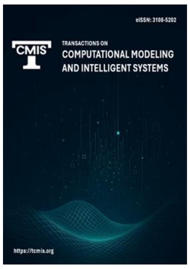

Transactions on
Computational Modeling
and Intelligent Systems

Volume 1, October 2025, Article 10008
https://doi.org/10.65112/tcmis.10008

eISSN: 3108-5202
www.tcmis.org

RESEARCH ARTICLE

Rebalancing of multi-manned mixed-model assembly lines with task relocation restrictions

Damla Camli[1*], Ibrahim Kucukkoc[2] and Zixiang Li[3,4]

[1] Continuous Improvement Department, Önen Food Company, Çanakkale, Turkiye.
[2] Department of Industrial Engineering, Balikesir University, Balikesir, Turkiye.
[3] Key Laboratory of Metallurgical Equipment and Control Technology of Ministry of Education, Wuhan University of Science and Technology, Wuhan, Hubei, China.
[4] Hubei Key Laboratory of Mechanical Transmission and Manufacturing Engineering, Wuhan University of Science and Technology, Wuhan, Hubei, China.

[*]Corresponding Author. Email: damla.camli@dardanel.com.tr (D. Camli)

Article Information

Received: 15 August 2025
Accepted: 2 October 2025
Published: 3 October 2025

AMS 2020 Classification: 90B10, 90B35

Abstract

Increasing competition and customized demands have led companies to use assembly lines more flexible and efficiently. Companies need to frequently rebalance their lines to adapt changes either in the product model demand or task processing times. During which, some tasks will be required to assign a different workstation (causing a change in the task allocation) due to the nature of the rebalancing procedure. However, as the number of relocations made during rebalancing increases, the likelihood of costs and quality errors will also arise. This study aims to efficiently balance mixed-model assembly lines while restricting the number of relocations to a limited value. A mixed-integer program is proposed to maximize line efficiency (minimising both cycle time and number of workstations, called type-E) considering the number of task relocations up to a certain value. An iterative algorithm is also developed for solving large-sized problems. The model allows lower and upper bounds to be imposed on the cycle time and aims to avoid 'substantial' changes in task assignments during rebalancing. The multi-manned workstation case is also integrated, which gives the advantage of determining the number of operators, i.e., increasing line efficiency. Tests have shown that the heuristic algorithm achieves competitive solutions in compare with the mixed-integer programming model within short periods of time, including large-size problems.

Keywords: Mixed-model assembly line, type-E, task relocation restrictions, rebalancing, mixed-integer programming, multi-manned workstations.

1. Introduction

Assembly lines ensure efficient use of resources which usually yields in cost reduction. At the same time, it increases the efficiency of the system and the production per unit time by minimising the number of workstations and/or cycle time. An assembly line consists of a set of workstations connected to each other by a transportation mechanism (conveyor or moving belt) [1, 2]. Each workstation is responsible from executing a set of tasks within the cycle time determined. The allocation of tasks to the workstations aiming to minimise or maximise an objective function under certain constraints is called the assembly line balancing problem [3]. The precedence relationship constraints and cycle time constraints are mandatory while some other constraints might be added based on the work environment, and the objectives sought.

Due to customized demands and increasing competition, companies have utilised mixed-model assembly lines, where similar models of a product are assembled on the same line with no need for set-up [4, 5]. It is of great strategic importance for companies to quickly adapt to changing market conditions, increase customer satisfaction and gain global competitive advantage. At this point, mixed model assembly lines provide many advantages to companies. Improving the processes increases productivity by producing more products in a shorter time and thus resource utilisation is minimised. This helps to control and reduce the costs of companies. In this sense, there are operational decisions that companies should make in order to maximise line efficiency. Some of them are the number of operators to be assigned to stations, cycle time and the number of stations to be opened [6].

In this study, a mathematical model that maximises line efficiency by minimising the multiplication of the number of stations and cycle time of mixed model assembly lines is developed. In this way, the aim is to meet rapidly changing customer demands in the fastest way possible by minimising idle time and optimising production capacity and resources with a balanced production line. At the same time, the mathematical model is formulated by allowing multi-manned workstations. Multi-manned workstations allow the utilisation of a cycle time lower than the maximum task processing time by multiplying the workstation capacity. In the assembly line rebalancing problem, the assignment of jobs to completely different stations implies major changes to the existing assembly line. Due to this situation, the number of allowed task relocations is added as a new constraint to the model. In the developed model, by imposing lower and upper limits on the cycle time (C_{min} and C_{max} respectively) flexibility is provided to the model in order to maximise the line efficiency. The method we propose provides high-quality solutions for the companies, even for large size problems with the iterative algorithm integrated into the mathematical model.

