Journal of Applied Research on Industrial Engineering

www.journal-aprie.com

J. Appl. Res. Ind. Eng. Vol. x, No. x (xx) x–x.

Paper Type: Original Article

Applying a Multi-Objective Particle Swarm Optimization Algorithm For Sequencing And Balancing a Mixed-Model Assembly Line Problem With Setup Times Between Tasks

Fahimeh, Tanhaie [1]

[1] Assistant Professor, Industrial Engineering Department, Faculty of basic science and Engineering, Kosar university of Bojnord ; fahimeh.tanhaie@kub.ac.ir

Citation:

Received: ----- *Revised:* ----- *Accepted:*---	LastName, Initial First Name., & LastName, Initial First Name . (Date). Paper Title. Journal of applied research on industrial engineering , Volume (Issue), PP.

Abstract

Mixed-model assembly is a particular set of production lines assembling a family of product models with similar specifications. Designing paced assembly lines faces two primary problems: balancing and sequencing. The balancing quality is closely associated with the described production sequence. Although these two are problems of one assembly method, they do not occur simultaneously; balancing poses a problem during the line designing, whereas sequencing becomes problematic at the fluctuating demand of markets. The present research presents a balancing and sequencing problem and the proper times to set up the machines between tasks. Unlike a majority of published studies, this paper contains two successive tasks' setup times in dynamic periods, in which periods also impact the flowing period. A mathematical is described with a number of objective functions, reducing the inappropriate assembly line sequence, reducing setup cost, and reducing the inappropriate product balance and the impact of this situation on incomplete tasks. Thus, the literature has presented several metaheuristic algorithms to solve the problems nearly optimally. This study uses a multi-objective particle swarm optimization algorithm, a suitable approach, to create models and solutions. Various problems are designed in different sizes and compared, and the decision variable sensitivity is investigated to prepare managerial intuitions. The findings propose that the presented algorithm can solve the research problems more efficiently.

Keywords: balancing, sequencing, setup time, MOAs.

Corresponding Author: Fahimeh Tanhaie

1| Introduction

An assembly line, a mass production process, includes a continuous movement of units through the sequences or workstations. A variety of products determine the various groups of assembly lines, such as the mixed-model assembly line (MMAL), which is applied in an assembly line [1]. In this type, the similarity among the product models is high enough such that the setup times between successive stations are negligible from one product model to another [2], thus simultaneously assembling different models to the line per sequence. This method needs no extra setup times and answers the diverse needs of customers. Therefore, this increases line flexibility, which plays a vital role in the system's promotion. Increased flexibility is considerably important in the highly competitive industrial production environment to fulfil customers' requests [3]. Two primary problems took place in the MMAL, disregarding the line shape. Line balancing, the first problem, alludes to the practical allocating of tasks to sequences so that some function calculations are optimized to match the priority relations between the tasks. The aim of model sequencing, the second problem, is to select the product arrangement and move it into the assembly line, where various models will be manufactured [4].

In the present study, we have weighed the two notions; thus, we identify the product sequence and line balancing. The paper simultaneously studies some objectives, such as reducing the inappropriate assembly line sequence, the entire setup cost, the inappropriate product balance and its impacts on incomplete works. We tackle the combination of product mix sequencing and assembly line balancing. Both problems suggest special time scaling, as assembly line planning is a multiple months to years decision, whereas sequencing might be specified daily or weekly. To acquire the Pareto solution, an ameliorated algorithm, multi-objective particle swarm optimization (MOPSO), is proposed in the MMALs' simultaneous sequencing and balancing, and subsequently, a comparison is performed between the presented model and two multi-objective algorithms (MOAs). The suggested algorithm presents a new solution demonstration method, which can efficiently be applied to the concurrent sequencing and balancing MMALs. As far as the authors are aware, no research has been conducted to solve the problem of the MMAL balancing and sequencing regarding the suggested algorithm and setup times between the products. This paper provides the first effort to bridge this research gap. The next sections of the study are arranged as follows. The related literature is defined in section 2, illustrating the problem and explaining a mathematical model presented in section 3. The problem-solving and numerical instances are allocated to section 4; the last part, the conclusion, is in section 5.

2| Literature review

The MMAL-related literature is categorized into three domains: some investigations study different models sequencing in the assembly lines. Researchers have regarded the problems of the assembly line sequencing by MMAL sequencing. The studies primarily specified an appropriate sequence for various models to maximize line application. Kinable, Cire, and van Hoeve [5] optimized the sequencing problems through a new method by evaluating the task location in the order and then determining the setup times between the tasks. A mathematical model was proposed by Nazar and Pillai [6], finding sequences able to minimize the zero idle time constraints, satisfying the capacity for the machine and production rate variation. Defersha and Mohebalizadehgashti [7] regarded a mixed-integer linear programming mathematical model using the genetic algorithm for sequencing and balancing problems concurrently. To specify the optimum product sequence to magnify client satisfaction, Rabbani, Heidari, and Farrokhi-Asl [8] suggested two MOAs. Several research articles have noted the sequencing problem of MMAL regarding a make-to-order environment. For example, Minner and Öner-Közen [9] have designed a precedence sequencing problem, where customers can specify the time extent needed for accomplishing their needs, as the decision process of Markov. The fuzzy approach application has also been considered in papers about sequencing problems, including manufacturing systems [10], supplier selection [11][12], and Decision-Making problems [13].

Several studies, such as Boysen and Betts, have regarded the line's balancing and design and several objectives, including cycle time. [14][15]. Salveson was the first scientist to model a single-model assembly line balancing problem (SMALBP) [16]. Subsequently, researchers studied the SMALBP and proposed multiple-solving approaches and mathematical models[17]. For instance, Fathi et al. reduced the workstation number in the SMALBP for U-shape and straight lines, compared some heuristics to discover solutions, and presented a comparative assessment[18]. It is improper to provide every model with an assembly line; therefore, manufacturers assemble a product set on mix-model assembly lines [19]. Akpınar et al. presented an accurate solution algorithm for a balancing problem of the setup assembly line according to the Benders decomposition algorithm. They compared his result and a mixed-integer linear technique [20]. Baykasoglu and Akpınar proposed a mixed-integer linear type programming model considering sequence-dependent setup times and zoning limitations [21]. Fleszar studied the MMAL balancing problem and proposed mixed-integer linear programming for it, in which each workstation has a limited approachability window, and merely a part of a workpiece is accessible [22].

Other published studies have regarded product model sequencing and line balancing together [23][24][25][26] to reduce favourable objective performances. Logically, we can assign each class to an MMAL problem. Basically, considering the problems of sequencing and balancing, the best approach is to consider them together. In this way, the line planner regards both problems simultaneously, which is more realistic. Thus, we prefer to regard both aspects by studying the articles that have tried to solve both problems. Thomopoulos presents a process where a single-model line balancing approach is adapted to mixed-model plans. He suggested a sequencing technique to specify the arrangement where models should move down the line [27]. Cao and Ma proposed a model for sequencing and balancing problems. They developed the related mathematical formula regarding two feasibly main objectives: reducing the entire over and idle time and maintaining a fixed rate of part application. For sequencing MMAL, they used a multi-objective genetic algorithm [28]. Merengo et al. proposed novel balancing and production sequencing methodologies that seek some usual goals: reducing the imperfect job rate or the possibility of blocking/starvation incidences and minimizing jobs in the process [29]. A researcher, Miltenburg, investigated mixed-model lines in time systems only, designed the joint problem, and then proposed a solution algorithm for practical size examples [30].

