

Article

# Analytic Hierarchy Process and Multilayer Network-Based Method for Assembly Line Balancing

László Nagy, Tamás Ruppert and János Abonyi \* 

MTA-PE “Lendület” Complex Systems Monitoring Research Group, University of Pannonia, P.O. Box 158, H-8200 Veszprém, Hungary; laszlo.nagy@fmt.uni-pannon.hu (L.N.); ruppert@abonyilab.com (T.R.)

\* Correspondence: janos@abonyilab.com

Received: 30 April 2020; Accepted: 2 June 2020; Published: 5 June 2020



**Abstract:** Assembly line balancing improves the efficiency of production systems by the optimal assignment of tasks to operators. The optimisation of this assignment requires models that provide information about the activity times, constraints and costs of the assignments. A multilayer network-based representation of the assembly line-balancing problem is proposed, in which the layers of the network represent the skills of the operators, the tools required for their activities and the precedence constraints of their activities. The activity–operator network layer is designed by a multi-objective optimisation algorithm in which the training and equipment costs as well as the precedence of the activities are also taken into account. As these costs are difficult to evaluate, the analytic hierarchy process (AHP) technique is used to quantify the importance of the criteria. The optimisation problem is solved by a multi-level simulated annealing algorithm (SA) that efficiently handles the precedence constraints. The efficiency of the method is demonstrated by a case study from wire harness manufacturing.

**Keywords:** assembly-line balancing; multi-objective optimization; simulated annealing; multilayer network

## 1. Introduction

Production line-based assembly lines are still the most widely applied manufacturing systems [1]. Assembly-Line Balancing (ALB) [2] deals with the balanced assignment of tasks to the workstations, resulting in the optimisation of a given objective function without violating precedence constraints [3]. The efficiency of these optimisation tasks is mostly determined by the model of the manufacturing process represented [4].

The concept of Industry 4.0 has already had a significant influence on how production and assembly lines are designed [5] and managed [6]. The requirement of practical design at a high automation level ensures that sensors and equipment can be integrated in a fast, secure, and reliable way. In our research, we study how the interoperability capabilities of Industry 4.0 solutions can be improved and how the efficiency of solution development can be increased.

As Internet of Things-based products and processes are rapidly developing in the industry, there is a need for solutions that can support their fast and cost-effective implementation. There is a need for further standardization to achieve more flexible connectivity, interoperability, and fast application-oriented development; furthermore, advanced model-based control and optimisation functions require a better understanding of sensory and process data [7].

Usually, production systems include multiple subsystems and layers of connectivity. Thus, although research-based solutions for classical operations typically use a graph-based representation of problems and flow-based optimisation algorithms, conventional single-layer networks quickly become incapable of representing the complexity and connectivity of all the details of the production

line. With the overlapping data in Industry 4.0 solutions, it should be highlighted that multilayer networks are expected to be the most suitable options for representing modern production lines. The concept of a multilayer network was developed to represent multiple types of relationships [8], and these models have been proven to be applicable to the representation of complex connected systems [9]. Network-based models can also represent how products, resources and operators are connected [10], which is beneficial in terms of solving manufacturing cell formation problems [11]. This work demonstrates how the multilayer network representation of production lines can be utilised in line balancing.

In the proposed novel network model, the layers represent the skills of the operators, the tools required for the activities, and the precedence constraints of the activities. At the same time, a multi-objective optimisation algorithm designs the assignment of activities and operators to network layers. The proposed multilayer network approach supports the intuitive formulation of multi-objective line balancing optimisation tasks. Besides the utilisation of operators, the utilisation of the tools and the number of skills of an operator are also taken into account. The main advantage of the proposed network-based representation is that the latter two objectives are directly related to the structural properties of the optimised network.

Line balancing is a non-deterministic polynomial-time hard (NP-hard) optimisation problem, which means that the computational complexity of the optimisation problem increases exponentially as the dimensions of the problem increase. This challenge explains why numerous meta heuristic approaches such as simulated annealing (SA) [12,13], hybrid heuristic optimisation [13,14], chance-constrained integer programming [15], recursive and dynamic programming [16], as well as tabu search [17] have been utilised in the field of production management. Fuzzy set theory provides a transparent and interpretable framework to represent the uncertainty of information and solve the ALB problem [18,19]. Among the wide range of heuristic methods capable of achieving reasonable solutions [20], SA is the most widely used search algorithm [21], so it has already been applied to solve mixed and multi-model line-balancing problems [22].

To deal with the complexity of this problem, an SA algorithm was also developed. The proposed algorithm utilises a unique problem-oriented sequential representation of the assignment problem and applies a neighbourhood-search strategy that generates feasible task sequences for every iteration. Since the algorithm has to handle multiple aspects of line balancing, the analytic hierarchy process (AHP) technique is used to quantify the importance of the objectives, also known as Saaty's method [23]. AHP is a method for multi-criteria decision-making which is used to evaluate complex multiple criteria alternatives involving subjective judgments [24]. This method is a useful and practical approach to solving complex and unstructured decision-making problems by calculating the relative importance of the criteria based on the pairwise comparison of different alternatives [25]. The method has been widely applied thanks to its effectiveness and interpretability. Two papers were found in which it has already been applied to determine the cost function of multi-objective SA optimisation problems. In the first case study of supplier selection, AHP was applied to calculate the weight of every objective by applying the Taguchi method [26] (Adaptive Tabu Search Algorithm—ATSA [27]). In contrast, in the second report, this concept was applied to the maintenance of road infrastructure [28].

The novelties of the work are the following:

- In Section 2, the main problem formulation is introduced, including the multilayer network representation of multiple aspects concerning the balancing of production lines, and the details of the objective function.
- In Section 3, an SA algorithm will be introduced based on a novel sequential representation of the line-balancing problem and the search algorithm that guarantees the fulfilment of the precedence constraints.
- Section 4 demonstrates how AHP can be used to aggregate the multi layer network-represented objectives of the line-balancing problem for SA.

## 2. Problem Formulation

In this section, the problem formulation is presented. First, the representation of production line modelling with the multilayer network is introduced in Section 2.1. The details of the minimised function and its AHP-based aggregation are given in Section 2.2.