The rest of the article is organised as follows: Section 2 provides the classification of assembly line balancing problems and a comprehensive literature review on the subject under consideration. Section 3 describes the mixed-integer programming model as well as the iterative algorithm, following a basic introduction of the notation and definitions. Section 4 presents the data set on which the model is tested, followed by the test results and their interpretation. Finally, Section 5 concludes the work with a discussion and potential topics for future studies.

2. Literature review

A review of the literature on assembly line balancing problems shows that assembly line balancing (ALB) problems are classified as follows according to the objective sought [3]:

- Type-1 ALB: To minimise the number of stations when the cycle time is given.
- Type-2 ALB: To minimise the cycle time when the number of stations is given.
- Type-E ALB: To maximise line efficiency minimising both the number of stations and the cycle time.
- Type-F ALB: To find a feasible solution given both the number of stations and the cycle time.

One might see Boysen et al. [7], Battaïa and Dolgui [8], and Li et al. [9] for the recent surveys and different classification schemes.

2.1. Literature review on MALBP

The literature on mixed-model assembly line balancing problem (MALBP) is rather extensive. The preliminary works in the literature are done by Thomopoulos [10, 11] who presented the first methods for mixed-models and worked on workload balancing. Subsequent research focused on multi-objective approaches and various optimisation techniques. Thomopoulos [11] and Gökçen [12] developed multi-objective models, while Emde [13] investigated workload smoothing methods. Belkharroubi and Yahyaoui [14] developed methods to reduce the number of workstations. In a later work by the same authors, Belkharroubi and Yahyaoui [15] aimed to minimise the number of workstations in mixed models where a greedy search procedure was employed.

Mönch et al. [16] emphasise the importance of adopting a single and flexible assembly line to gain competitive advantage in challenging conditions such as changing customer demands and short product lifetimes and show that variable takt times (average cycle time) increase labour productivity and simplify complex assembly lines. Meng et al. [17] aimed to minimise the number of stations by considering ergonomic conditions in mixed-model lines. Zhang et al. [18] proposed a MINLP model that aims to minimise the cost of opening station and the recruitment cost of operators. The study is important in terms of economic sustainability. Nourmohammadi et al. [19] aimed to minimise the cycle time at a certain number of stations. They developed a multi-objective MILP model to minimise cycle time, maximum ergonomic risks of stations and total ergonomic risks. Li et al. [20] focused on a mixed model type-2 assembly line balancing problem. They developed a mixed integer-based heuristic with a cost-oriented objective function and a customised neighbourhood search algorithm. Delice et al. [21] aimed to minimise the sum of workstation setup cost, supermarket setup cost and transportation cost in mixed-model assembly lines. The aim of the model developed in Dalle Mura and Dini [22] is to minimise the number of workstations, while balancing noise exposure and energy expenditure of operators. Çil et al. [23] addressed the type-2 MALBP (i.e. MALBP-2) and multiple operator assignment problem. They optimised the number of operators to be assigned to stations while minimising the cycle time. Workers and robots can work together as operators at stations. Alakas and Toklu [24] also considered the MALBP-2 and proposed a constraint programming approach as a solution method. Yuwei et al. [25] considered the type-1 MALBP (i.e. MALBP-1) and used genetic algorithm as the solution method. Yaphiar et al. [26] considered a mixed model type-1 assembly line balancing and operator assignment problem. Each workstation may contain only humans as operators, only robots, or humans and robots simultaneously. Lalaoui and Afia [27] considered the MALBP-2 and used simulated annealing algorithm as a solution method. Sadeghi et al. [28] considered the MALBP-1. Variable neighbourhood descent and sequential positional weighting algorithms were used as solution methods. With one operator at each station, given the available time, it minimises the number of workstations while smoothing the workload of the operators. Naderi et al. [29] developed a mathematical model to minimise the number of operators to work for a given cycle time and number of stations and used a linear relaxation method. An efficient logic-based Benders decomposition algorithm is also developed. Kucukkoc et al. [30] considered the underground workstations in mixed-model two-sided lines and proposed a mathematical model as well as an ant colony optimization algorithm for solving the problem. Tanhaie et al. [31] addressed the MALBP-1. The developed mathematical model aims at both minimising the number of stations and minimising the total cost of task repetition and operator assignment to perform assembly tasks. Anh and Van Hop [32] studied the mixed model assembly line balancing problem of type-1, task processing time is defined and modelled as fuzzy stochastic processing times considering the natural uncertainties occurring in production. Schibelbain et al. [33] addressed a robotic assembly line problem under practical constraints, focusing on the challenges of mixed-model manufacturing. Two main issues are highlighted, namely the computational costs increasing with the problem size and the difficulties in developing a practical performance model using mathematical programming. Sikora [34] proposed an exact method and heuristic algorithm for mixed-model assembly lines, taking into account the random production sequence. Branch-and-bound algorithm using Markov chains is proposed to optimize the assignment of tasks, the length of the stations. Zhang et al. [35] addressed MALBP with a focus on human and robot co-operation in the context of Industry 5.0, together with the challenges arising from uncertainties and dynamic market demands.

