for the real-world situations.

There are several surveys on the ALBP available in the literature [1-10]. There are several classifications for ALBP. For example, in terms of the number of models in a line, this problem can be categorized into single-model, mixed-model and multi-model. In single-model, only one type of product; in mixed-model, different models of one product; and in multi-model several products in batches are assembled.

In addition, based on the properties of the products, technical or operational requirements, the layout of assembly lines can be one-sided or two-sided lines. In one-sided assembly lines, only one side of the line is used, whereas in two-sided assembly lines, both sides of the line are used for assembly tasks on the same product in parallel. For large-sized products, such as cars, two-sided assembly line is more suitable than one-sided,

**Paper history:**
Received 28 July 2015
Received in revised form 06 January 2016
Accepted 28 January 2016

**Keywords:**
Worker Assignment
Mixed-model
Skill
Particle Swarm Optimization
Simulated Annealing
Taguchi Method
because this structure has a shorter line length, lower cost of tools and fixtures and fewer material handling systems. Figure 1 shows a two-sided assembly line layout.

In a two-sided assembly line, two operators, each of which placed on the opposite side of the mated-station, work together in parallel on different tasks without interfering with one another on the same individual product [11].

There are two famous objective functions for solving TSALBP (Type-I and Type-II); Type-I is minimization of the number of mated-stations (i.e., the line length) for a given cycle time; and Type-II is minimization of the cycle time for a fixed number of mated-stations.

Since, the number of stations for the same number of mated-stations in Type-I can be different [12], the number of mated-stations as well as the number of stations can be considered in TSALBP.

Also, TSALBP can be categorized to one objective and multi-objective. Some works [13, 14] used one objective function and other studies [11, 12, 15] used more than one objective for two-sided assembly line balancing.

Similar to one-sided ALBP, the two-sided ALBP is NP-hard [16]. Therefore, metaheuristic algorithms are used for solving large-sized assembly line balancing problems in reasonable time to obtain optimal or near optimal solutions. These algorithms can be classified as Simulated Annealing [17], Genetic Algorithm [18-21], Ant Colony Optimization (ACO) ([15, 22, 23]), Tabu Search [24], Particle Swarm Optimization [11], and Bee Colony Optimization [14, 25, 26]. For example, Simaria and Vilarinho [15] presented a mathematical model and used ACO algorithm for a mixed-model TSALBP with precedence, zoning, capacity and synchronism constraints. Also, Özcan and Toklu [12] addressed the type-I of mixed-model TSALBP and presented a SA algorithm to solve it. They considered the minimization of number of mated-stations and the number of stations for a given cycle time.

Most of researchers assumed that the operation time is deterministic [27], and is not dependent on the skills of workers. However, in many real-world situations, the assembly tasks are done manually, and a high-skilled worker can assemble products faster than a low-skilled worker. So, the skills of workers can affect the balance of a line. Furthermore, distinguishing between the levels of skills permits a manager to decide which task should be done by which worker, and it causes a good saving for human cost and time. Therefore, some authors verified the assembly line balancing and worker assignment. For example, Ramezanian and Ezzatpanah [28] considered the modeling and solving of multi-objective mixed-model one-sided assembly line balancing and worker assignment problem. They minimized the total cycle time and the operating costs. They used a goal programming approach and imperialist competitive algorithm to solve it. Also, Sungur and Yavuz [29] introduced assembly line balancing with hierarchical worker assignment to minimize the total cost and formulated the problem as an integer linear programming model. Recently, Zacharia and Nearchou [30] presented a multi-objective evolutionary algorithm for the solution of the bi-criteria single-model one-sided assembly line worker assignment and balancing problem.

There are a few papers which investigated the different classes of workers’ skill in assembly line balancing. For example, Corominas et al. [31] verified ALBP with skilled and unskilled workers.

To the best of our knowledge, there is no paper addressing the worker assignment with different skills for mixed-model TSALBP. So, in this research, a mathematical model is developed for this problem when the operation times are dependent on the skills of workers to minimize the number of mated-stations, the number of stations and the total human cost for a given cycle time. Additionally, a PSO algorithm is used to solve it. The efficiency of the PSO algorithm was compared with the SA algorithm in the literature [12].