### 2.1. Multilayer Network-Based Representation of Production Lines

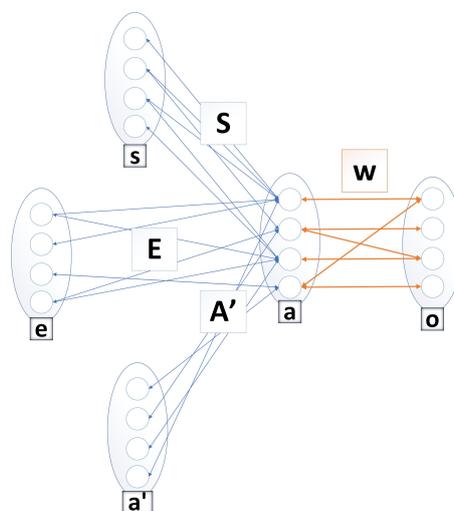
The proposed network model of the production line consists of a set of bipartite graphs that represent connections between operators,  $\mathbf{o} = \{o_1, \dots, o_{N_o}\}$ ; skills of the operators needed to perform the given activity,  $\mathbf{s} = \{s_1, \dots, s_{N_s}\}$ ; equipment,  $\mathbf{e} = \{e_1, \dots, e_{N_e}\}$ ; activities (operations),  $\mathbf{a} = \{a_1, \dots, a_{N_a}\}$ ; and the precedence constraints between activities,  $\mathbf{a}' = \{a'_1, \dots, a'_{N_{a'}}\}$ . The relationships between these sets are defined by bipartite graphs  $G_{i,j} = (O_i, O_j, E_{i,j})$  represented by  $\mathbf{A}[O_i, O_j]$  biadjacency matrices, where  $O_i$  and  $O_j$  denote a general representation of the sets of objects, such that  $O_i, O_j \in \{\mathbf{s}, \mathbf{e}, \mathbf{a}', \mathbf{a}, \mathbf{o}\}$ .

The edges of these bipartite networks represent structural relationships; e.g., the biadjacency matrix  $\mathbf{A}[\mathbf{a}, \mathbf{a}']$  represents the precedence constraints or  $\mathbf{A}[\mathbf{a}, \mathbf{o}]$  represents the assignments of activities to operators. Moreover, the edge weights can be proportional to the number of shared components/resources or time/cost (see Table 1) [10].

**Table 1.** Definition of the biadjacency matrices of the bipartite networks used to illustrate how a multidimensional network can represent a production line.

	Nodes	Description
W	Activity (a)–operator (o)	Operator assigned to the activity
S	Activity (a)–skill (s)	Skill/education required for a category of activities
E	Activity (a)–equipment (e)	Equipment which is in use in an activity
A'	Activity (a)–activity (a')	Precedence constraint between activities

As can be seen in Figure 1, these bipartite networks are strongly connected. The proposed model can be considered as an interacting or interconnected network [8], where bipartite networks define the layers. Since different types of connections are defined, the model can also be handled as a multidimensional network. As illustrated in Figure 2, when relationships between the sets  $O_i$  and  $O_j$  are not directly defined, it is possible to evaluate the relationship between their elements  $o_{i,k}$  and  $o_{j,l}$  in terms of the number of possible paths or the length of the shortest path between these nodes [10].



**Figure 1.** Illustrative network representation of a production line. The definitions of the symbols are given in Table 1.

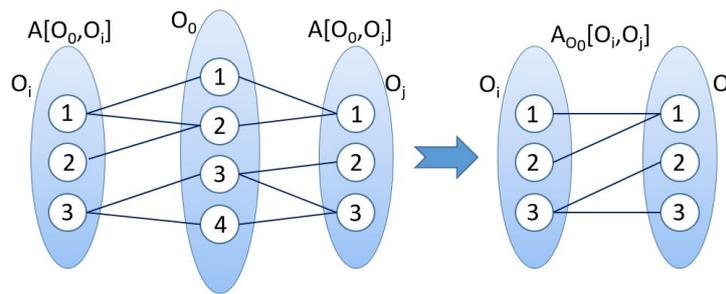


Figure 2. Projection of a property connection.

In the case of connected unweighted multipartite graphs, the number of paths intersecting the set  $O_0$  can be easily calculated based on the connected pairs of bipartite graphs as follows:

$$A_{O_0}[O_i, O_j] = A[O_0, O_i]^T \times A[O_0, O_j]. \tag{1}$$

In the proposed network model, the optimisation problem is defined by the allocation of tasks that require the allocation of different skills and tools to an operator that might necessitate extra training, labour and investment costs. The main benefit of the proposed network representation is that these costs can be directly evaluated based on the products of the biadjacency matrices  $S$  and  $W$ :

$$A[s, o] = A[s, a]A[a, o] = SW. \tag{2}$$

The resultant network  $A[s, o]$  represents how many times a given skill should be utilised by an operator, while its unweighted version  $A^u[s, o]$  models which skills the operators should have.

The design of the presented network model is based on the analysis of the semantically standardized models of production lines [29], and the experience gained in the development project connected to the proposed case study. The details of the multilayer network-based modelling of a wire-harness production process can be found in [10].

### 2.2. The Objective Function

A simple assembly line balancing problem (SALBP) assigns  $N_a$  tasks/activities to  $N_o$  workstations/operators. Each activity is assigned to precisely one operator, and the sum of task times of workstation should be less or equal to the cycle time  $T_c$  [30]. Precedence relations between activities must not be violated [31]. There are two important variants of this problem [32]: SALBP-1 aims to minimise  $N_o$  for a given  $T_c$ , while the goal of SALBP-2 is to minimise  $T_c$  for a predefined  $N_o$  [1,33,34]. In this paper, the SALBP-2 problem was investigated and extended to include the following skill and equipment-related objective functions:

Station-time-related objective: The main objective of line balancing is to minimise the cycle time  $T_c$ , which is equal to the sum of the maximum of the station times  $T_j$ . The utilisation of the whole assembly line can be calculated as follows:

$$T_c = \arg \max_j T_j = \sum_{i=1}^{N_a} w_{i,j} t_i, \tag{3}$$

where  $t_i$  represents the elementary activity times of the  $a_i$ -th activity.