Overall, this review of the literature shows that the MALBP covers a wide range of research, and various solution strategies have been used.

2.2. Literature review on type-E ALB problem

Studies on type-E assembly line balancing in the literature are rather limited. Recently, El Machouti et al. [36] reviewed the literature of simple assembly line balancing problems of type-E. Going back to the earlier studies, Plans and Corominas [37] developed MILP and heuristics for type-E ALB problem to improve efficiency. Then Wei and Chao [38] developed a new exact solution method to solve the type-E ALB problem based on solving several SALBP-2 formulations. Zacharia and Nearchou [39] proposed a direct solution method based on triangular fuzzy numbers without solving several type-1 or type-2 ALB problems [40]. Esmaeilbeigi et al. [40] developed a mixed-integer linear programming model to minimise cycle time, number of stations and smoothness index. Jusop and Rashid [41] solved the type-E ALB problem using genetic algorithm and also considered resource constrained scenarios. Khalid et al. [42] proposed a method to improve line efficiency with artificial immune systems. These studies addressed various aspects of the type-E ALB problems and presented different solution methods.

Manavizadeh et al. [43] conducted one of the preliminary works on type-E MALBP (i.e. MALBP-E), minimising both cycle time and the number of workstations as well as other objectives. A multi-objective genetic algorithm was proposed for this aim. Su et al. [44] proposed a combination of Petri net and binary search using a two-stage heuristic algorithm to solve the MALBP-E. Legesse et al. [45] presented metaheuristics to minimise cycle time and number of stations in stochastic environments.

Rebalancing studies usually focussed on cost minimisation and can be performed for straight line, parallel line, U-shaped and two-sided lines [46]. Çimen et al. [47] reviewed the related studies in the literature for assembly line rebalancing problems. According to Çimen et al. [47], the objective function of most of the studies for assembly line rebalancing problems are maximising the workload balance, minimising the rebalancing costs, and minimising task relocation. Also, a small proportion of the existing studies address to mixed-model assembly line rebalancing. Agpak [48] aimed to minimise cycle time fluctuations caused by rebalancing. Yang et al. [49] aimed to reduce the number of stations and minimised the total processing time for reassigned tasks with a cost function in mixed-model lines. Serin et al. [50] minimised the workload change and rebalancing cost. Oliveira et al. [51] developed mathematical and heuristic models for minimising the number of stations and average workload in an environment of arbitrarily mixed-models of products. Makssoud [52] developed a model to minimise changes and costs in an existing assembly line.

As seen in the survey given above, there are quite limited number of studies in the literature studying the type-E mixed-model assembly line rebalancing problem. This study contributes to literature addressing a mixed-model assembly line rebalancing problem allowing multi-manned operators under task relocation constraints. Different from the literature, the number of task relocations is not minimised but restricted to a certain number which can be decided by the line manager. In this way, companies will be able to easily adapt the newly balanced task allocation with minimal change requirements for operators and/or equipment. Moreover, managers will have the flexibility to determine the magnitude of the change based on the infrastructure they have. Hence, the mathematical model to be presented in the following section provides the decision maker with the advantage of maximising the line efficiency by adding the maximum task relocation limit (B), which can be made in the existing assembly line considering the multi-operator conditions in mixed-model lines.

3. Problem description and mathematical model

More than one product model ($m \in M$) is assembled on a mixed-model assembly line in an intermixed way. Each task is re-assigned to exactly one workstation ($k \in K$) considering the precedence relationship constraints caused by technological requirements. S represents the set of precedence relationships, where $(r, s) \in S$ if task r immediately precedes task s in the precedence relationships diagram. The precedence relations by different models are constituted to a common diagram to ensure that common tasks between models are assigned to the same workstation. Each task ($i \in I$) needs a deterministic processing time (t_{im}) and the sum of the processing times of tasks assigned to a workstation cannot exceed the cycle time (C) on a model basis.