Keun Kim et al. described a novel evolutionary technique. Their goal was to tackle sequencing and balancing problems in MMAL. They proposed a novel genetic procedure, an endosymbiotic evolutionary algorithm, to untangle both problems [31]. Mendes et al. suggested a simulated annealing meta-heuristic to obtain line configurations. This configuration would have the least workstations and a smooth workload balance among the workstations [32]. Van Hop explained a fuzzy heuristic to unravel the problem elicited from the mixed precedence constraints and aggregating fuzzy numbers. The approach was generally planned to use a varying-section exchange approach and sequence the jobs in order. Afterwards, according to the fuzzy numbers, tasks were assigned to workstations considering technological constraints and the extent of cycle time [33]. Manavizadeh et al. [34] regarded three purposes: reducing cycle times, overloading work and wastages, and sequencing and balancing problems. Kucukkoc and Zhang [35] investigated the sequencing and balancing problems in the parallel assembly line, which has multi-line stations, making the task allocation more complicated and pliable, and proposed a heuristic solution technique for the problem. To reduce the station numbers, Hamzadayi and Yildiz [36] investigated the Type-I assembly line sequencing and balancing problem, which was different from most related research having fixed stations. Lian et al. [37] added the independent country to ameliorate the colonial competitive algorithm (CCA); the improved CCA exhibits great problem-solving capability. Faccio, Gamberi, and Bortolini [38] described a hierarchical sequencing and balancing system for paced MMALs to hamper excess work. The technique applies "jolly operators" to supplement a flexible workforce, balancing the overload of workstations.

Liao et al. developed an unpaced synchronous transfer conception in a mixed-model two-sided assembly line balancing and sequencing problem. An improved genetic algorithm with suitable solution representation is proposed to solve the problem. The findings show that the presented algorithm can solve the research problems more efficiently than other algorithms [27]. Meng et al. considered preventive maintenance scenarios in robust mixed-model assembly line balancing and sequencing problems to minimize the makespan. They solved and compared various problems of different sizes with a robust mathematical model and a multi-objective cooperative differential evolution algorithm. The results show the efficiency of the proposed model [39]. A disassembly line balancing and sequencing problem was modelled by Edis et al. to maximize final profits and minimize processing times. They developed a mixed-integer programming model with valid inequalities to meet an optimal sequence of tasks and optimal assignment of tasks to workstations. The findings show that the approach reaches a near-optimal solution with small optimality gaps [40]. To quantify the model, Zeng et al. considered a sequencing and balancing problem in a multi-objective robotic disassembly line. The model was solved with a proposed genetic simulated annealing algorithm to minimize the number of tool replacements, maximize total profit, and minimize energy consumption in the disassembly line. The proposed algorithm was compared with some metaheuristics approaches, and the results show the superiority of the presented approach [41].

Chawasemerwa et al. developed a model for a short period to increase flexibility in time scheduling. They considered a flexible working environment and improved system management quality and customer satisfaction [42]. Shahvaroughi Farahani et al. predicted stock price trends by considering some influencing factors. The model analyzed the relationship between some indicators to enhance stock price prediction. They solved the model with a Neural Network that operated better in stock price prediction accuracy than other algorithms [43]. Ejlali et al. developed a multi-objective model for a three-level relief cycle logistics in an uncertain environment. Some factors, such as vehicle routing and inventory transfer, were considered in the presented article. The particle swarm optimization algorithm was applied to solve the model, which decreased the time to reach an optimal solution [44].

Overall, the literature reviews show significant attention to solving the sequencing and balancing problems in MMAL with different solution algorithms. The present research presents a balancing and sequencing problem and the proper times to set up the machines between tasks. To acquire the Pareto solution, an ameliorated algorithm, multi-objective particle swarm optimization, is proposed, and subsequently, a comparison is performed between the presented model and two multi-objective algorithms. The suggested algorithm presents a new solution-demonstration method, which can be applied to the concurrent sequencing and balancing MMALs efficiently. Table 1 summarizes some notable papers in the literature review by considering the problem type, objectives, and setup time to show the contribution of the research.

.

Table 1. An overview of approaches to MMAL problems in the literature

Publication	Problems	Objectives	Setup Time	
			YES	NO
Kinable, Cire, and van Hoeve [5]	Sequencing	Reducing the entire over and idle time	*	
Mendes et al. [33]	Sequencing and balancing	Smooth workload balance among the workstations		*
Manavizadeh et al. [35]	Sequencing and balancing	Reduction of cycle times, the overload of work, and wastages		*
Akpınar et al. [21]	balancing	Reduced number of workstations	*	
Defersha and Mohebalizadeh gashti [7]	Sequencing and balancing	Magnifying client satisfaction		*
Fathi et al. [19]	balancing	Reduced number of workstations		*
Hamzadayi and Yildiz [37]	Sequencing and balancing	reduce the station numbers		*
Nazar and Pillai [6]	sequencing	minimize the zero idle time constraints		*
Merengo et al. [30]	Sequencing and balancing	reducing the imperfect jobs and minimizing jobs in the process		*
Baykasoglu and Akpınar [22]	balancing	Minimizing setup times	*	
Thomopoulos [28]	Sequencing and balancing	Reducing the entire over and idle time and maintaining a fixed rate of part application		*

Meng et al. [41]	Sequencing and balancing	Minimize the makespan		*
The proposed model	Sequencing and balancing	Reducing the inappropriate assembly lines sequence, reducing setup cost, and reducing the inappropriate product balance	*	

As far as the authors are aware, no research has been conducted to solve the problem of the MMAL balancing and sequencing regarding the suggested algorithm and setup times between the products. This paper provides the first effort to bridge this research gap.

3| Problem definition and mathematical model

We tackle the combination of product mix sequencing and assembly line balancing. Both problems suggest special time scaling, as the assembly line planning is a multiple months to years decision, whereas the sequencing might be specified daily or weekly. To acquire the Pareto solution, an ameliorated algorithm, multi-objective particle swarm optimization (MOPSO), is proposed in the MMALs' simultaneous sequencing and balancing, and subsequently, a comparison is performed between the presented model and two multi-objective algorithms (MOAs). The suggested algorithm presents a new solution-demonstration method, which can be applied to the concurrent sequencing and balancing MMALs efficiently. Unlike the majority of published studies, this paper contains two successive tasks' setup times in dynamic periods, in which periods also impact the flowing period.

Various identical models are available in each period, and it is better to allocate to their serial places; thus, each will be produced as per their serial relevance. Various tasks are actually available, and it is better to allocate them to workstations based on the serial position, depending on the workstation. Such a situation needs a setup time to be defined between a number of tasks they are performing inside a workstation successively and without pause. This paper proposes the problem assumptions, describes parameters and variables, and suggests a mathematical model. The assumptions of the problem are mentioned here:

- Associating a model type with each cycle time.
- Supposing a total precedence diagram for models.
- Supposing the setup times between models as negligible.
- A fixed-rate releasing of the products to the conveyor.
- Ignoring operator walks time.
- Knowing the number of workstations.
- Assuming that the stations are all closed.
- Knowing each product's task performance times.
- Considering work utility, customer cost, and operator's idle time cost.