The remainder of this paper is structured as follows. The related assumptions and the mathematical model of the problem are given in Section 2. Section 3 presents the proposed PSO algorithm in detail. Section 4 provides numerical experiments. Finally, Section 5 is devoted to conclusions and recommendations for future research.

2. PROBLEM DEFINITION

Mixed-model two-sided assembly lines are often used in a range of industries that assemble large-sized products such as cars. These lines are flexible and permit to produce different models of one product.

There are a few studies for mixed-model TSALBP. This problem and worker assignment with different skill levels to make a better decision for the real-world situation is considered in this research.

In this section, the problem assumptions, the notations and the mathematical model for the mixed-model TSALBP and worker assignment with different skill levels are presented.
2.1 Problem Assumptions

The assumptions of the problem are given as follows:

1. Tasks are performed by operators in parallel at both sides of the line. It means there is a two-sided assembly line.
2. Products with similar production characteristics are assembled on the same two-sided assembly line.
3. Some tasks may be required to be performed at one-side of the line, while others may be performed at either side of the line.
4. Each task is allowed to assign to only one workstation, and each task should be done only once.
5. Precedence diagram for each model is known.
6. Workers with different skills are available (low-skill, medium-skill, high-skill) and the operation time depends on the skill of the worker.
7. The workers’ skill levels do not change during the production planning horizon.
8. Similar tasks among different models exist.
9. The travel times of the operators are zero.
10. There is no buffer in the line.
11. The cycle time is given.

2.2 Mathematical Model

With the above assumptions, the mathematical model for multi-objective mixed-model TSALBP and worker assignment with different levels of skill is presented using the following notations:

Indices:
- $i, j$: Task
- $p, r$: Skill
- $j, g$: Mated-station
- $l$: Product model
- $k, f$: Side of the line; (1: indicates a left-side station) and (2: indicates a right-side station)
- $(j, k)$: Station of mated-station $j$ and its operation direction is $k$

Parameters:
- $I$: Set of tasks in the combined precedence diagram
- $J$: Set of mated-stations
- $L$: Set of skills (low, high, ...)
- $M$: Set of product models
- $S_L$: Set of tasks which should be performed at a left-side station; $SL \subset I$
- $S_R$: Set of tasks which should be performed at a right-side station; $SR \subset I$
- $S_E$: Set of tasks which may be performed at either side of a station; $SE \subset I$
- $P(i)$: Set of immediate predecessors of task $i$
- $P_i(i)$: Set of all predecessors of task $i$
- $S_i(i)$: Set of all successors of task $i$
- $P_o$: Set of tasks that have no immediate predecessors: $P_o=\{ i|P(i)=\emptyset \}$
- $t_{ mdl }$: Operation time of task $i$ for model $m$ with skill $l$
- $HC_{ijkl}$: Human cost of worker with skill $l$ in station $(j, k)$
- $\psi$: A very large positive number
- $C(i)$: Set of tasks whose operation directions are opposite to operation direction of task $i$
  \[
  C(i) = \begin{cases} 
  S_L & \text{if } i \in S_R \\
  S_R & \text{if } i \in S_L \\
  \emptyset & \text{if } i \in S_E 
  \end{cases}
  \]
- $K(i)$: Set of indicating the preferred operation directions of task $i$
  \[
  K(i) = \begin{cases} 
  \{1\} & \text{if } i \in S_R \\
  \{2\} & \text{if } i \in S_L \\
  \{1, 2\} & \text{if } i \in S_E 
  \end{cases}
  \]
- $C$: Cycle time
- $W_{ jkm }$: Subset of all tasks that can be assigned to station $(j, k)$ of model $m$
- $||W_{ jkm }||$: Number of tasks in subset $W_{ jkm }$
- $||M||$: Number of models
- $||L||$: Number of skills