The existing task allocation of the line (called 'current state') is also considered during line balancing (rebalancing) and the total number of task relocations is restricted up to a certain value. M_{ik} shows the current allocation of tasks to workstations in the current state. If task i was assigned to workstation k, M_{ik} gets the value 1; otherwise it gets 0. If task i is assigned to a different workstation than the current state, this is considered as a 'relocation' and is expressed by the decision variable f_i. Thus, if the workstation to which task i is assigned has changed, f_i gets the value 1; otherwise, it gets the value 0. The developed model rebalances the line in compliance with the maximum number of relocations allowed, represented by B. To reflect the real-world conditions, more than one operator can be assigned to a workstation. OP denotes the total number of operators allowed while RP denotes the maximum number of operators that can be assigned to a workstation.

The notation used in the developed model is given in Table 1.

Table 1. Notation used in the mathematical model.

Sets	
Stations	$k \in K$
Tasks	$i, r, s \in I$
Models	$m \in M$
Parameters	
Maximum number of stations	K_{max}
Minimum cycle time	C_{min}
Maximum cycle time	C_{max}
Processing time of task i for model m	t_{im}
Set of precedence relationships	S
A precedence relationship; task r immediately precedes task s	$(r, s) \in S$
The maximum number of relocations allowed	B
If task i is assigned to station k in the current state, 1; otherwise, 0	M_{ik}
Decision Variables	
Cycle time	C
If task i is assigned to station k, 1; otherwise, 0	x_{ik}
The number of operators assigned to workstation k	z_k
If the station where task i is assigned has changed, 1; otherwise, 0	f_i

The MINLP model developed for the mixed-model type-E assembly line rebalancing problem (built over the work by [53]) is as follows [54]:

Objective function:

$$Minimise\ C \sum_{k=1}^{K_{max}} z_k \tag{1}$$

Subject to:

$$\sum_{k=1}^{K_{max}} x_{ik} = 1 \quad \forall\, i \tag{2}$$

$$\sum_{i=1}^{N} t_{im} x_{ik} \leq C z_k \quad \forall\, m, k \tag{3}$$

$$\sum_{k=1}^{K_{max}} k(x_{rk} - x_{sk}) \leq 0 \quad \forall\, (r,s) \in S \tag{4}$$

$$C \geq C_{min} \tag{5}$$

$$C \leq C_{max} \tag{6}$$

$$z_k \leq RP \quad \forall\, k \tag{7}$$

$$\sum_{k=1}^{K_{max}} z_k \leq OP \tag{8}$$

$$\sum_{i=1}^{N} f_i \leq B \tag{9}$$

$$x_{ik} - M_{ik} \leq f_i \quad \forall\, i,k \tag{10}$$

$$z_{k-1} \geq z_k \quad \forall\, k \mid k \neq 1 \tag{11}$$

$$x_{ik}, f_i \in \{0,1\} \quad \forall\, i,k \tag{12}$$

$$C, z_k \in \mathbb{Z}^+ \tag{13}$$

The objective (given in Eq. (1)) is to allocate tasks to workstations for maximising the line efficiency (type-E ALB), which equivalents to minimising the multiplication of the cycle time and the number of workstations considering that the total task time ($\sum_i \sum_m t_{im}$) is constant. Since tasks cannot be split, Constraint (2) guarantees that each task is assigned to exactly one workstation. Constraint (3) ensures that the total processing time of the tasks assigned to each workstation for each model cannot exceed the cycle time. For this, it is important whether the workstation is open or not, because tasks are only assigned to open workstations. Constraint (4) ensures that priority relationship constraints are not violated. If task r is an immediate predecessor of task s, then task s cannot be assigned to an earlier workstation than task r. Constraints (5) and (6) ensure that the cycle time gets a value within the minimum and maximum values allowed. Constraint (7) restricts that the number of operators assigned to a station should not exceed RP, given as a parameter. Constraint (8) ensures that the total number of operators assigned to workstations across the line does not exceed the total number of operators available in terms of operational resources. The total number of changes in the task assignments during rebalancing cannot the maximum number of relocations allowed, this is expressed by Constraint (9). Constraint (10) is to check whether each task is assigned to the same station both in the current state and rebalanced solution. If a task is assigned to a different workstation rather than the one to which it was currently assigned, it means a change. Constraint (11) ensures that workstations are utilised sequentially starting from 1. Constraints (12) and (13) define the sign constraints regarding the decision variables.