3.1| Model formulation

The first step's applied notation for the problem formulation is as follows:

M	Models' number
$m, n, k \in \{1, 2, ..., M\}$	Models' index

T_m	Model m's number of tasks
$t \in \{1, 2, .., T_m\}$	Tasks' index
C_m	Model m's required cycle time
J	Stations' number
$j \in \{1, 2, .., J\}$	Stations' index
$I = \sum_{m=1}^{M} d_m$	Total products number
$i \in \{1, 2, .., I\}$	Position Index in total products
L_j	The station j's line length
d_m	Model m's demand
t_{mj}	Model m's completion time at station j
v_c	Conveyor speed
pre_t	Task j's set of immediate predecessors
γ	Products' launch interval to the line
t_{tmj}	Processing time of model m's task at workstation j
Z_{imn}	if ith and i + 1th products in a sequence are model m and n
y_{im}	if ith product in a sequence is model m
S_{jmn}	Setup cost at station j for models m and n
x_{tmj}	if task i of model j is assigned to station k
C_{Idl}	Idle operators' time cost (per unit time)
C_U	Utility customer work cost (per unit time)
Z_{ij}	starting position of product i at station j
U_{ij}	Utility worker for product i in a sequence at station j

Decision variables

$$Z_{imn} = \begin{cases} 1 & \text{if ith and i + 1th products in a sequence are model m and n} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{im} = \begin{cases} 1 & \text{if ith product in a sequence is model m} \\ 0 & \text{otherwise} \end{cases}$$

$$x_{tmj} = \begin{cases} 1 & \text{if task i of model j is assigned to station k} \\ 0 & \text{otherwise} \end{cases}$$

Multi-objective modelling

Here, we propose a mathematical model for the described problem:

$$z1 = \min\left(\sum_{i=1}^{I}\sum_{j=1}^{J}\left(C_u * U_{ij}\right) + \sum_{j=1}^{J}\sum_{i=1}^{I}\sum_{m=1}^{M}\sum_{n=1}^{M}\left(Z_{imn} * \mathrm{s}_{jmn}\right)\right)$$
$$z2 = \min\left\{\max\left\{0, \sum_{t=1}^{T_m}\sum_{m=1}^{M}\sum_{j=1}^{J}\left(C_{Idl} * \left(\mathrm{t}_{tmj} * \mathrm{x}_{tmj} - C_m\right)\right)\right\}\right\} \tag{1}$$

$$\sum_{i=1}^{I} y_{im} = d_m \qquad \forall m \tag{2}$$

$$\sum_{m=1}^{M} y_{im} = 1 \qquad \forall i \tag{3}$$

$$\sum_{j=1}^{J} x_{tmj} = 1 \qquad \forall m, t = 1, 2, .., \mathrm{T}_m \tag{4}$$

$$\sum_{m=1}^{M} z_{imn} = \sum_{k=1}^{M} z_{i+1nk} \qquad i = 1, 2, ..., I-1, \forall n \tag{5}$$

$$\sum_{m=1}^{M} z_{imn} = \sum_{k=1}^{M} z_{1nk} \qquad \forall n \tag{6}$$

$$\sum_{j=1}^{J} j * x_{hmj} \le \sum_{j=1}^{J} j * x_{tmj} \qquad \forall m, t = 1, 2, .., \mathrm{T}_m, h \in pre_t \tag{7}$$

$$Z_{1j+1} = \sum_{l=1}^{j} L_l \qquad j = 1, 2, ..., J-1 \tag{8}$$

$$Z_{i+1j} = Z_{ij} + v_c * \left(\sum_{m=1}^{M} y_{im} * t_{mj} - U_{ij} - \gamma\right) \qquad i = 1, 2, .., I-1, j = 1, 2, ..., J \tag{9}$$

$$\sum_{m=1}^{M}\sum_{n=1}^{M} z_{imn} = 1 \qquad \forall i \tag{10}$$

$$\sum_{i=1}^{M}\sum_{n=1}^{M} z_{imn} = d_m \qquad \forall m \tag{11}$$

$$U_{ij} \geq \left(Z_{ij} + v_c * \sum_{m=1}^{M} y_{im} * t_{mj} - \sum_{l=1}^{j} L_l \right) \Big/ v_c \qquad i = 1, 2, .., I-1,\ j = 1, 2, ..., J \tag{12}$$

$$U_{Ij} \geq \left(Z_{Ij} + v_c * \sum_{m=1}^{M} y_{Im} * t_{mj} - \left(\sum_{l=1}^{j-1} L_l + v_c * \gamma \right) \right) \Big/ v_c \qquad j = 1, 2, ..., J \tag{13}$$

$$y_{im} \in \{0,1\} \qquad \forall i, m \tag{14}$$

$$x_{tmj} \in \{0,1\} \qquad \forall m, j, t = 1, 2, .., \mathrm{T}_m \tag{15}$$

$$z_{imn} \in \{0,1\} \qquad \forall i, \mathrm{m}, \mathrm{n} \tag{16}$$

The sequence and balance can induce incomplete and independent tasks in each approach. The first sentence of the first function reduces the inappropriate assembly line sequence and its impact on unfinished tasks. The second term reduces the sum of setup costs. The second objective function decreases the inappropriate products' balance and its effects on incomplete tasks. The satisfaction of each model's request is guaranteed by Constraint (2). Constraint (3) indicates that precisely one product can be allocated for each position of each sequence, and Constraint (4) implies one station allocated to each task.

Constraints (5-6) guarantee maintaining the product sequence from one cycle to another. Constraint set (7) is the preferred constraint, considering the priority relation among each model's tasks. Constraint (8) imposes that the action for every cycle's first product must begin at the station's left border. Constraint (9) describes the worker's beginning position for the product i +1 in a sequence. Constraint (10), a group of position constraints, explains that precisely one product should fill every position in each sequence. Constraint (11) enforces that all the requests regarding MPS as the limitation should be answered. Constraint (12) shows the efficient time of the worker for product i in a sequence at station j, and constraint (13) shows the efficient time of the worker for end product i in a sequence at this station. Finally, Constraints (14-16) describe all binary decision variables.

4| Problem unravelling and numerical examples

Regarding the MMAL sequencing and balancing problems as NP-hard, more complicated models might be considered NP-hard [41], and subsequently, researchers have used MOAs to unravel the problem.

The multi-objective problem (MOP) usually has inconsistent objectives in various calculation units. Thus, no single solution is accessible to ameliorate each objective concurrently [45]. An MOPSO algorithm is applied to unravel this MOP.

4.1| MOPSO Algorithm

The MOPSO algorithm is an important amelioration approach for multi-objective optimization problems. This approach characteristically provides robust and measurable solutions. While moving along the search space, the swarm, the population of particles, has been set to the start point with accidental positions and velocities. Generations have been updated to discover optimal solutions by applying a swarm. An operator containing social and local elements to obtain the velocity of motions. MOPSO can use the Pareto dominance relation to assess the solutions' advantages and choose the foremost leaders, afterwards kept in an external restricted archive containing top non-dominated solutions accomplished to date. The calculation is as follows:

$$\vec{y}(t+1)=\vec{y}(t)+\vec{v}(t+1) \tag{17}$$

$$\vec{v}(t+1)=\omega\cdot\vec{v}(t)+(C_1\cdot b_1)\cdot(\vec{p}_{best}(t)-\vec{y}(t))+(C_2\cdot b_2)\cdot(\overrightarrow{rep}_h(t)-\vec{x}(t)) \tag{18}$$

where $\vec{y}(t)$ and $\vec{v}(t)$ are, respectively, the present particle i's position and velocity.

b1 and b2 define uniform random numbers between [0, 1], specifying the finest location for the particle in each repeat. C_1 and C_2 stand for personal and global learning coefficients. ω is the inertia weight that controls the former velocity's impact on the present velocity. $\vec{p}_{best}(t)$ shows the particle i's best experience. $\overrightarrow{rep}_h(t)$ defines the finest particle put forward from the repository given by a global neighbourhood.

Fig. 1 shows the conformation of the basic MOPSO.