Variables:
- $x_{ijkl}$: 1, if task $i$ is assigned to station $(j, k)$ with skill level $l$; 0, otherwise
- $t_{ jml }$: Finish time of task $i$ for model $m$ with skill $l$
- $F^j$: 1, if mated-station $j$ is utilized; 0, otherwise
- $G_{ jkl }$: 1, if station $(j, k)$ is utilized by a worker with skill $l$; 0, otherwise
- $U_{ jk }$: 1, if the work content of station $(j, k)$ for all models is different from zero, then station $(j, k)$ is utilized for all models; 0, otherwise
- $v_{ jkm }$: 1, if station $(j, k)$ is utilized for model $m$; 0, otherwise.
- $z_{ip}$: 1, if task $i$ is assigned before task $p$ in the same station; 0, if task $p$ is assigned before task $i$ in the same station

In this paper, a multi-objective mathematical model for mixed-model TSALBP and worker assignment with different levels of skill based on the formulation presented in the literature [12] is proposed. This mathematical model is as follows:

\begin{align}
\text{Min } & \sum_{j \in J} F^j \\
\text{Min } & \sum_{j \in J} \sum_{i \in I} \sum_{k=1}^{2} G_{ jkl } \\
\text{Min } & \sum_{j \in J} \sum_{i \in I} HC_{ijkl} \cdot G_{ jkl } \\
\text{S.t.:} & \\
\sum_{j \in J} \sum_{k=1}^{2} x_{ijkl} = & 1 \quad \forall i \in I \\
\sum_{j \in J} \sum_{k=1}^{2} \sum_{l \in L} g_{ijkl} - & \sum_{j \in J} \sum_{k=1}^{2} \sum_{l \in L} P_o \cdot x_{ijkl} \leq 0 \quad \forall i \in I, m \in M, l \in L \\
\sum_{i \in I} t_{jml} \leq & C \quad \forall i \in I, m \in M, l \in L
\end{align}
In this section, we present a particle swarm optimization to solve the mixed-model TSALBP and worker assignment with different skills. The proposed algorithm is based on the procedure of the SA algorithm presented elsewhere [12]. However, it was changed and adopted for the considered problem.

3. 1. The Standard PSO Algorithm

One of the population-based metaheuristic algorithms is PSO that was introduced in 1995. In this algorithm, a swarm of particles searches a D-dimensional space to find the best solution. Each particle at each iteration has a velocity and fitness value, and the velocity of each particle is computed based on the best previous positions of its own and the population [27].

The parameters of this algorithm are two positive constants (c1 and c2) which are called cognitive and social coefficients; two uniform random values (r1 and r2) between 0 and 1; and the inertia weight (W).

There are different methods for the selection of these parameters. For example, Kennedy and eberhart [32] used Equation (19) for inertia weight:

\[ W = W_{\text{max}} - \frac{W_{\text{min}} - W_{\text{max}}}{\text{Itr}_{\text{max}}} \times \text{Itr} \]

where \( W_{\text{max}}, W_{\text{min}}, \text{Itr}_{\text{max}}, \text{Itr} \) are the initial inertia weight, final inertia weight, the maximum number of iterations and the number of iteration, respectively.

The basic PSO structure is as follows [27]:

**Step 1.** Generate the position \( (X_i^j) \) and the velocity \( (V_i^j) \) of each particle in the initial swarm according to the following relations:

\[ X_i^0 = X_{\text{Min}} + \text{Random}(X_{\text{Max}} - X_{\text{Min}}) \]
\[ V_i^0 = V_{\text{Min}} + \text{Random}(V_{\text{Max}} - V_{\text{Min}}) \]

**Step 2.** Compute the news positions and the new velocities of particles:

\[ X_{i,k+1} = c_1r_1(X_{i,k}^{\text{best}} - X_{i,k}) + c_2r_2(X_{g}^{\text{best}} - X_{i,k}) + W_kV_{i,k} \]

**Step 3.** Map the positions of particles to solution space and compute the corresponding fitness values according to the fitness function of the optimization problem.