3.1. rMILP

Due to the nonlinearity (owing to the multiplication of endogenous variables in the objective function) and complexity of the proposed model, an optimal solution cannot be obtained in a reasonable time for large size problems. Only small size problems can be solved with the MINLP model. Therefore, an iterative algorithm is integrated into the MINLP model in order to solve large size problems. This approach is commonly referred to as ε-constraint method in the literature (see [55], [56] and [57]). The notation, sets, parameters and decision variables used for the rMILP model are generally the same as those used for the MINLP model. The major difference is that the cycle time is not defined as a decision variable in the rMILP model. Defining the cycle time as a parameter rather than a decision variable transforms the model into a type-I assembly line balancing problem. The mathematical model is run iteratively for each value of C between C_{min} and C_{max}. This also linearises the objective function of the model. The flow chart of the utilised algorithm is shown in Figure 1.

Variables used during the procedure are C_{best} which is best cycle time value found, K_{best} which is best number of stations found, $Model_{best}$ which is the best objective function value found. Firstly, C_{min} and C_{max} are determined for the given problem. C_{min} is determined in such a way that it is not lower than the processing times of the tasks. C_{max} is taken within a certain range in order not to reduce the production speed significantly. The maximum number of relocations that can be made to the existing assembly line is also determined based on the characteristics of the problem and the resources available. The current state M_{ik} is also gathered based on the existing line configuration of the line to be rebalanced. After initialising all parameters and variables, the developed model is solved by setting the cycle time to C_{min}. It is checked whether the line efficiency of the obtained solution is better than the line efficiency of the current best. If a better solution is found, the algorithm continues with the obtained solution by increasing the cycle time by C_{step}. This loop continues until the cycle time reaches the maximum cycle time limit, C_{max}, and the algorithm terminates reporting the best solution with the highest line efficiency.

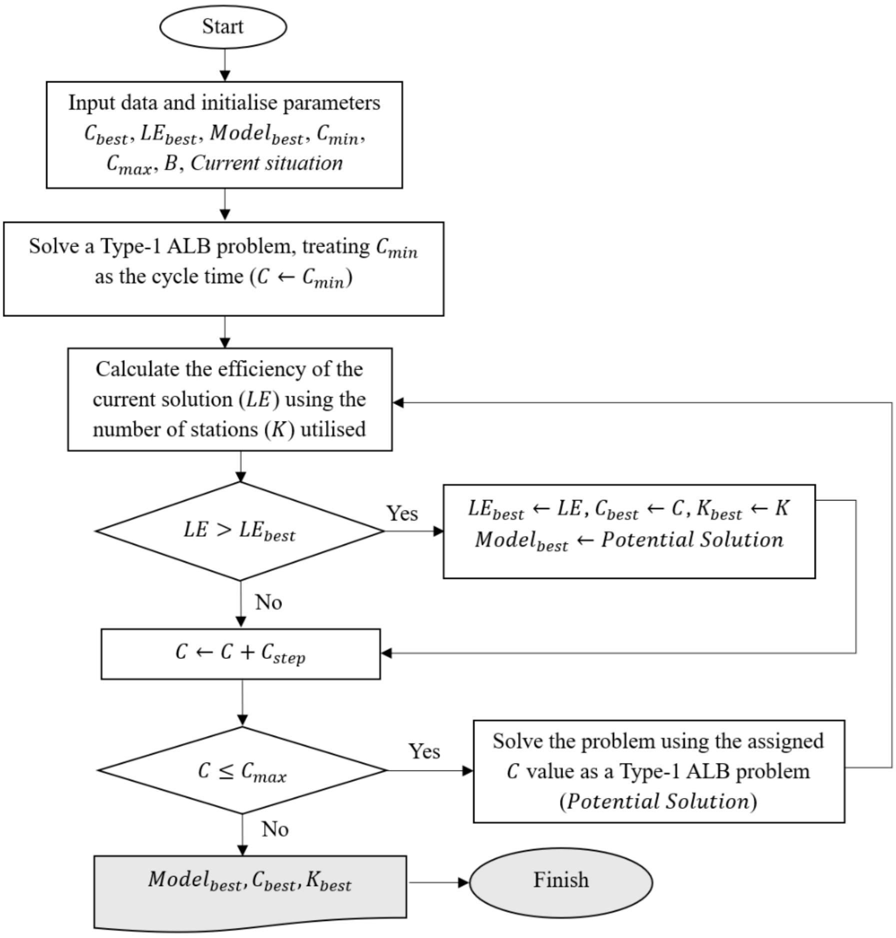

Figure 1. Flow chart of the rMILP [58].