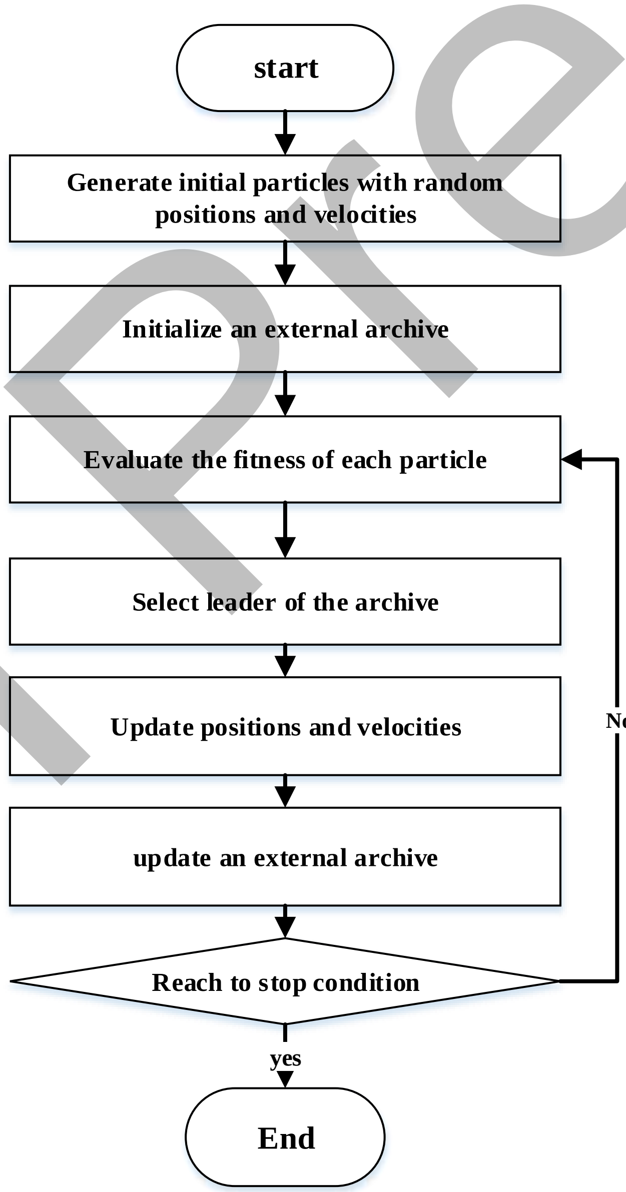

Fig. 1. MOPSO conformation

$$b_1 = b_2, \quad b_1 + b_2 > 4 \tag{19}$$

$$b = b_1 + b_2 \tag{20}$$

$$\omega = \frac{2}{b-2} + \sqrt{b^2 - 2b} \tag{21}$$

$$C_1 = \omega b_1 \tag{22}$$

$$C_2 = \omega b_2 \tag{23}$$

$$\vec{y}(\mathrm{t+1}) = \vec{y}(\mathrm{t}) + \vec{v}(\mathrm{t+1}) \tag{24}$$

$$\vec{v}(\mathrm{t+1}) = \omega \cdot \vec{v}(\mathrm{t}) + (\mathrm{C}_1 \cdot b_1) \cdot (\vec{\mathrm{p}}_{best}(\mathrm{t}) - \vec{y}(\mathrm{t})) + (\mathrm{C}_2 \cdot b_2) \cdot (\overrightarrow{\mathrm{rep}}_h(\mathrm{t}) - \vec{y}(\mathrm{t})) \tag{25}$$

$$\vec{y}(\mathrm{t}) = \min(\max(\vec{y}(\mathrm{t}), \vec{y}(\mathrm{t})_{\min}), \vec{y}(\mathrm{t})_{\max}) \tag{26}$$

$$\vec{v}(\mathrm{t}) = \min(\max(\vec{v}(\mathrm{t}) - \vec{v}(\mathrm{t})_{\max}), \vec{v}(\mathrm{t})_{\max}) \tag{27}$$

where $\vec{y}(\mathrm{t})_{\max}$ and $\vec{y}(\mathrm{t})_{\min}$ do not permit to leave the practical solution, and $\vec{v}(\mathrm{t})_{\max}$ is applied to restrict the velocity.

4.2| **Initialization**

As the primary population, a presentation of practical solutions that define problem-specific features must be proposed in the first step. The present study suggests MOPSO in which each proposed solution includes three parts. The first part shows the task sequence, the second part shows the task allocation in the assembly line, and the third part describes the model sequence. The solution is represented below.

The task sequence: the string representation is largely applied for MMAL sequencing problems. We require a technique that modifies the continuous space to a discrete one to apply a PSO algorithm fundamentally for continuous space. Therefore, we select a solution representation equal to the total tasks in the problem and allocate a range of [0, 1] numbers randomly to each element. Afterwards, the tasks are arranged from small to large based on the corresponding number's value in the solution.

Task allocation in the assembly line: the task allocation vector has J elements, where the jth element defines the number of tasks allocated to station j.

Model sequencing: every component of a model sequence vector correlates with a model. The model sequence means the elements' permutation in this vector.

An example with 16 tasks, 4 stations, 3 models, and MPS = ABCC is depicted in Fig. 2.

Random numbers	0.22	0.43	0.74	0.69	0.21	0.93	0.35	0.48	0.18	0.05	0.78	0.46	0.36	0.62	0.83	0.29
Tasks sequencing	4	8	13	12	3	16	6	10	2	1	14	9	7	11	15	5
Task assignment	4				3			4				5				
Model sequencing	A				B			C				C				

Fig. 1. Solution representation

4.3| **Computational results**

The trials of this study are coded in MATLAB R2013b and run on Intel CORE i7 2.30 GHz on a private computer with 6 GB RAM. oded in MATLAB 13.0 and run on a private computer with Intel (R) Core (TM) 2 Duo CPU T9550 @ 2.67GHz and 4GB of memory.

4.3.1 | Parameter Setting

The finest choice for factors and the parameters set plainly escalates the study procedure. Subsequently, the finest tuning value (Table 2) for a parameter's set was chosen by performing vast trials before running MOA techniques, including NSGA-II, multi-objective, MOPSO, and colony optimization (MOACO).

Table 2. Parameter setting of each algorithm

Parameter setting	**NSGA-II**	**MOACO**
Population size	310	-
Number of generations	40	-
Cross-over (Probability)	0.83	-
Mutation (Probability)	0.17	-
Number of ants	-	115
Pheromone evaporation coefficient ϱ	-	0.04

In this study, the presented MOPSO has applied the Taguchi approach to choose the finest parameters. The Minitab software was used to design the tests, and the outcomes of each parameter were inspected in four levels. The multi-objective plan improvement might be simplified through the Taguchi approach, which might provide extra programming pliability for a number of usages. Taguchi may reduce the number of tests [46]; this might be affirmed by concerning two main implements of the signal-to-noise ratio (S/N) and the orthogonal array (OA), as shown in Eq. (28) [47].

$$\left(\frac{S}{N}\right)\text{ratio} = 10 \times \log\left(\frac{\bar{y}^2}{s^2}\right) \tag{28}$$

where $\bar{y}$ shows the average response and s^2 indicates the response variance for the sample. Then, the analysis of data was done using a graphical Taguchi method. Each level's mean S/N ratios are shown in Fig. 3.

Fig. 3. The mean S/N ratio plot

According to Fig. 3, the average S/N ratios analysis might assist in selecting the proper parameter amounts with superior outcomes. The best level for the factors becomes 1, 3, 2, and 4, respectively.