**Step 4.** Update \( X_{i,k+1} \).

**Step 5.** If the stopping criterion (for example, a given maximum number of iterations) is not met, go to Step 2; otherwise stop the algorithm.

3.2. Initial Solution Generation

In this scheme, a random worker (with skill l) is assigned to each station. The initial solution for the assigning tasks to the stations is shown in a list of priority (LP). The length of this list is equal to the number of tasks which can be
assigned to the stations. The position and the value of each element in an LP represents the priority of each task. For example, if we have four tasks for assignment, a random initial LP can be shown as LP = (2, 1, 4, 3). It means Task 2 and Task 3 have the highest and lowest priorities, respectively.

Since we are going to use a PSO algorithm, and at the first time, it is developed for a continuous space, applying a method to change continuous space to discrete one is necessary. For this purpose, we use the number of column and sorting method as follows. For the above priority list, we changed these spaces as Table 1.

At first, this list is generated randomly and the tasks of the list are assigned to the mated-stations with considering their operation direction, precedence constraints and priority values. Then the set of assignable tasks is updated, and this process continues until there is no task for assignment.

Note: In this process, if the time of a station after adding a new task is greater than the cycle time; a new mated-station is created, and an operator with a random skill is assigned to it. Then the task is added to the new station.

### 3. 3. A Feasible Solution

Task-oriented and station-oriented assignments are two different procedures for balancing assembly lines [6]. In the proposed PSO algorithm, for creating a feasible solution, a station-oriented procedure based on the approach proposed by Ötcen and Toklu [12] for solving mixed-model two-sided ALBP is used. However, it was changed and adopted for the conditions of the mentioned problem.

In this paper, if a mated-station is opened, according to the direction and the priority of the task which should be assigned, a worker with a random skill will be assigned to this station. It causes to have a simultaneous worker assignment and line balancing. If both sides of the mated-station are loaded as much as possible, then the current mated-station is closed, and another mated-station will be created to assign the rest of tasks.

The flowchart of the current procedure and its notations are shown as follows:

- **NM**: Number of mated-station
- **NL**: Number of left-side station
- **NR**: Number of right-side station
- **AT**: Set of assignable tasks
- **\( L_{SNM}^1 \)**: The load of station including unavoidable idle times on the left-side station of the current mated-station for all \( m \in M \)
- **\( L_{SNM}^2 \)**: The load of station including unavoidable idle times on the right-side station of the current mated-station for all \( m \in M \)
- **\( ST_{NM}^1 \)**: The set of tasks which are assigned to the left side station of the current mated-station
- **\( ST_{NM}^2 \)**: The set of tasks which are assigned to the right-side station of the current mated-station
- **\( S_i \)**: Number of skill \( l \)

<table>
<thead>
<tr>
<th>TABLE 1. Changing continuous space to discrete space</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Start</strong></td>
</tr>
<tr>
<td><strong>Step 1</strong></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
</tr>
</tbody>
</table>

### 3. 4. Objective Functions

The objectives of the proposed algorithm for mixed-model two-sided assembly line balancing problems and worker assignment with different levels of skills for a given cycle time are as follows:

1. Minimizing the number of mated-stations. It is equivalent to maximize the weighted line efficiency. According to the nature of mixed-model assembly line and the skills of the workers, the weighted line efficiency for a given line balance is calculated as follows:

   \[
   WLE = \frac{\sum_{m \in M} q_m (\sum_{l=1}^{m} l \times S_l)}{C_{NS}} \times 100
   \]

   where \( q_m \) is computed by Eq. (25). In this equation, \( D_m \) denotes the demand, over the planning horizon, for model \( m \).

   \[
   q_m = \frac{D_m}{\sum_{m \in M} D_m}
   \]

2. Minimizing the weighted smoothness index. By using this index, the idle time between the stations will be as equal as possible. The following equation is used for computing this index:

   \[
   WSI = \frac{\left( \sum_{m \in M} q_m (\sum_{j=1}^{2} \sum_{l=1}^{m} l \times S_l^2 - L_{S_{max}}^2) \right)^{1/2}}{N_{S}}
   \]

   where, \( L_{S_{max}} \) is the maximum station time, including unavoidable idle times.