3.2. Numerical example

A numerical example, obtained from the P19 Thomopoulos, composed of 19 tasks and three different models [63] will be solved using the proposed models. The problem was solved using rMILP. The parameters (C_{min}, C_{max}, K_{max}, and OP) used for the problem are assumed to be 1.5, 3.0, 5 and 5, respectively. The model was tested for different B values by starting from 0, i.e. the current state for verification, and increasing by $C_{step} = 1$ up to the number of tasks (considering an extreme case that all tasks may be assigned to a different workstation). Thus, the algorithm was run through 19 iterations (except verification phase) in total and outputs were obtained. The optimum solution was found within 0.029 seconds for $RP = 1$ and 0.019 seconds for $RP = 2$.

Figure 2 shows the number of workstations cycle time and line efficiency for each B value. When reporting the B values, as mentioned before, only the B values that create a break (causing an increase in the line efficiency) are reported for the length constraints.

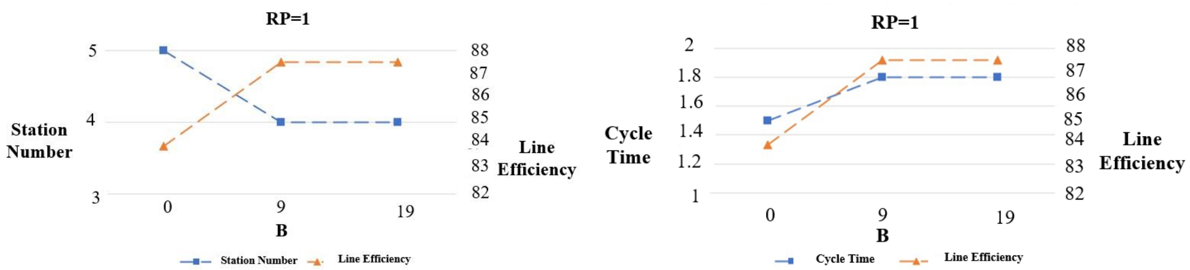

Figure 2. Change in the number of stations, cycle time and line efficiency across different B values when RP=1.

When Figure 2 is analysed, it is seen that the number of workstations is 5, the cycle time is 1.5 time-units and the line efficiency value is 84% for $B = 0$ (current state). Line efficiency increases as the B value increases indicating that a more effective line balancing solution is obtained as more changes are allowed. For example, when $B = 9$ (allowing up to nine task assignment relocations), the optimum solution requires four workstations with a cycle time value of 1.8 time-units, achieving a line efficiency of 87.50%. This indicates that a balancing solution with a line efficiency of as high as 87.5% can be achieved with changing the assignments of nine tasks. The line efficiency does not improve more even it is allowed to change the assignments of all tasks (see the situation of $B = 19$). These results show the effect of the proposed approach. Figure 3 shows the current situation, and the best solution obtained by rMILP when $RP = 1$.

(a)

Workstation No	WS1	WS2	WS3	WS4	WS5
Task Assignments	2,3,4,5	15	3,7,11,13,17	6,9,10,12,14,18	15,16,19
Workload Model 1	1.0	0.5	1.29	1.2	1.1
Workload Model 2	1.0	0.5	1.5	1.3	1.4
Workload Model 3	1.4	1.5	1.4	1.4	1.5

(b)

Workstation No	WS1	WS2	WS3	WS4
Task Assignments	2,3,4,5	1,8,11,16	7,9,10,12, 13,14,17	6,15,18,19
Workload Model 1	1.0	0.8	1.7	1.6
Workload Model 2	1.2	0.9	1.8	1.8
Workload Model 3	1.8	1.8	1.8	1.8

Figure 3. (a) Current situation and (b) the best solution obtained by rMILP when RP=1.

In the second scenario, it is allowed that two operators can be assigned to a workstation ($RP = 2$) for the same test problem. Figure 4 shows the change in the number of workstations, cycle time and line efficiency for each B value in the case of $RP = 2$. When calculating the line efficiency, the number of operators (OP) should be taken into account instead of the number of stations since some of the workstations might be allocated more than one operator. For example, if two operators are assigned to a workstation, the capacity of the station is doubled (see the column OP in Table 2).