As the theoretical minimum of  $T_c$  is

$$T_c^* = \frac{\sum_{i=1}^{N_a} t_i}{N_o}, \tag{4}$$

the following ratio evaluates the efficiency of the balancing of the activity times:

$$Q_T(\pi) = \frac{T_c^*}{T_c} = \frac{\sum_{i=1}^{N_a} t_i}{\sum_{i=1}^{N_a} w_{i,j} t_i} \tag{5}$$

Skill-related (training) objective: The training cost is calculated with the node degree between skill-operator elements  $s - o$ . The number of skills  $N_s$  is divided by the sum of the node degrees  $k_i$  between sub-networks  $s$  and  $o$  in the multilayer representation:

$$Q_S(\pi) = \frac{N_s}{\sum_i^{s-o,o} k_i} \tag{6}$$

Equipment-related objective function: The equipment cost is calculated with the node degree between equipment-operator elements  $e - o$ . The number of pieces of equipment  $N_e$  is divided by the sum of the node degrees  $k_i$  between sub-networks  $e$  and  $o$  in the multilayer representation:

$$Q_E(\pi) = \frac{N_e}{\sum_i^{e-o,o} k_i} \tag{7}$$

Since the importance of these objectives is difficult to quantify, a pairwise comparison is used to evaluate their relative importance, and the analytic hierarchy process (AHP) is used to determine the weights  $\lambda$  in the objective function:

$$Q(\pi) = \lambda_1 Q_T(\pi) + \lambda_2 Q_S(\pi) + \lambda_3 Q_E(\pi), \tag{8}$$

where  $Q_T(\pi) \in [0, 1]$  represents the balance of the production line, and  $Q_S(\pi) \in [0, 1]$  and  $Q_E(\pi) \in [0, 1]$  measure the efficiency of how the skills and tools are utilised, respectively.

The application of AHP-based weighting is beneficial to integrate the normalised values of the easy to evaluate station-time and equipment-related objectives, and the less specific training-related costs. Although the pairwise comparison of the importance of these objectives and cost-items is subjective, the consistency of the comparisons can be evaluated based on the numerical analysis of the resulted comparison matrices (which will be shown in the next section), which clarifies the reason for our choice of AHP as an ideal tool to extract expert knowledge for the formalisation of the cost function.

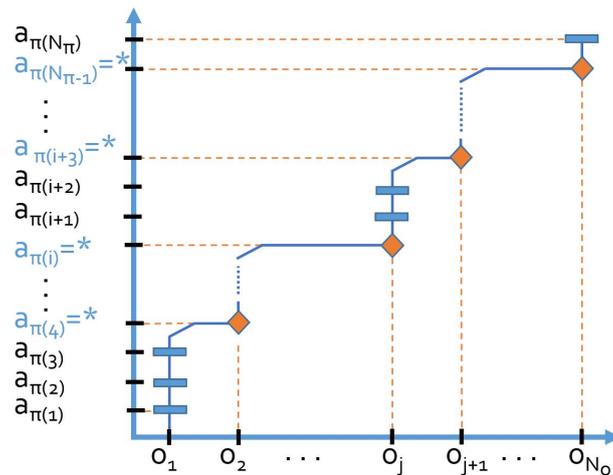
### 3. Simulated Annealing-Based Line-Balancing Optimization

This section presents the proposed optimisation algorithm. The representation of the SA problem is introduced in Section 3.1. Section 3.2 discusses how the precedence constraints of the activities are represented, while Section 3.3 presents how the assignment of activities to operators is formulated by a sequencing problem that can be efficiently solved by the proposed simulated annealing algorithm.

#### 3.1. Representation of the Problem

In the proposed network representation (Figure 1), the assignment of activities to operators is defined by the elements  $w_{i,j}$  of the matrix  $\mathbf{W}$  that represent the  $i$ th activity assigned to the  $j$ th operator. Instead of the direct optimisation of these  $N_a \times N_o$  elements, a sequence  $N_\pi = N_a + N_o - 1$  is optimised, where  $N_a$  represents the number of activities and  $N_o$  denotes the number of operators.

The concept of sequence-based allocation is illustrated in Figure 3, where the horizontal axis represents the fixed order of the operators  $o_j$  and the vertical axis stands for the activities  $a_i$ , where  $\pi(i)$  represents the index of the activity by the  $i$ th sequence number. The ordered activities are assigned to the operators by  $N_o - 1$  boundary elements, represented as  $a_{\pi(i)} = *$ , which ensure that the next activity in the sequence is assigned to the following operator.



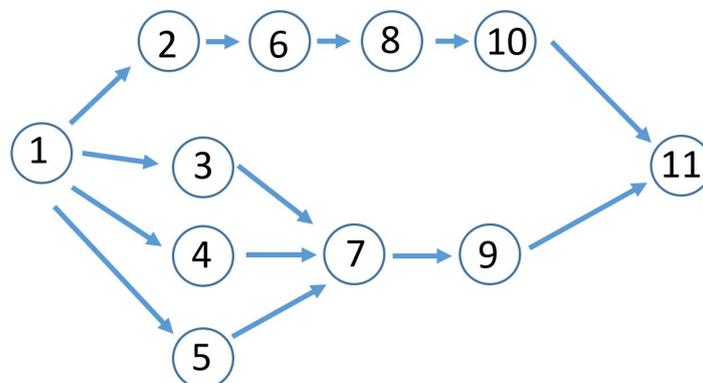
**Figure 3.** Illustration of the sequencing method. The activities are separated into the different groups of activities that are assigned to different operators.

### 3.2. Handling Precedence Constraints

In addition to these three objectives of the simulated production line, a so-called soft limit is also defined, which is the amount of the unaccomplished precedence of the activities ( $A'$ ). This limitation of the order with regard to the activities is stored in the multilayer network.

The completion of a task is a precondition for the start of another because tasks depend on other tasks. The  $\pi$  sequence has some constraining condition and cannot be entirely arbitrary. The precedence graph is used to represent these dependencies in SALBP [1,35,36]. Figure 4 shows a problem from a well-known example by Jackson [37] with  $N_a = 11$  tasks, where task 7 requires tasks 3–5 to be completed directly (direct predecessor) and task 1 indirectly (indirect predecessor). The precedence graph can be described by matrix  $A'(i, j)$ ,  $i, j = 1, 2, \dots, N_a$ , where  $A'(i, j) = 1$  if task  $i$  is the direct predecessor of task  $j$ , otherwise, it is 0 [32]. The precedence graph is partially ordered if tasks cannot be performed in parallel. It must be determined whether a permutation  $\pi = (\pi_1, \pi_2, \dots, \pi_{N_a})$  is feasible or not according to the precedence constraint.