3. Minimizing the total human cost (HC). It can be calculated as follows:

   \[
   HC = \sum_{j=1}^{2} \sum_{k=1}^{m} l \times S_l \times H C_{AK} \times G_{jkl}
   \]

   Based on the weighted sum method [33], the objective function of the proposed approach can be shown as:

   \[
   Minimize \quad E = W_1 (\frac{WLE}{W_{LE}}) + W_2 (\frac{WSI}{W_{SI}}) + W_3 (\frac{HC}{HC_0})
   \]
where, \( WLE_0 \), \( WSI_0 \) and \( HC_0 \) are the initial objective function values and \( W_1 \), \( W_2 \) and \( W_3 \) represent the importance of each objective in this method.

Equation (28) shows the weighted line efficiency will be maximized and the weighted smoothness index and human cost will be minimized for a given cycle time.

If \( W_1 = W_2 = W_3 = \frac{1}{3} \), then the integrated objective function \( (E) \) can be as follows:

\[
\text{Minimize } E = \frac{1}{3} \left( \frac{WLE}{WLE_0} \right) + \frac{1}{3} \left( \frac{WSI}{WSI_0} \right) + \frac{1}{3} \left( \frac{HC}{HC_0} \right)
\]

(29)

3.5. Lower Bound  
A lower bound for the number of stations of mixed-model two-sided assembly lines based on the lower bound presented by Hu et al. [34] was presented in the literature [12]. It is adopted for mixed-model TSALBP and worker assignment with different levels of skill. In this equation, \( t_{ml} \) is used for operation time. It means if all tasks are done by high-skilled workers, the number of stations will be minimized.

\[
\begin{align*}
\text{Max} &= \max \left\{ \sum \sum \rho \eta \Psi \eta \tau \right\} \\
\text{LB} &= 2 \cdot \text{Max} + \\
\text{max} &= \frac{\sum \sum \rho \eta \Psi \eta \tau \eta \tau}{c}
\end{align*}
\]

(30)

(31)

4. COMPUTATIONAL RESULTS AND DISCUSSION

In this section, the performance and the efficiency of the proposed approach is investigated on a set of test problems in terms of running time and solution quality. Moreover, a numerical example is solved by the mentioned algorithm in detail. Furthermore, we compare the proposed PSO algorithm with the SA published by Özcan and Toklu [12] for mixed-model TSALBP. But their algorithm is adopted for the assumption of this paper. Both algorithms are coded in MATLAB software and run on a personal computer with 2.2 GHz Intel Core 2 Due CPU and 1 GB of RAM memory.
4.1 Parameter Settings In metaheuristic algorithms, choosing the best combination of the parameters can intensify the search process and prevents premature convergence. So, setting the parameters can influence on the performance of these algorithms.

In the proposed particle swarm optimization algorithm, the Taguchi method [35] is used for the best parameter selections. Three levels are selected for each parameter \((C_1, C_2, W_{max}, W_{min}\) and the swarm size), and they are shown in Table 2.

Taguchi method uses orthogonal arrays for decreasing the number of experiments for parameters setting. These arrays for the proposed approach are presented in Table 3 which shows 16 tests with different levels are necessary to select the best value for each parameter. We examine these levels for a mixed-model TSALBP with 30 tasks and worker assignment with different three skill levels. Each test is run five times, and the average of the objective function is obtained to calculate the \((S/N)\) ratio. In the Taguchi method, the \(S/N\) ratio is as follows [30]:

\[
S/N = -10 \log_2 \left( \sum_{i=1}^{4} \text{(objective function)}^2 \right)
\]  

(32)

The larger \(S/N\) ratio is equivalent to the least objective function. So, each factor's level which shows the maximum \(S/N\) ratio is the best one. Therefore, according to Figure 3, the best level of each factor obtained by Taguchi method is reported in Table 4.