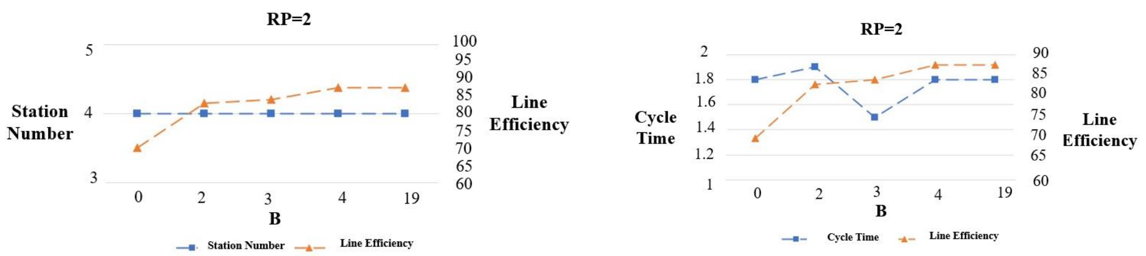

Figure 4. Combinations of cycle time and number of stations across varying B values when RP=2.

As seen in Figure 4, for $B = 0$, the number of workstations is four and the cycle time is 1.8 time-units corresponding the 70% of line efficiency. When the results are analysed, it is seen that the line efficiency improves with the increase in the B value. For example, when only two relocations are allowed (i.e. $B = 2$), line efficiency increases to 82.89% with $K = 4$ $(also\ OP = 4)$ and $C = 1.9$. When $B = 4$, cycle time can be reduced to 1.8 time-units while keeping the number of workstations constant, which increases the line efficiency to 87.50%. Beyond this point, line efficiency does not improve even more relocations are allowed for the task assignments. Table 2 presents the best solutions across different B values. Decision makers can decide which solution to utilise based on their resources taking into account the extent of change they will be exposed to.

Table 2. Best results for P19 across different B values when RP=2

B	Cycle Time	OP	Number of Stations	OP*C	LE (%)
0	1.8	5	4	9	70.00
2	1.9	4	4	7.6	82.89
3	1.5	5	4	7.5	84.00
4	1.8	4	4	7.2	87.50*
19	1.8	4	4	7.2	87.50*

* *The overall best objective function value*

Figure 5 shows the current situation and the best solution obtained by rMILP when $RP = 2$.

(a)

Workstation No	WS1	WS2	WS3	WS4
Number of Operators	2	1	1	1
Task Assignments	1,2,3,4,5	9,11,13, 14,17,18	6,7,8,10, 12,16	15
Workload Model 1	1.5	1.8	0.7	1.1
Workload Model 2	1.2	1.8	1.4	1.3
Workload Model 3	2.8	1.1	1.8	1.5

(b)

Workstation No	WS1	WS2	WS3	WS4
Number of Operators	1	1	1	1
Task Assignments	2,3,4,5	1,9,10,11, 13,17	6,7,8,10, 12,14,16	15,18,19
Workload Model 1	1.0	1.8	0.9	1.4
Workload Model 2	1.2	1.1	1.6	1.8
Workload Model 3	1.8	1.8	1.8	1.8

Figure 5. (a) Current situation and (b) the best solution obtained by rMILP when RP=2.

These results indicate that the optimal solution could change in response to a change in the maximum number of relocations allowed practically. A higher B value could allow a better solution in terms of line efficiency. More importantly, the same line efficiency could be obtained with a lower number of changes in task relocations.

This section presented the problem definition, the mathematical model, the workflow diagram of the proposed approach, and a small illustrative example of the model. The results of the computational experiments will be provided in the next section.

4. Computational experiments

In this section, the comparison and performance analysis of the two proposed models, MINLP and rMILP, will be presented for large datasets. The public datasets of different sizes, different production models, and processing times were solved for this aim. The models were coded in Python 3.0 programming language and solved with Gurobi Optimizer. The personal computer used for the tests has Intel(R) Core (TM) i5-10300H CPU at 2.50GHz with 16 GB RAM.

MINLP and rMILP models were run using six different data sets [59]. The sizes of the datasets used are different from each other and therefore the precedence relationship diagrams, number of tasks and number of models are different from each other. This analysis proves how good the performance of rMILP is and shows that it differs from the MINLP model by finding the optimal result in a short time on large data sets. Table 3 shows the parameters used for all data sets when $RP = 1$. Minimum cycle time, maximum cycle time, maximum number of stations and step size (C_{step}) are given for each problem.