4.3.2 |. Small-Sized Problems

In this part, a test is performed by applying a number of small test problems, which are self-made. Tables 3 and 4 represent MPS and other input data related to small test problems.

Table 3. Small-size test problems (models)

TP	MPS	Number of total products	Number of models	No. of feasible solutions
1	(3,1)	4	2	4
2	(2,2)	4	2	6
3	(1,2,3)	6	3	60
4	(4,2,2)	8	3	420
5	(3,3,2)	8	3	560
6	(3,1,4)	8	3	280
7	(2,2,4)	8	3	310
8	(3,1,5)	9	3	475

Table 4. Small-size test problems (Priorities)

Problem 1

Models / Tasks	1	2	Priorities
1	4	1	-
2	1	-	-
3	-	2	1,2
4	2	-	3
5	3	1	3,4

Problem 2

Models / Tasks	1	2	Priorities
1	6	4	-
2	3	-	-
3	-	-	2
4	4	5	3
5	-	3	3,4
6	2	6	3,5
7	6	4	6
8	5	3	7

Problem 3

Models / Tasks	1	2	3	Priorities
1	7	5	2	-
2	4	-	4	-
3	5	8	5	2
4	7	2	3	3
5	2	-	6	3,4
6	-	7	7	-
7	7	4	-	3,5
8	8	5	2	7
9	-	-	-	8
10	9	3	5	9

Problem 4

Models / Tasks	1	2	3	Priorities
1	6	4	3	-
2	4	5	7	-
3	3	-	4	-
4	-	3	5	1,3
5	-	7	8	4
6	5	8	5	-
7	2	3	-	4,5
8	7	4	-	-
9	-	-	-	7,8
10	9	-	5	5
11	-	4	8	7,10
12	5	6	7	8,11

Applying the GAMS software, the problems are unravelled, and a comparison is made between the outcomes and the MOPSO method's results in Table 5 to accredit the improved algorithm. Based on the outcomes, the solutions attained by applying the MOPSO algorithm are aligned with the GAMS software's solutions.

Table 5. A comparison between MOPSO and GAMS in small-sized problems

TP	MPS	Approach	Run Time(sec)	Z1	Z2	GAP1	GAP2
1	(3,1)	MOPSO	6.092	674	395	0	0.106
		Gams	5.170	674	357		
2	(2,2)	MOPSO	9.426	754	430	0	0.069

		Gams	7.817	754	402		
3	(1,2,3)	MOPSO	10.433	986	548	0.027	0.053
		Gams	13.092	960	520		
4	(4,2,2)	MOPSO	14.328	1120	741	0.027	0.061
		Gams	21.872	1090	698		
5	(3,3,2)	MOPSO	16.392	1256	730	0.044	0.089
		Gams	23.970	1203	670		
6	(3,1,4)	MOPSO	14.826	1390	816	0.045	0.046
		Gams	19.377	1329	780		
7	(2,2,4)	MOPSO	17.164	1428	877	0.029	0.082
		Gams	25.754	1387	810		
8	(3,1,5)	MOPSO	20.086	1432	901	0.024	0.041
		Gams	**28.754**	**1398**	**865**		

4.3.3 | Large-Sized Problems

A number of large-sized problems were designed and unravelled by the improved MOPSO and other MOA approaches. Table 6 represents MPS and some practical solutions for each experiment problem.

Table 6. Large-size test problems

TP	MPS	Number of total products	Number of total products	Number of models	Number of feasible solutions
1	(5,4,2,4,5)	20	20	5	1.4666e+11
2	(3,5,2,4,6)	20	20	5	9.7773e+10
3	(3,5,3,6,5,2)	24	24	6	8.3115e+14
4	(4,3,6,2,7,2)	24	24	6	2.9684e+14
5	(4,6,5,4,3,2,4)	28	28	7	2.1272e+19
6	(5,1,4,3,6,5,4)	28	28	7	8.5089e+18
7	(3,8,4,3,3,2,3,4)	30	30	8	4.4064e+21
8	(1,3,4,2,4,6,3,7)	30	30	8	1.7626e+21

Some self-made large-sized experiment problems are solved using the presented MOPSO. A comparison is performed between run time results and the Cplex solver. As represented by the tests in Table 7, the accurate approach generates reasonable solutions for problems with small sizes; however, it was unable to solve large-sized problems in a proper time.

Table 7. Experimental results (TP: test problem, NT: number of tasks)

		Cplex solver	**Proposed MOPSO**		**NSGA-||**	**MOACO**
TP	NT	Run time (s)	Time (s)	%(time) from optimal	%(solution) from MOPSO	%(solution) from MOPSO
1	5	3.940	5.161	0	0	0
2	8	25.307	41.962	3.5	0	0
3	10	60.397	49.205	2.5	0	2

4	12	217.974	93.492	4.2	5	4
5	18	>3700s	398.693	-	6	7
6	20	>3700s	737.47	-	8	5
7	25	>7300	1073.086	-	2	3
8	30	>7300s	1387.026	-	4	8
9	35	>11000s	1843.943	-	9	7
10	40	>11000s	2640.982	-	4	3
11	40	>11000s	299,946	-	7	5
12	45	>11000s	378,647	-	9	6

Table 7 compares the optimality gap and the problem's computational time. A main difficult aspect of the Cplex solver is its inability to solve large-sized problems in a proper time. The gap percentage from optimal solutions is altered from 0 to 4.2 in the MOPSO algorithm. Considering the run times, the outcomes attained from MOPSO are proportionately better in the near-optimal solution than the MOACO and NSGA-II.

4.3.4 | Sensitivity analysis

Changes in the values of the objective functions are investigated by selecting and analyzing one of the parameters that affect the objective function. The selected parameter is changed to examine the sensitivity of the problem to it. In this model, the conveyor speed parameter has been selected because this parameter has a great impact on determining the overload. This sensitivity analysis has been performed on the objective functions. Due to the multi-objective nature of the model, the change of the mentioned values is compared to the change of a Pareto point and is examined in one problem, shown in Table 8 and Fig. 4.

Table 8. Changes in the objective functions in relation to the conveyor speed parameter

Number	Changes	Objective function 2	Objective function 1
1	0.8	453	**1021**
2	0.9	497	**969**
3	1	511	**943**
4	1.1	543	896
5	1.2	597	832

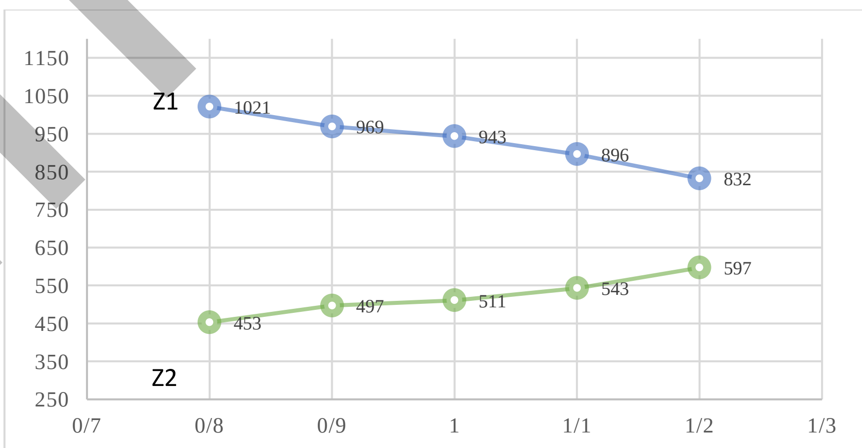

Fig 4. Changes in the objective functions in relation to the conveyor speed parameter

For these alternatives, different points are obtained, as shown in Fig. 5, which are not superior to each other, and it is the management that decides on this increase or decrease. However, it seems that increasing the speed will help more in cost management.