Based on the transitive closure  $A^*$  of  $A'$ ,  $\pi$  is feasible if  $A^*(p_j, p_i) = 0, \forall i, j, i < j$ ; otherwise,  $\pi$  is infeasible [32]. A sub-sequence  $(\pi_i, \pi_{i+1}, \dots, \pi_j)$ , where  $i < j$  of  $\pi$ , can be defined by  $\pi_{(i:j)}$ . For example, a feasible sequence  $\pi$  of the precedence graph in Figure 4 is  $\pi = (1, 4, 3, 2, 5, 7, 6, 8, 9, 10, 11)$  and  $\pi_{(2:4)} = (4, 3, 2)$  is a sub-sequence of  $\pi$ .



**Figure 4.** Precedence graph of the example problem taken from Jackson [32,37].

As will be presented in the next subsection, the key idea of the algorithm is that it determines the interchangeable sets of activity pairs and uses these in the guided simulated annealing optimisation.

### 3.3. Sequence-Based Activity Grouping and Operator Assignment

The optimization algorithm is shown in Algorithm 1 and consists of the following steps:

- Generating the initial feasible sequence.
- SA I: Optimization of the sequences of the activities.
  - SA II (embedded in SA I): in the case of a specific sequence, the activities are assigned to the operators by optimizing the location of the boundary elements in sequence  $\pi$  as has been presented in Figure 3, so SA I uses a cost function that relates to the optimal assignment.

---

**Algorithm 1:** Pseudocode of the proposed SA-ALB algorithm

---

```

Input:  $s, e, Time, Precedence$ 
Output:  $\pi, Q(\pi)$ 
Annealing:  $maxiter, T, T_{max}, T_{min}, m_{max}, m_{min}$ 
1  $\alpha = (\frac{T_{min}}{T_{max}})^{1/maxiter}, T^1 = T_{max}$ 
2  $\alpha_m = (\frac{m_{min}}{m_{max}})^{1/maxiter}, m^1 = m_{max}$ 
3 Begin
4  $T = T_{max}; T_{max} =$  maximum value of temperature
5 while  $T < T_{min}; T_{min} =$  minimum value of temperature
6
7 Generate initial sequence  $\pi$ , which satisfies the constraints
8 Generate initial placement of the boundary elements
9 Evaluate the cost function  $Q(\pi)$ , as functions (5), (6) and (7)
10 for  $i = 1$  to  $maxiter$  do
11
12     Select randomly one interchangeable activity pair
13     Interchange the activities and evaluate the new solution by implementing SA II that
        optimizes the placement of the boundary elements in this sequence
14         // SA II is working with the same principle as this main SA I
15
16      $NewQ(\pi_{new}) = Q(\pi_{new})$ 
17      $\Delta = Q(\pi_{new}) - Q(\pi)$ 
18     if  $\Delta < 0$  then
19          $\pi = \pi_{new}$ 
20          $Q(\pi) = Q(\pi_{new})$ 
21     else
22         if  $random() < exp(\frac{-\Delta}{T})$  then
23              $\pi = \pi_{new}$ 
24              $Q(\pi) = Q(\pi_{new})$ 
25      $T^{i+1} = \alpha T^i, m^{i+1} = \alpha_i m^i$ 
26     End

```

---

### 4. Case Study

This study was inspired by an industrial case study of wire harness manufacturing, where operators work with several tools that perform different activities at workstations to manufacture cables. The problem assumes that it is possible to improve the manufacturing efficiency if the resources, activities, skills and precedence are better designed.

The development of the proposed line-balancing algorithm is motivated by a development project which was defined to improve the efficiency of an industrial wire harness manufacturing process [38].

In this work, a subset of this model is used which consists of 24 activities, five operators, six skills and eight pieces of equipment.

The elementary activity times that influence the line balance were determined based on expert knowledge [39] (see Table 2).

A more detailed description of the activities, pieces of equipment and skills can be found in Tables 2–6.

**Table 2.** List of the elementary activities that should be allocated in the line balancing problem.

Activity ID	Description	Time
A1	Connector handling	4 s
A2	Connector handling	3 s
A3	Connector handling	2 s
A4	Connector handling	3 s
A5	Insert 1st end + routing	10 s
A6	Insert 2nd end	5 s
A7	Insert 1st end + routing	10 s
A8	Insert 2nd end	5 s
A9	Insert 1st end + routing	10 s
A10	Insert 2nd end	5 s
A11	Insert 1st end + routing	10 s
A12	Insert 2nd end	5 s
A13	Insert 1st end + routing	10 s
A14	Insert 2nd end	5 s
A15	Insert 1st end + routing	10 s
A16	Insert 2nd end	5 s
A17	Insert 1st end + routing	10 s
A18	Insert 2nd end	5 s
A19	Taping	15 s
A20	Taping	13 s
A21	Taping	11 s
A22	Taping	17 s
A23	Taping	15 s
A24	Quality check	10 s

The following tables give a more detailed description of the activities, equipment (Table 3) and skills (Table 4) which are involved in the proposed case study. Furthermore, the activity–equipment (Table 5) and activity–skill (Table 6) connectivity matrices show the requirements of the given base activity.

**Table 3.** List of equipment that should be allocated in the line balancing problem.

Equipment ID	Description
E1	Connector fixture
E2	Connector fixture
E3	Routing tool
E4	Insertion tool
E5	Taping tool (expert)
E6	Taping tool (normal)
E7	Taping tool (normal)
E8	Repair tool

**Table 4.** Description of skills that should be used in the studied production process.

Skill ID	Description
S1	Connector handling skill
S2	Insertion (normal) and routing skills
S3	Insertion (expert) skill
S4	Taping (normal) skill
S5	Taping (expert) skill
S6	Quality (expert) skill

**Table 5.** Activity–equipment matrix that defines which equipment are required to perform a given activity.

	E1	E2	E3	E4	E5	E6	E7	E8
A1	1	1						
A2	1	1						
A3	1	1						
A4	1	1						
A5			1					
A6				1				
A7			1					
A8				1				
A9			1					
A10				1				
A11			1					
A12				1				
A13			1					
A14				1				
A15			1					
A16				1				
A17			1					
A18				1				
A19					1			
A20					1			
A21						1	1	
A22						1	1	
A23						1	1	
A24								1

**Table 6.** Activity–skill matrix that defines which skills are required to perform a given activity.

	S1	S2	S3	S4	S5	S6
A1	1					
A2	1					
A3	1					
A4	1					
A5		1				
A6			1			
A8			1			
A9		1				
A10			1			
A11		1				
A12			1			
A13		1				
A14			1			
A15		1				

Table 6. Cont.