4.2 Numerical Experiments In this section, the efficiency of the PSO algorithm for solving mixed-model TSALBP and worker assignment with different levels of skills is examined over a set of benchmarked test problems \((P9, P12, P14, P20, P25, P30, P39, P47\) and \(P65\)). In these experiments, it is supposed that the human cost of a worker with skill 1, 2 and 3 are 90, 60 and 40 dollars per period, respectively. Table 5 presents these obtained results.

Also, the results of the proposed PSO algorithm are compared with the SA algorithm published by Özcan and Toklu [12] for mixed-model TSALBP. But we adopted their algorithm for the assumption of this paper.

For this purpose, each test problem is solved five times with both algorithms and then the best and the average objective function \((E)\), and also the average of WSI, WLE, human cost and the running times are reported. Figure 4 and Figure 5 compare the best and the average of both algorithms for minimization of \(E\) over several test problems. As can be seen, the PSO algorithm provides better results than the SA algorithm.

Figure 6 and Figure 7 indicate the obtained results of WSI and human cost for different problems by PSO and SA algorithms. Since minimization of both objective functions are considered, the proposed PSO can have better results than the SA algorithm.

Figure 8 demonstrates the average results for maximization of WLE for both algorithms. Clearly, the PSO has larger values of WLE than the SA algorithm.

As well as the comparisons between the objective functions of these problems, the average running times of the proposed PSO algorithm are compared with the SA algorithm.

Figure 9 shows the running time of the PSO algorithm is more than the SA algorithm. Its reason may be for population-based PSO algorithm. But since assembly line balancing and worker assignment problems are related to long-term decision making problems, these differences may be ignorable. Because this decision is usually made one time during several years and in this situation, having better results may be more beneficial than taking more time per second.

<table>
<thead>
<tr>
<th>TABLE 2: Factors and their levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor</td>
</tr>
<tr>
<td>Level</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TABLE 3: The use of orthogonal arrays</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>12</td>
</tr>
<tr>
<td>13</td>
</tr>
<tr>
<td>14</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>16</td>
</tr>
</tbody>
</table>

* The number of tasks
**TABLE 4.** Selected levels of the PSO algorithm

<table>
<thead>
<tr>
<th>Factor</th>
<th>Swarm size</th>
<th>Cognitive coefficient ($C_2$)</th>
<th>Social coefficient ($C_3$)</th>
<th>Maximum inertia weight ($W_{max}$)</th>
<th>Minimum inertia weight ($W_{min}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Value</td>
<td>n</td>
<td>0.6</td>
<td>0.6</td>
<td>1</td>
<td>0.45</td>
</tr>
</tbody>
</table>

![Figure 3. The mean S/N ratio plot for the selected levels of each factor](image)

**TABLE 5.** The obtained results of the proposed algorithm for different problems

<table>
<thead>
<tr>
<th>C</th>
<th>LB</th>
<th>E</th>
<th>WSI</th>
<th>Human cost</th>
<th>WLE%</th>
<th>SMI(NS)</th>
<th>(S1,S2,S3)</th>
<th>ET*</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>6</td>
<td>1</td>
<td>0.32</td>
<td>0.30</td>
<td>0.25</td>
<td>180</td>
<td>180</td>
<td>81.25</td>
</tr>
<tr>
<td>9</td>
<td>7</td>
<td>1</td>
<td>0.36</td>
<td>0.33</td>
<td>0.56</td>
<td>150</td>
<td>150</td>
<td>94.64</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>0.39</td>
<td>0.35</td>
<td>0.93</td>
<td>0.93</td>
<td>134</td>
<td>130</td>
<td>90.62</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>0.43</td>
<td>0.39</td>
<td>0.93</td>
<td>0.93</td>
<td>130</td>
<td>130</td>
<td>80.55</td>
</tr>
</tbody>
</table>

![Image](image)