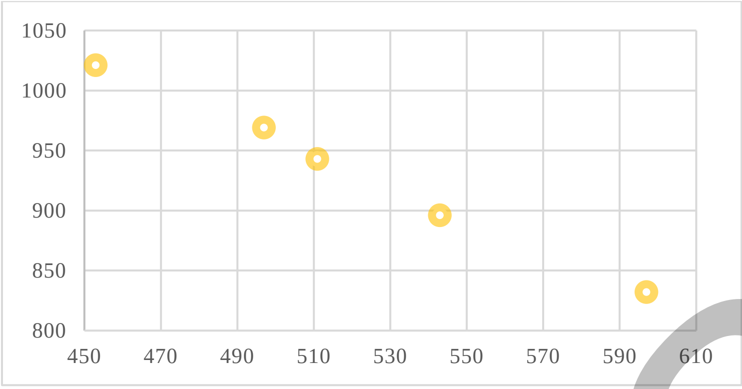

Fig 5. Alternatives obtained by the sensitivity analysis

4.3.5 | Comparison between MOAs

This part utilizes four main comparison metrics to compare the performance of MOAs for the problems with large sizes that were produced.

Spacing metric: shows the equal dispensation of the solutions in the Pareto front by Eq. (29) [48].

$$Spacing = \left[\frac{1}{n-1} \times \sum_{i=1}^{n} \left(\bar{d} - d_i\right)^2\right]^{1/2} \tag{29}$$

indicates the distance between point i, and the closest point of the Pareto front represents the average of

n represents the Pareto optimal solutions' number.

The number of Pareto solutions: This function metric is surveyed as the non-dominated solutions quantity that each algorithm might obtain. Computation outcomes of the spacing metric and Pareto solutions' number are demonstrated in Table 9 and Fig. 6

Table 9. Comparison metrics (S, NS)

	MOPSO		MOACO		NSGA-II	
TP	S	NPS	S	NPS	S	NPS
1	4.33	79	5.72	76	6.21	79
2	5.25	85	5.95	84	5.73	81
3	6.40	74	6.46	68	6.73	73
4	5.57	92	7.38	89	7.91	91
5	5.93	80	6.33	74	7.74	78
6	6.59	101	6.98	94	6.36	96
7	7.38	69	8.16	63	9.06	66
8	7.71	90	8.30	80	8.38	83

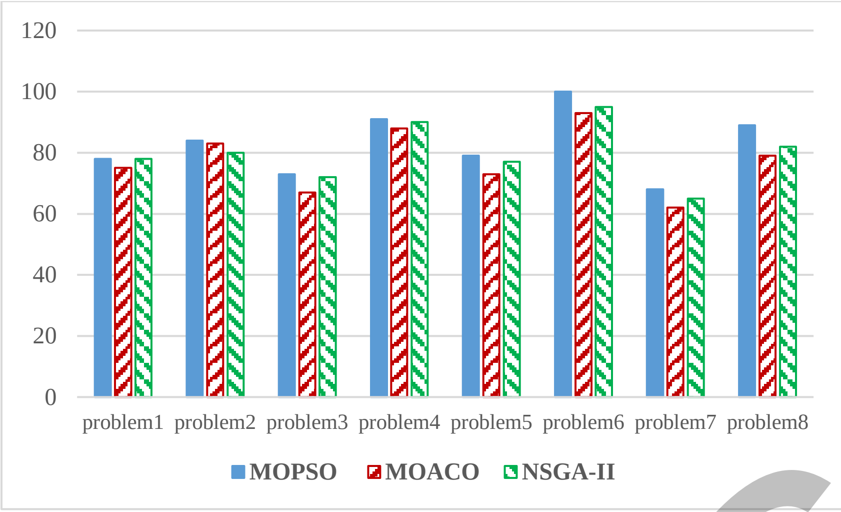

Fig. 6. The number of Pareto solutions

According to the findings, the improved MOPSO algorithm shows a superior function because of the lower value, demonstrating more equal solutions divided in the Pareto front. Due to the number of Pareto solutions, the improved MOPSO functions superiorly and converges more toward the Pareto-optimal Frontier than the other algorithms.

Diversification metric: Eq. (30) calculates this metric to specify the solution set distribution [49].

$$D = \sqrt{\sum_{i=1}^{n} \max\left(\left\| x_i - y_i \right\|\right)} \tag{30}$$

where n represents the solutions' number and indicates the distance between solutions.

Error Ratio: When the Pareto-optimal solutions are identified, this metric calculates the approach's nonconvergence toward the Pareto-optimal frontier through Eq. (31) [50].

$$E = \frac{\sum_{i=1}^{n} e_i}{n} \tag{31}$$

In this equation, n shows the Pareto-optimal solutions' number and, if the solution i indicates the Pareto-optimal frontier' member f and otherwise. The computation outcomes of the diversification metric and the Error Ratio are demonstrated in Table 10 and Fig. 7.

Table 10. Comparison metrics (D, ER)

	MOPSO		MOACO		NSGA-II	
TP	D	ER	D	ER	D	ER
1	14.75	0.1159	12.35	0.2303	10.96	0.1935
2	19.97	0.0905	14.96	0.1208	13.63	0.1373
3	16.27	0.1457	13.64	0.2274	14.88	0.1925

4	20.88	0.2120	18.63	0.2368	14.63	0.2243
5	10.82	0.1839	10.04	0.2502	10.74	0.2607
6	21.34	0.1504	17.63	0.1537	16.86	0.1574
7	18.96	0.1614	14.83	0.1921	12.37	0.2513
8	17.05	0.1757	13.28	0.2176	13.69	0.2364

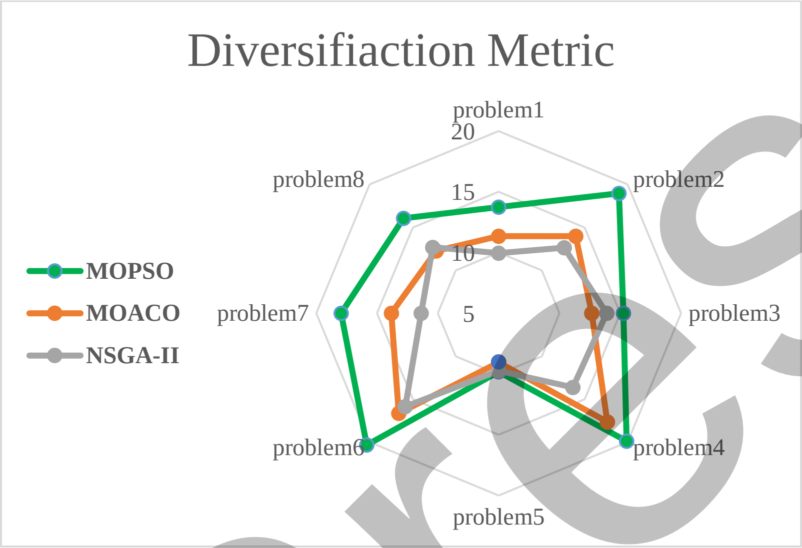

Fig. 7. The diversification metric

According to the error ratio and diversification metric, the improved MOPSO algorithm functions better and converges better than the other algorithms. Thus, based on the outcomes attained from comparison metrics and experiment problems, this algorithm is the proper approach for this problem.