	S1	S2	S3	S4	S5	S6
A7		1				
A16			1			
A17		1				
A18			1			
A19				1		
A20				1		
A21					1	
A22					1	
A23					1	
A24						1

The tables illustrate that the practical implementation of line balancing problems is also influenced by how much equipment is needed for the designed production line and how many skills should be learnt by the operators.

All the collected information is transformed into network layers, as shown in Figure 5. The top of the figure shows the bipartite networks that represent the details of the assignments, while the bottom of the figure represents the tree layers of the network that define the activity–operator, skill–operator, and equipment–operator assignments. As can be seen, this representation is beneficial as it shows how similar operators, skills and equipment can be grouped into clusters.

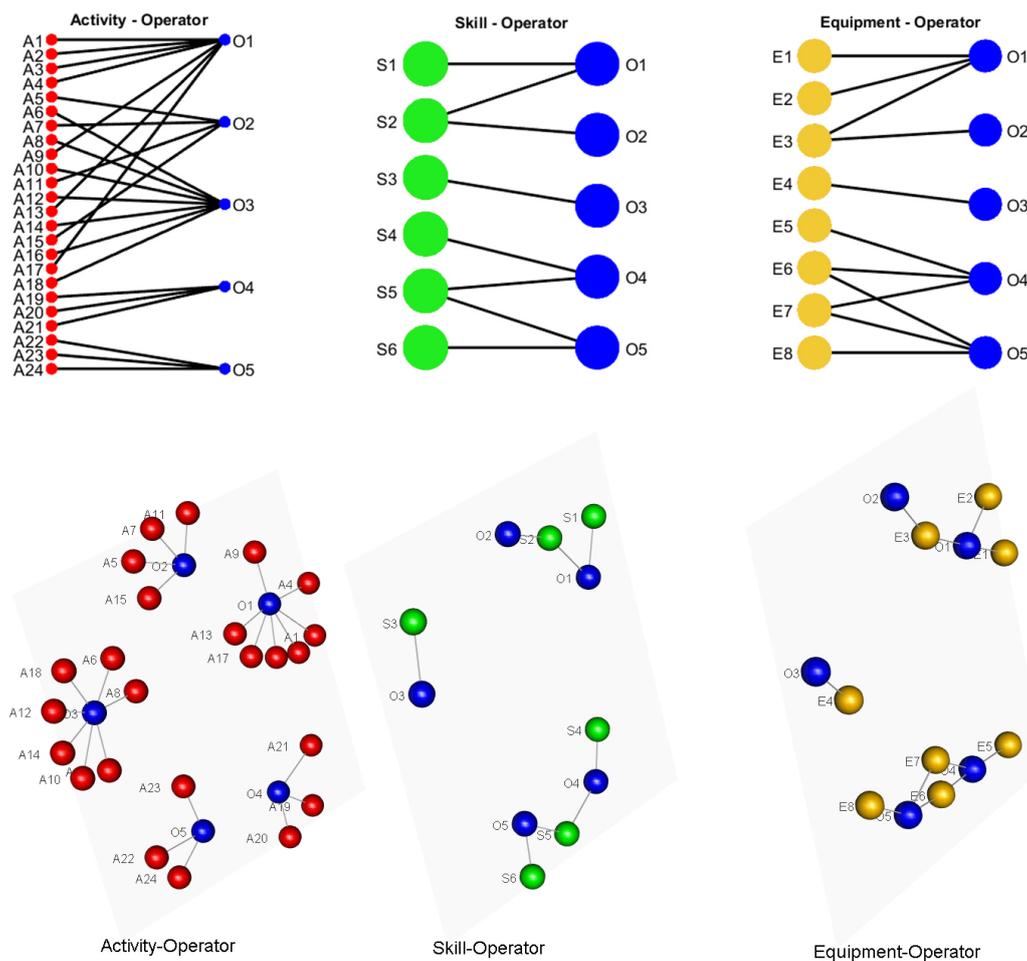


Figure 5. Illustration of the skill–operator and equipment–operator assignments after line balancing.

Although this is not shown in the figure, the weights of the edges represent the costs or benefits of the assignments. The final form of the network is formed based on a multi-objective optimisation of the sets of active edges.

As some of these objectives are difficult to measure, we utilised the proposed AHP-based method to convert the pairwise comparisons of the experts into weights of criteria. The structure of the decision problem is represented in Figure 6. As this figure illustrates, the AHP is used to compare difficult to evaluate equipment and skill assignment costs and the importance of the objectives. The pairwise comparison was performed by a process engineer, and the resulting comparison matrices can be found in Tables 7–9. Based on the analysis of the the eigenvalues of these matrices [25], we found that the evaluations were consistent.

Since the activities cannot be performed in parallel, a precedence graph defines the most crucial question, namely whether a permutation of sequence  $\pi$  is feasible. Based on the transitive closure of the adjacency matrix of the graph, the interchangeable sets of activities can be defined as depicted in Figure 7.

The result of the optimization is shown in Figure 5, which illustrates that the five operators assigned to different skills and pieces of equipment.

The reliability and the robustness of the proposed method are evaluated by ten independent runs of the optimisation algorithm to highlight how the stochastic nature of the proposed method influences the result, as well as showing the effect of the number of operators on the solutions. The aim of the analysis of the independent runs was to estimate the variance of the solutions caused by the stochastic nature of the process and the optimisation algorithm. The sample size of such repeat studies can be determined based on the statistical tests of the estimated variance. In our analysis, we found that ten experiments were sufficient to get a proper estimation of the variance (which is in line with the widely applied ten-fold cross-validation concept).

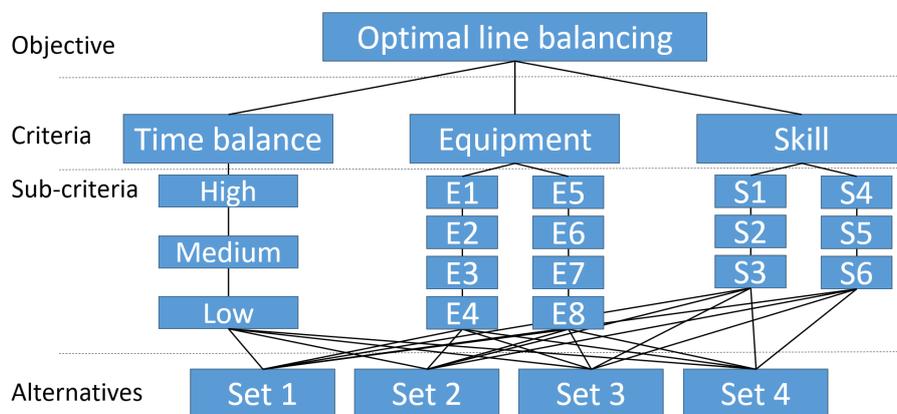


Figure 6. Analytic hierarchy process (AHP) used to solve a decision problem.