**Figure 3.** The mean S/N ratio plot for the selected levels of each factor

| P | 5  | 1 | 0.40| 0.34       | 0.74 | 258     | 250        | 90.12|24 [1.3,1.2,1.1] |
| 12 | 6  | 1 | 0.29| 0.24       | 0.15 | 180     | 180        | 95.42|12 [2.0,2.0,0.0] |
| 7  | 1  | 0.32| 0.29| 0.19       | 0.15 | 180     | 180        | 81.79|12 [2.0,2.0,0.0] |
| 8  | 1  | 0.33| 0.31| 0.61       | 0.61 | 150     | 150        | 84.36|12 [1.1,1.1,1.1] |

![Image](image)

**Figure 4.** The obtained results of the proposed algorithm for different problems

| P | 10 | 2 | 0.45| 0.41       | 2.65 | 3.21    | 392        | 430  |71.37|36 [1.1,1.1] |
| 14 | 11 | 2 | 0.45| 0.43       | 2.77 | 3.41    | 346        | 350  |71.37|36 [1.1,1.1] |
| 12 | 1  | 0.48| 0.45| 3.20       | 3.23 | 324     | 320        | 80.60|36 [1.1,1.1] |
| 13 | 1  | 0.45| 0.45| 2.36       | 2.87 | 306     | 300        | 82.93|36 [1.1,1.1] |

![Image](image)

**Figure 5.** The obtained results of the proposed algorithm for different problems

| P | 20 | 1 | 0.53| 0.48       | 5.75 | 5.08    | 396        | 360  |74.51|56 [2.1,1.3] |
| 20 | 22 | 0 | 0.52| 0.49       | 7.25 | 6.90    | 350        | 330  |71.16|36 [1.1,1.1] |
| 24 | 0  | 0.48| 0.45| 6.02       | 5.73 | 320     | 280        | 77.21|36 [1.1,1.1] |
| 26 | 0  | 0.47| 0.45| 5.91       | 5.33 | 296     | 280        | 80.44|36 [1.1,1.1] |

![Image](image)

**Figure 6.** The obtained results of the proposed algorithm for different problems

| P | 35 | 1 | 0.51| 0.48       | 10.19| 9.93    | 438        | 420  |77.49|76 [2.1,1.3] |
| 25 | 38 | 1 | 0.53| 0.49       | 10.94| 9.46    | 386        | 360  |74.18|76 [2.1,1.3] |
| 41 | 0  | 0.49| 0.44| 8.87       | 8.48 | 388     | 370        | 82.39|36 [1.1,1.1] |
| 44 | 1  | 0.48| 0.44| 8.93       | 8.28 | 324     | 300        | 82.29|36 [1.1,1.1] |

![Image](image)

**Figure 7.** The obtained results of the proposed algorithm for different problems

| P | 25 | 1 | 0.43| 0.41       | 2.90 | 2.28    | 384        | 370  |90.83|77 [2.1,1.3] |
| 30 | 27 | 1 | 0.42| 0.38       | 2.98 | 2.41    | 332        | 320  |91.34|77 [2.1,1.3] |
| 29 | 1  | 0.41| 0.40| 3.71       | 3.46 | 288     | 280        | 88.60|36 [1.1,1.1] |
| 31 | 1  | 0.45| 0.40| 5.37       | 4.98 | 260     | 250        | 86.08|36 [1.1,1.1] |

![Image](image)

**Figure 8.** The obtained results of the proposed algorithm for different problems

| P | 24 | 1 | 0.45| 0.39       | 4.65 | 3.71    | 494        | 450  |82.06|77 [2.1,1.3] |
| 39 | 26 | 1 | 0.48| 0.44       | 5.40 | 4.62    | 430        | 410  |81.47|77 [2.1,1.3] |
| 28 | 1  | 0.51| 0.48| 6.40       | 5.96 | 400     | 380        | 80.10|36 [1.1,1.1] |
| 30 | 0  | 0.44| 0.43| 4.19       | 3.59 | 372     | 370        | 88.88|36 [1.1,1.1] |

![Image](image)