5| Conclusion

In the present paper, we studied the tasks balancing in each station and concurrent model sequencing in a dynamic environment, where all periods impact their subsequent period. Sequence and balance can induce an incomplete and independent task in each process. We reduced the work overload generated due to the line balance and product sequence. With attention to the presented model, decision-makers do not remark any priority in sequencing and balancing problems separately; therefore, the model was presented by evaluating both problems to reach the best composition of the sequence of models and the balance of the tasks in workstations. Moreover, reducing the setup times between two subsequent tasks was more applicable.

The MOPSO algorithm's efficacy suggested for unravelling the MMAL balancing and sequencing problem was studied by unravelling a number of small-sized problems through the GAMS software and performing a comparison between the findings. Usually, the outcomes of both approaches were aligned. Finally, we applied the presented MOPSO and other MOA approaches to answer several produced experiment problems in large sizes. The Taguchi approach was then applied to adjust the values of all algorithms' parameters. Lastly, the MOPSO algorithm was chosen as the problem's most proper approach based on the computational outcomes and a number of comparison metrics, such as the error ratio, amount of Pareto solutions, and diversity. A main difficult aspect of the Cplex solver is its inability to solve large-sized problems in a proper time. The gap percentage from optimal solutions in developed problems is altered from 0 to 4.2 in the MOPSO algorithm. Considering the run times, the outcomes attained from the MOPSO are proportionately better in near-optimal solutions than the MOACO and NSGA-II. The problem's extension to several fields regarding other lines or other assumptions containing zoning constraints might be possible in the future. Moreover, proposing a novel method of accurate approach might be profitably suggested for the sequencing and balancing problem.

Acknowledgements

The authors would like to thank the anonymous pundits of this paper for their study-provoking and perceptive commentary and corrections.

Funding

Kosar University of Bojnord supported the Study with grant number 0207301901.

Data Availability

All data generated or analyzed during this study are included in this article.

Conflicts of Interest

The authors have no conflict of interest to disclose.

References

[1] Soman, C. A., Van Donk, D. P., & Gaalman, G. (2004). Combined make-to-order and make-to-stock in a food production system. *International journal of production economics*, *90*(2), 223–235. DOI: 10.1016/S0925-5273(02)00376-6

[2] Kazemi, S. M., Ghodsi, R., Rabbani, M., & Tavakkoli-Moghaddam, R. (2011). A novel two-stage genetic algorithm for a mixed-model U-line balancing problem with duplicated tasks. *International journal of advanced manufacturing technology*, *55*(9–12), 1111–1122. DOI: 10.1007/s00170-010-3120-6

[3] Manavizadeh, N., Hosseini, N. S., Rabbani, M., & Jolai, F. (2013). A Simulated Annealing algorithm for a mixed model assembly U-line balancing type-I problem considering human efficiency and Just-In-Time approach. *Computers and industrial engineering*, *64*(2), 669–685. DOI: 10.1016/j.cie.2012.11.010

[4] Hamzadayi, A., & Yildiz, G. (2012). A genetic algorithm based approach for simultaneously balancing and sequencing of mixed-model U-lines with parallel workstations and zoning constraints. *Computers and industrial engineering*, *62*(1), 206–215. DOI: 10.1016/j.cie.2011.09.008

[5] Kinable, J., Cire, A. A., & van Hoeve, W. J. (2017). Hybrid optimization methods for time-dependent sequencing problems. *European journal of operational research*, *259*(3), 887–897. DOI: 10.1016/j.ejor.2016.11.035

[6] Abdul Nazar, K. P., & Madhusudanan Pillai, V. (2018). Mixed-model sequencing problem under capacity and machine idle time constraints in JIT production systems. *Computers and industrial engineering*, *118*, 226–236. DOI: 10.1016/j.cie.2018.02.032

[7] Defersha, F. M., & Mohebalizadehgashti, F. (2018). Simultaneous balancing, sequencing, and workstation planning for a mixed model manual assembly line using hybrid genetic algorithm. *Computers and industrial engineering*, *119*, 370–387. DOI: 10.1016/j.cie.2018.04.014

[8] Rabbani, M., Heidari, R., & Farrokhi-Asl, H. (2018). A bi-objective mixed-model assembly line sequencing problem considering customer satisfaction and customer buying behaviour. *Engineering optimization*, *50*(12), 2123–2142. DOI: 10.1080/0305215X.2018.1431234

[9] Öner-Közen, M., & Minner, S. (2017). Impact of priority sequencing decisions on on-time probability and expected tardiness of orders in make-to-order production systems with external due-dates. *European journal of operational research*, *263*(2), 524–539. DOI: 10.1016/j.ejor.2017.05.042

[10] Parveen, N., & Kamble, P. N. (2020). Decision-making problem using fuzzy topsis method with hexagonal fuzzy number. *Advances in intelligent systems and computing, 1025, 421*–430. DOI: 10.1007/978-981-32-9515-5_40

[11] Rudnik, K., & Kacprzak, D. (2017). Fuzzy TOPSIS method with ordered fuzzy numbers for flow control in a manufacturing system. *Applied soft computing journal*, *52*, 1020–1041. DOI: 10.1016/j.asoc.2016.09.027

[12] Kumar, S., Kumar, S., & Barman, A. G. (2018). Supplier selection using fuzzy topsis multi criteria model for a small scale steel manufacturing unit. *Procedia computer science*, 133, 905–912. DOI: 10.1016/j.procs.2018.07.097

[13] Rouyendegh, B. D., Yildizbasi, A., & Üstünyer, P. (2020). Intuitionistic Fuzzy TOPSIS method for green supplier selection problem. *Soft computing*, *24*(3), 2215–2228. DOI: 10.1007/s00500-019-04054-8

[14] Boysen, N., Scholl, A., & Wopperer, N. (2012). Resequencing of mixed-

model assembly lines: Survey and research agenda. *European journal of operational research*, *216*(3), 594–604. DOI: 10.1016/j.ejor.2011.08.009

[15] Boysen, N., Fliedner, M., & Scholl, A. (2009). Sequencing mixed-model assembly lines: Survey, classification and model critique. *European Journal Of Operational Research*, 192(2), 349–373. DOI: 10.1016/j.ejor.2007.09.013

[16] Belkharroubi, L., & Yahyaoui, K. (2022). Solving the mixed-model assembly line balancing problem type-I using a Hybrid Reactive GRASP. *Production and manufacturing research*, *10*(1), 108–131. DOI: 10.1080/21693277.2022.2065380

[17] Thomopoulos, N. T. (1970). Mixed Model Line Balancing with Smoothed Station Assignments. *Management science*, *16*(9), 593–603. DOI: 10.1287/mnsc.16.9.593

[18] Fathi, M., Fontes, D. B. M. M., Urenda Moris, M., & Ghobakhloo, M. (2018). Assembly line balancing problem: A comparative evaluation of heuristics and a computational assessment of objectives. *Journal of modelling in management*, *13*(2), 455–474. DOI: 10.1108/JM2-03-2017-0027

[19] Akpnar, S., & Mirac Bayhan, G. (2011). A hybrid genetic algorithm for mixed model assembly line balancing problem with parallel workstations and zoning constraints. *Engineering applications of artificial intelligence*, *24*(3), 449–457. DOI: 10.1016/j.engappai.2010.08.006

[20] Akpinar, S., Elmi, A., & Bektaş, T. (2017). Combinatorial Benders cuts for assembly line balancing problems with setups. *European journal of operational research*, *259*(2), 527–537. DOI: 10.1016/j.ejor.2016.11.001

[21] Akpinar, Ş., & Baykasoğ lu, A. (2014). Modeling and solving mixed-model assembly line balancing problem with setups. Part I: A mixed integer linear programming model. *Journal of manufacturing systems*, *33*(1), 177–187. DOI: 10.1016/j.jmsy.2013.11.004