Table 7. AHP TOP matrix that shows the relative importance of the objectives. It can be seen that, in this pair-wise comparison, the skill-related cost is evaluated as being twice as important as the equipment-related costs.

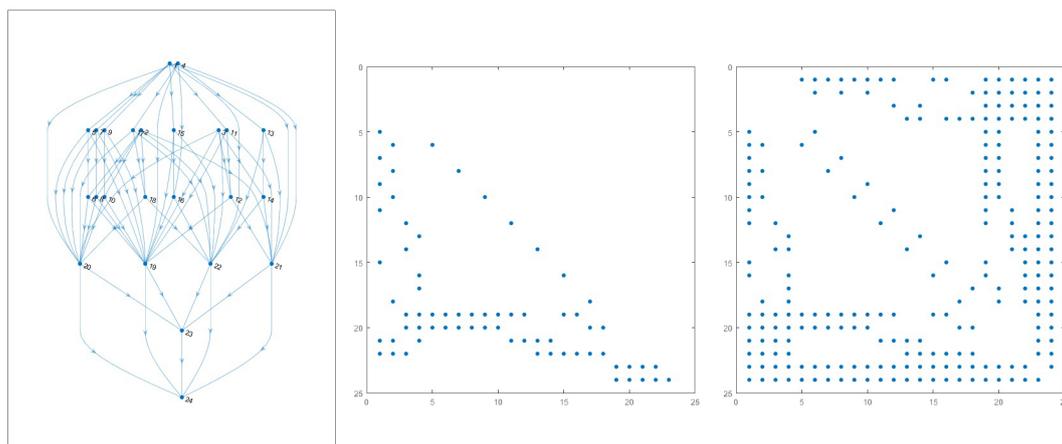
	Balancing	Equipment	Skill
Balancing		2.00	4.00
Equipment	0.50		2.00
Skill	0.25	0.50	

**Table 8.** AHP equipment matrix that shows the relative importance of the equipment.

	E1	E2	E3	E4	E5	E6	E7	E8
E1		1	0.33	0.50	2	3	3	2
E2	1		0.33	0.50	2	3	3	2
E3	3	3		2	5	7	7	5
E4	2	2	0.50		3	5	5	3
E5	0.50	0.50	0.20	0.33		2	2	1
E6	0.33	0.33	0.14	0.20	0.50		1	0.50
E7	0.33	0.33	0.14	0.20	0.50	1		0.50
E8	0.50	0.50	0.20	0.33	1	2	2	

**Table 9.** AHP skill matrix that shows the relative importance of the skills.

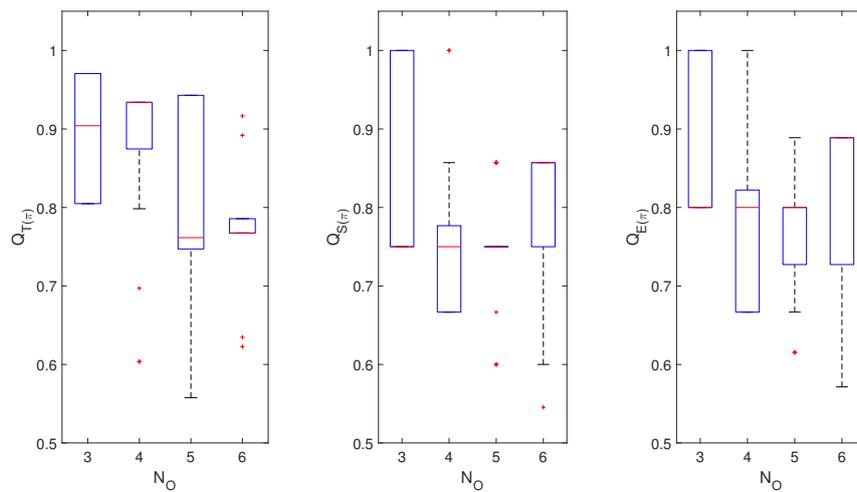
	S1	S2	S3	S4	S5	S6
S1		0.20	0.30	0.50	0.50	0.14
S2	5		2	3	3	0.50
S3	3	0.50		2	2	0.33
S4	2	0.30	0.50		1	0.20
S5	2	0.30	0.50	1		0.20
S6	7	2	3	5	5	



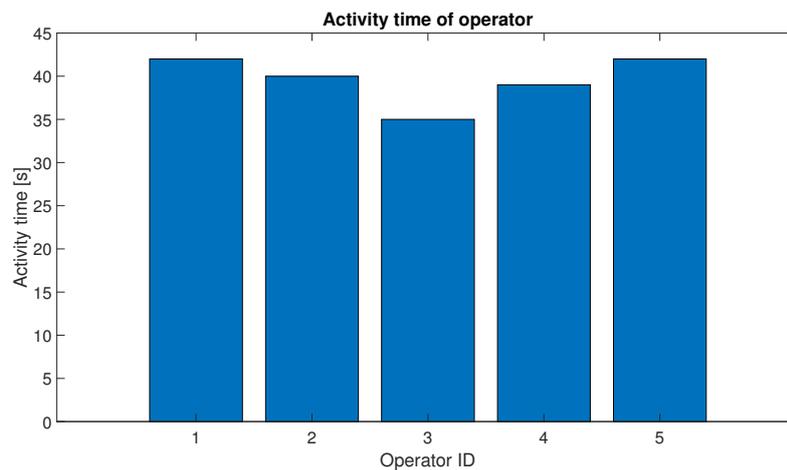
**Figure 7.** Possible path (left), precedence (middle) and transitive closure (right) of the activities (the unmarked pairs are interchangeable).