**Figure 9.** The obtained results of the proposed algorithm for different problems

| P | 45 | 1 | 0.64| 0.60       | 18.24| 16.42   | 732        | 700  |70.55|77 [2.1,1.3] |
| 47 | 50 | 1 | 0.59| 0.56       | 15.83| 14.54   | 644        | 630  |76.67|77 [2.1,1.3] |
| 55 | 1  | 0.58| 0.54| 16.40      | 15.70| 582     | 560        | 76.19|50 [1.1,1.1] |
| 60 | 1  | 0.54| 0.53| 15.22      | 14.40| 506     | 490        | 82.03|48 [1.1,1.1] |

![Image](image)

**Figure 10.** The obtained results of the proposed algorithm for different problems

| P | 300 | 0 | 0.45| 0.41       | 66.6 | 64.1    | 606        | 560  |81.6 |48 [1.1,1.1] |
| 65 | 320 | 0 | 0.45| 0.43       | 73.3 | 68.7    | 512        | 480  |80.4 |48 [1.1,1.1] |
| 340 | 0 | 0.45| 0.40| 65.9       | 47.8 | 552     | 480        | 85.1 |48 [1.1,1.1] |
| 360 | 0 | 0.42| 0.40| 52.0       | 44.6 | 440     | 420        | 87.3 |48 [1.1,1.1] |

**W**: The worst result, **M**: The average result, **B**: The best result, **ET**: Elapsed time, S1: Skill 1, S2: Skill 2, S3: Skill 3
5. CONCLUSION

This paper dealt with a multi-objective mixed-model TSALBP with assignment workers when task operation time is dependent on the skills of workers. Three objectives in this paper were considered: (1) minimizing the number of mated-stations, (2) minimizing the number of stations, and (3) minimizing the total human cost for a given cycle time. The first and the second objectives of this problem are equivalent to maximize the weighted line efficiency and minimize the weighted smoothness index.

A PSO algorithm is presented to solve this problem and also the Taguchi method is used for its parameter settings. Several test problems with different cycle times are solved by the proposed algorithm. Also the efficiency of this algorithm was evaluated with the SA algorithm published by Özcan and Toklu [12] over different problems. The obtained results indicated the solution quality of the proposed algorithm was better than the solution of the SA algorithm.

As well as using other metaheuristics for this problem, this research can be enriched with other assumptions such as learning effect of workers, U-shaped and parallel stations for future researches.
6. REFERENCES


Simultaneous Multi-skilled Worker Assignment and Mixed-model Two-sided Assembly Line Balancing

P. Fattahi\textsuperscript{a}, P. Samouei\textsuperscript{b}, M. Zandieh\textsuperscript{c}

\textsuperscript{a}Department of Industrial Engineering, Faculty of Engineering, Alzahra University, Tehran, Iran
\textsuperscript{b}Department of Industrial Engineering, Faculty of Engineering, Bu-Ali Sina University, Hamedan, Iran
\textsuperscript{c}Department of Industrial Management, Management and Accounting Faculty, Shahid Beheshti University, Tehran, Iran

\textbf{Paper Info}

\textbf{Paper history:}
Received 28 July 2015
Received in revised form 06 January 2016
Accepted 28 January 2016

\textbf{Keywords:}
Worker Assignment
Mixed-model
Skill
Particle Swarm Optimization
Simulated Annealing
Taguchi Method

In this paper, a mathematical model for solving the problems of bi-directional mixed-model two-sided assembly line balancing with the simultaneous worker assignment and multi-skilled workforce is developed. The time of operations, dependent on the skill level of the operators, is considered. Therefore, three different objective functions are considered, which are: 1) the number of stations, 2) the number of operators, and 3) the total cost of workers in a defined cycle time. In addition, for the sake of maximizing the weighted line efficiency and minimizing the variation of loading of stations, two goals were considered. This problem is NP-hard, so a particle swarm optimization algorithm is developed and its results are compared with those of the simulated annealing algorithm, which is also used in the new problem. In addition, different numerical examples were solved by both algorithms and the results were compared in terms of solution quality and execution time. The results show that the proposed algorithm in this paper has a higher efficiency than the other.

\textbf{doi}: 10.5829/idosi.ije.2016.29.02b.10