[22] Fleszar, K. (2017). A new MILP model for the accessibility windows assembly line balancing problem level 2 (AWALBP-L2). *European journal of operational research*, *259*(1), 169–174. DOI: 10.1016/j.ejor.2016.10.008

[23] Kara, Y., Ozcan, U., & Peker, A. (2007). Balancing and sequencing mixed-model just-in-time U-lines with multiple objectives. *Applied mathematics and*

computation, *184*(2), 566–588. DOI: 10.1016/j.amc.2006.05.185

[24] Battini, D., Faccio, M., Persona, A., & Sgarbossa, F. (2009). Balancing-sequencing procedure for a mixed model assembly system in case of finite buffer capacity. *International journal of advanced manufacturing technology*, *44*(3–4), 345–359. DOI: 10.1007/s00170-008-1823-8

[25] Karabati, S., & Sayin, S. (2003). Assembly line balancing in a mixed-model sequencing environment with synchronous transfers. *European journal of operational research*, 149, 417–429. DOI: 10.1016/S0377-2217(02)00764-6

[26] Öztürk, C., Tunali, S., Hnich, B., & Örnek, A. (2013). Balancing and scheduling of flexible mixed model assembly lines with parallel stations. *International journal of advanced manufacturing technology*, *67*(9–12), 2577–2591. DOI: 10.1007/s00170-012-4675-1

[27] Liao, S. G., Zhang, Y. B., Sang, C. Y., & Liu, H. (2023). A genetic algorithm for balancing and sequencing of mixed-model two-sided assembly line with unpaced synchronous transfer. *Applied soft computing*, *146*. DOI: 10.1016/j.asoc.2023.110638

[28] Cao, Z., & Ma, S. (2008). Balancing and sequencing optimization of the mixed model assembly lines. *International Symposium on Computer Science and Computational Technology*, 1, 732–736. DOI: 10.1109/ISCSCT.2008.144.

[29] Merengo, C., Nava, F., & Pozzetti, A. (1999). Balancing and sequencing manual mixed-model assembly lines. *International journal of production research*, *37*(12), 2835–2860. DOI: 10.1080/002075499190545

[30] Miltenburg, J. (2002). Balancing and scheduling mixed-model U-shaped production lines. *International journal of flexible manufacturing systems*, *14*(2), 123–155.

[31] Kim, Y. K., Kim, J. Y., & Kim, Y. (2006). *An endosymbiotic evolutionary algorithm for the integration of balancing and sequencing in mixed-model u-lines* [presentation]. European journal of operational research (Vol. 168, pp. 838–852). DOI: 10.1016/j.ejor.2004.07.032

[32] Mendes, A. R., Ramos, A. L., Simaria, A. S., & Vilarinho, P. M. (2005). Combining heuristic procedures and simulation models for balancing a PC camera assembly line. *Computers and industrial engineering*, *49*(3), 413–431.

DOI: 10.1016/j.cie.2005.07.003

[33] Van Hop, N. (2006). A heuristic solution for fuzzy mixed-model line balancing problem. *European journal of operational research*, 168, 798–810. DOI: 10.1016/j.ejor.2004.07.029

[34] Manavizadeh, N., Rabbani, M., & Radmehr, F. (2015). A new multi-objective approach in order to balancing and sequencing U-shaped mixed model assembly line problem: a proposed heuristic algorithm. *International journal of advanced manufacturing technology*, *79*(1–4), 415–425. DOI: 10.1007/s00170-015-6841-8

[35] Kucukkoc, I., & Zhang, D. Z. (2017). Balancing of mixed-model parallel U-shaped assembly lines considering model sequences. *International journal of production research*, *55*(20), 5958–5975. DOI: 10.1080/00207543.2017.1312586

[36] Hamzadayi, A., & Yildiz, G. (2013). A simulated annealing algorithm based approach for balancing and sequencing of mixed-model U-lines. *Computers and industrial engineering*, *66*(4), 1070–1084. DOI: 10.1016/j.cie.2013.08.008

[37] Modrák, V., & Moskvich, V. (2012). Impacts of RFID implementation on cost structure in networked manufacturing. *International journal of production research*, *50*(14), 3847–3859. DOI: 10.1080/00207543.2011.594968

[38] Faccio, M., Gamberi, M., & Bortolini, M. (2016). Hierarchical approach for paced mixed-model assembly line balancing and sequencing with jolly operators. *International journal of production research*, *54*(3), 761–777. DOI: 10.1080/00207543.2015.1059965

[39] Meng, K., Tang, Q., Zhang, Z., & Li, Z. (2023). Robust mixed-model assembly line balancing and sequencing problem considering preventive maintenance scenarios with interval processing times. *Swarm and evolutionary computation*, *77*, 101255. DOI: 10.1016/j.swevo.2023.101255

[40] Edis, E. B., Sancar Edis, R., & Ilgin, M. A. (2022). Mixed integer programming approaches to partial disassembly line balancing and sequencing problem. *Computers and operations research*, *138*. DOI: 10.1016/j.cor.2021.105559

[41] Zeng, Y., Zhang, Z., Yin, T., & Zheng, H. (2022). Robotic disassembly line

balancing and sequencing problem considering energy-saving and high-profit for waste household appliances. *Journal of cleaner production*, *381, 135-209*. DOI: 10.1016/j.jclepro.2022.135209

[42] Chawasemerwa, T., Taifa, I. W., & Hartmann, D. (2018). Development of a Doctor Scheduling System: A Constraint Satisfaction and Penalty Minimisation Scheduling Model. *International journal of research in industrial engineering*, *7*(4), 396–422. DOI: 10.22105/riej.2018.160257.1068.

[43] Shahvaroughi Farahani, M., Farrokhi-Asl, H., & R. (2023). Hybrid metaheuristic artificial neural networks for stock price prediction considering efficient market hypothesis. *International journal of research in industrial engineering*, *12*(3). DOI: 10.22105/riej.2023.361216.1336

[44] Ejlali, B., Bagheri, S. F., & Ghaziyani, K. (2019). Integrated and periodic relief logistics planning for reaction phase in uncertainty condition and model solving by PSO algorithm. *International journal of research in industrial engineering*, *8*(4). DOI: 10.22105/riej.2020.219060.1120

[45] Coello Coello, C. A. (1999). A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques. *Knowledge and information systems*, *1*(3), 269–308. DOI: 10.1007/bf03325101

[46] Gray, C. T. (1988). Introduction to quality engineering: Designing quality into products and processes, G. Taguchi, Asian productivity organization, 1986. number of pages: 191. price: $29 (U.K.). *Quality and reliability engineering international*, *4*(2), 198–198. DOI: 10.1002/qre.4680040216

[47] Jensen, W. A. (2017). Response Surface Methodology: Process and Product Optimization Using Designed Experiments 4th edition. *Journal of quality technology*, *49*(2), 186–188. DOI: 10.1080/00224065.2017.11917988

[48] Srinivas, N., & Deb, K. (1994). Muiltiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary computation*, *2*(3), 221–248. DOI: 10.1162/evco.1994.2.3.221

[49] McMullen, P. R. (1998). JIT sequencing for mixed-model assembly lines with setups using tabu search. *Production planning and control*, *9*(5), 504–510. DOI: 10.1080/095372898233984

[50] Van Veldhuizen, D. A., & Lamont, G. B. (1999). Multiobjective

evolutionary algorithm test suites. *Proceedings of the acm symposium on applied computing, 351–357*. DOI: 10.1145/298151.298382