Figure 8 presents the different time, skill and equipment-related objectives in the case of different operators. As the results show, the increase in the number of operators decreases the efficiency of the utilisation of the tools and skills (this trend is the main driving force for forming manufacturing cells). The process can be well balanced in the case of 3–5 operators; e.g., in the case of five operators, in one of the best solutions, the balancing objectives are a time cost of 94.3%, training cost of 75.0% and equipment cost of 72.8%. Figure 9 shows the different total activity times of each operator during the simulation. In this case, the station times do not differ greatly, and the result is optimal [10]. The proposed algorithm was implemented in MATLAB and is available on the website of the authors ([www.abonyilab.com/about-us/software-and-data](http://www.abonyilab.com/about-us/software-and-data)); interested readers can make further comparisons, and the proposed problem can serve as a benchmark for constrained multi-objective line balancing.

Based on the simple modification of the code, the algorithm can be compared to classical simulated annealing-based line balancing; this comparison demonstrates that the main benefit of the proposed constrained handling is the acceleration of the optimisation. At the same time, the application of the inner-loop-based assignment significantly reduces the variance and increases the chance of obtaining improved line balancing results.



**Figure 8.** Boxplot of time, skill and equipment-related objectives for different independent runs of the algorithm and with different numbers of operators.



**Figure 9.** Comparison of the operators’ activity times.

### 5. Conclusions

We proposed an assembly line balancing algorithm to improve the efficiency of production systems by the multiobjective assignment of tasks to operators. The optimisation of this assignment is based on a multilayer network model that provides information about the activity times, constraints and benefits (objectives) of the assignments, where the layers of the network represent the skills of the operators, the tools required for their activities and the precedence constraints of their activities.

The training and equipment costs as well as the precedence of the activities are also taken into account in the activity–operator layer of the network. As these costs and benefits are difficult to evaluate, the analytic hierarchy process (AHP) technique is used to quantify the importance of the criteria. The optimisation problem is solved by a multi-level simulated annealing algorithm (SA) that efficiently handles the precedence constraints thanks to the proposed problem-specific representation.

The proposed algorithm was implemented in MATLAB and the applicability of the method demonstrated with an industrial case study of wire harness manufacturing. The results confirm that multilayer network-based representations of optimisation problems in manufacturing seem to be potential promising solutions in the future.

The main contribution of the work is that it presents tools that can be used for the efficient representation of expert knowledge that should be utilised in complex production management

problems. The proposed multilayer network-based representation of the production line supports the incorporation of advanced (ontology-based) models of production systems and provides an interpretable and flexible representation of all the objectives of the line balancing problem.

The AHP-based pairwise comparison of the importance of the nodes, edges and complex paths of this network can be used to evaluate the objectives of the optimisation problems. The integration of the network-based knowledge representation and the AHP-based knowledge extraction makes the application of the proposed methodology attractive in complex optimisation problems.

**Author Contributions:** L.N., T.R. and J.A. developed the methodology and software; L.N. prepared the original draft; J.A. reviewed and edited the manuscript. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported from the Higher Educational Institutional Excellence Program 2019 the grant of the Hungarian Ministry for Innovation and Technology (Grant Number: NKFIH-1158-6/2019). Tamás Ruppert was supported by the project 2018-1.3.1-VKE-2018-00048–Development of intelligent Industry 4.0 solutions of production optimization in existing plants and the ÚNKP-19-3 New National Excellence Program of the Ministry of Human Capacities.

**Acknowledgments:** The authors are grateful for the valuable comments and suggestions offered by the anonymous reviewers.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

SA	Simulated annealing
ALB	Assembly line balancing
$G_{i,j}$	Bipartite graphs between the $i$ th and $j$ th sets of objects
$O_i, O_j$	General representation of a set of objects as $O_i, O_j \in \{s, e, a', a, w\}$
$\mathbf{a} = a_1, \dots, a_{N_a}$	Index of activities
$\mathbf{o} = o_1, \dots, o_{N_o}$	Index of operators
$\mathbf{s} = s_1, \dots, s_{N_s}$	Index of skills
$\mathbf{e} = e_1, \dots, e_{N_e}$	Index of equipment
$\mathbf{w} = w_1, \dots, w_{N_w}$	Index of workstations
<b>W</b>	Workstation assigned for the activity, $N_a \times N_w$
<b>O</b>	Operators assigned for the activity, $N_a \times N_o$
<b>S</b>	Skills assigned for the activity, $N_a \times N_s$
<b>E</b>	Equipment assigned for the activity, $N_a \times N_e$
<b>A'</b>	Precedence constraint between activities, $N_a \times N_a$
<b>T</b> = $t_1, \dots, t_{N_a}$	Activity time
$c_1$	Station-time-related cost
$c_2$	Skill-related (training) cost
$c_3$	Equipment-related cost
$T_c$	Cycle time
$N_w$	Number of workstations
$N_o$	Number of operators
$N_s$	Number of skills
$N_e$	Number of pieces of equipment
$N_\pi$	Number of sequence elements

## References

1. Becker, C.; Scholl, A. A survey on problems and methods in generalized assembly line balancing. *Eur. J. Oper. Res.* **2006**, *168*, 694–715. [[CrossRef](#)]
2. Boysen, N.; Fliedner, M.; Scholl, A. Assembly line balancing: Which model to use when? *Int. J. Prod. Econ.* **2008**, *111*, 509–528. [[CrossRef](#)]

3. Nilakantan, J.M.; Ponnambalam, S.; Nielsen, P. Application of Particle Swarm Optimization to Solve Robotic Assembly Line Balancing Problems. In *Handbook of Neural Computation*; Elsevier: Amsterdam, The Netherlands, 2017; pp. 239–267.
4. Hazır, Ö.; Delorme, X.; Dolgui, A. A survey on cost and profit oriented assembly line balancing. *IFAC Proc. Vol.* **2014**, *47*, 6159–6167. [[CrossRef](#)]
5. Rüßmann, M.; Lorenz, M.; Gerbert, P.; Waldner, M.; Justus, J.; Engel, P.; Harnisch, M. *Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries*; Technical Report 9; Boston Consulting Group: Boston, MA, USA, 2015.
6. Gola, A. Reliability analysis of reconfigurable manufacturing system structures using computer simulation methods. *Eksplot. Niezawodn.* **2019**, *21*, 90–102. [[CrossRef](#)]
7. Honti, G.M.; Abonyi, J. A review of semantic sensor technologies in internet of things architectures. *Complexity* **2019**, 2019. [[CrossRef](#)]
8. Boccaletti, S.; Bianconi, G.; Criado, R.; Del Genio, C.I.; Gómez-Gardenes, J.; Romance, M.; Sendina-Nadal, I.; Wang, Z.; Zanin, M. The structure and dynamics of multilayer networks. *Phys. Rep.* **2014**, *544*, 1–122. [[CrossRef](#)]
9. Kivelä, M.; Arenas, A.; Barthelemy, M.; Gleeson, J.P.; Moreno, Y.; Porter, M.A. Multilayer networks. *J. Complex Netw.* **2014**, *2*, 203–271. [[CrossRef](#)]
10. Ruppert, T.; Honti, G.; Abonyi, J. Multilayer network-based production flow analysis. *Complexity* **2018**, 2018, 1–15. [[CrossRef](#)]
11. Pigler, C.; Fogarassy-Vathy, Á.; Abonyi, J. Scalable co-Clustering using a Crossing Minimization—Application to Production Flow Analysis. *Acta Polytech. Hung.* **2016**, *13*, 209–228.
12. Romeijn, H.E.; Smith, R.L. Simulated annealing for constrained global optimization. *J. Glob. Optim.* **1994**, *5*, 101–126. [[CrossRef](#)]
13. Suresh, G.; Sahu, S. Stochastic assembly line balancing using simulated annealing. *Int. J. Prod. Res.* **1994**, *32*, 1801–1810. [[CrossRef](#)]
14. Chiang, W.C.; Urban, T.L. The stochastic U-line balancing problem: A heuristic procedure. *Eur. J. Oper. Res.* **2006**, *175*, 1767–1781. [[CrossRef](#)]
15. Ağpak, K.; Gökçen, H. A chance-constrained approach to stochastic line balancing problem. *Eur. J. Oper. Res.* **2007**, *180*, 1098–1115. [[CrossRef](#)]
16. Guerriero, F.; Miltenburg, J. The stochastic U-line balancing problem. *Nav. Res. Logist. (NRL)* **2003**, *50*, 31–57. [[CrossRef](#)]
17. Chiang, W.C. The application of a tabu search metaheuristic to the assembly line balancing problem. *Ann. Oper. Res.* **1998**, *77*, 209–227. [[CrossRef](#)]
18. Tsujimura, Y.; Gen, M.; Kubota, E. Solving fuzzy assembly-line balancing problem with genetic algorithms. *Comput. Ind. Eng.* **1995**, *29*, 543–547. [[CrossRef](#)]
19. Özcan, U.; Toklu, B. Multiple-criteria decision-making in two-sided assembly line balancing: A goal programming and a fuzzy goal programming models. *Comput. Oper. Res.* **2009**, *36*, 1955–1965. [[CrossRef](#)]
20. Baykasoglu, A. Multi-rule multi-objective simulated annealing algorithm for straight and U type assembly line balancing problems. *J. Intell. Manuf.* **2006**, *17*, 217–232. [[CrossRef](#)]
21. McMullen, P.R.; Frazier, G. Using simulated annealing to solve a multiobjective assembly line balancing problem with parallel workstations. *Int. J. Prod. Res.* **1998**, *36*, 2717–2741. [[CrossRef](#)]
22. Güden, H.; Meral, S. An adaptive simulated annealing algorithm-based approach for assembly line balancing and a real-life case study. *Int. J. Adv. Manuf. Technol.* **2016**, *84*, 1539–1559. [[CrossRef](#)]
23. de Jong, P. A statistical approach to Saaty’s scaling method for priorities. *J. Math. Psychol.* **1984**, *28*, 467–478. [[CrossRef](#)]
24. Ho, W.; Ma, X. The state-of-the-art integrations and applications of the analytic hierarchy process. *Eur. J. Oper. Res.* **2018**, *267*, 399–414. [[CrossRef](#)]
25. Saaty, T.L. Decision making—The analytic hierarchy and network processes (AHP/ANP). *J. Syst. Sci. Syst. Eng.* **2004**, *13*, 1–35. [[CrossRef](#)]
26. Stoma, P.; Stoma, M.; Dudziak, A.; Caban, J. Bootstrap Analysis of the Production Processes Capability Assessment. *Appl. Sci.* **2019**, *9*, 5360. [[CrossRef](#)]
27. Che, Z. Clustering and selecting suppliers based on simulated annealing algorithms. *Comput. Math. Appl.* **2012**, *63*, 228–238. [[CrossRef](#)]

28. Coulter, E.D.; Sessions, J.; Wing, M.G. Scheduling forest road maintenance using the analytic hierarchy process and heuristics. *Silva Fenn.* **2006**, *40*, 143–160. [[CrossRef](#)]
29. Lu, Y.; Morris, K.C.; Frechette, S. Current standards landscape for smart manufacturing systems. *Natl. Inst. Stand. Technol. NISTIR* **2016**, *8107*, 39.
30. Bryton, B. Balancing of a Continuous Production Line. Ph.D. Thesis, Northwestern University, Evanston, IL, USA, 1954.
31. Erel, E.; Sarin, S.C. A survey of the assembly line balancing procedures. *Prod. Plan. Control* **1998**, *9*, 414–434. [[CrossRef](#)]
32. Leitold, D.; Vathy-Fogarassy, A.; Abonyi, J. Empirical working time distribution-based line balancing with integrated simulated annealing and dynamic programming. *Cent. Eur. J. Oper. Res.* **2019**, *27*, 455–473. [[CrossRef](#)]
33. Hackman, S.T.; Magazine, M.J.; Wee, T. Fast, effective algorithms for simple assembly line balancing problems. *Oper. Res.* **1989**, *37*, 916–924. [[CrossRef](#)]
34. Scholl, A.; Voß, S. Simple assembly line balancing—Heuristic approaches. *J. Heuristics* **1997**, *2*, 217–244. [[CrossRef](#)]
35. Hoffmann, T.R. Assembly line balancing with a precedence matrix. *Manag. Sci.* **1963**, *9*, 551–562. [[CrossRef](#)]
36. Sacerdoti, E.D. *A Structure for Plans and Behavior*; Technical Report; SRI International's Artificial Intelligence Center: Menlo Park, CA, USA, 1975.
37. Jackson, J.R. A computing procedure for a line balancing problem. *Manag. Sci.* **1956**, *2*, 261–271. [[CrossRef](#)]
38. Ruppert, T.; Abonyi, J. Software sensor for activity-time monitoring and fault detection in production lines. *Sensors* **2018**, *18*, 2346. [[CrossRef](#)] [[PubMed](#)]
39. Ong, N.; Boothroyd, G. Assembly times for electrical connections and wire harnesses. *Int. J. Adv. Manuf. Technol.* **1991**, *6*, 155–179. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).