MINLP and rMILP models were run under 1800 seconds time limit for six different data sets using the parameters given in Table 3. The results of the tests are presented in Table 4. Table 4 shows the cycle time, number of stations and line efficiency values of rebalancing solutions for each data set when $RP = 1$. Eq. (14) calculates the relative percentage deviation, i.e. $RPD(\%)$, where the objective is to maximise the line efficiency.

$$RPD(\%) = \frac{LE_{rMILP} - LE_{MINLP}}{LE_{MINLP}} \times 100 \tag{14}$$

Table 3. Parameters used for datasets

Datasets/ Parameters	C_{min}	C_{max}	K_{max}	C_{step}
P-70 Tongue	156	234	18	5
P-83 Arcus	3691	5537	14	111
P-89 Lutz-2	20	35	17	-
P-94 Mukherje	171	257	20	6
P-111 Arcus	6615	9923	16	198
P-148 Bartholdi-1	383	575	12	-

Table 4. Test results for MINLP and rMILP when $RP = 1$.

Dataset	MINLP					rMILP					
	C	K	OP	LE (%)	Time (sec)	C	K	OP	LE (%)	Time (sec)	RPD (%)
P-70 Tongue	220	16	16	98.83	1200	176	20	20	98.83	202.00	0.00
P-83 Arcus	5467	14	14	98.51	1800	5451	14	14	98.80	269.39	0.29
P-89 Lutz-3	35	14	14	98.46	65.52	35	14	14	98.46	58.67	0.00
P-94 Mukherje	214	20	20	98.21	1800	213	20	20	98.67	694.27	0.46
P-111 Arcus	9424	16	16	96.59	1800	8397	18	18	96.36	914.00	-0.23
P-148 Bartholdi-1	470	12	12	99.57	406.03	470	12	12	99.57	388.69	0.00

Table 4 shows the line efficiencies of the rMILP and MINLP models for each data set. MINLP finds the optimal solutions for three test cases. For P-111 Arcus, rMILP finds a worse solution with a -0.23% RPD. This might be caused by the value of C_{step}, as the optimal C value might be missed. Minimising the C_{step} to a minimal unit might help finding better solutions if one ignores the increased solution time. For the three test problems (i.e. P-70 Tongue, P-89 Lutz-3, and P-148 Bartholdi-1), both models give the same result under 1800 seconds time limit. However, for P-70 Tongue, the rMILP model obtains the optimal result in as short as 202 seconds, while the MINLP model requires 1200 seconds to find the optimal solution. For the remaining two problems (P-83 Arcus and P-94 Mukherje) rMILP shows better performance than the MINLP with an average RPD of 0.375%.

Overall, the results show that for all large datasets, the rMILP algorithm maximizes line efficiency by obtaining good solutions in less time than the MINLP algorithm. The MINLP algorithm was run even for 60 minutes when it produced feasible solutions, but no further improvement was observed, suggesting that its performance does not improve after a certain point. These findings demonstrate that the rMILP algorithm offers companies a clear advantage in terms of both computational efficiency and solution quality. For the datasets examined, the rMILP algorithm consistently guarantees a line efficiency above 95% in a reasonably short computational time.

Table 5 and Table 6 show the parameters and results of the MINLP and rMILP algorithms for $RP = 2$. Each data set was rerun so that two operators could be assigned to the station. The MINLP and rMILP algorithms were run for 1800 seconds and the results are reported.

Table 5. Parameters used for datasets $RP = 2$

Datasets/ Parameters	C_{min}	C_{max}	K_{max}
P-70 Tongue	156	234	21
P-83 Arcus	3691	5537	20
P-89 Lutz-2	20	35	18
P-94 Mukherje	171	257	24
P-111 Arcus	6615	9923	21
P-148 Bartholdi-1	383	575	12

Table 6. Comparison of MINLP and rMILP for $RP = 2$

Dataset	MINLP					rMILP					
	C	K	OP	LE (%)	Time (s)	C	K	OP	LE (%)	Time (s)	RPD (%)
P-70 Tongue	167	11	21	99.20	1800	206	9	17	99.34	59.55	0.14
P-83 Arcus	5414.5	8	14	99.46	1800	3802	10	20	99.15	220.13	-0.31
P-89 Lutz-3	27	9	18	99.27	1800	27	9	18	99.27	66.04	0.00
P-94 Mukherje	176	13	24	99.51	1800	176	13	24	99.51	199.17	0.00
P-111 Arcus	7145	15	21	97.07	1800	9373	9	17	97.44	892.39	0.38
P-148 Bartholdi-1	470	6	12	99.57	1800	470	6	12	99.57	514.95	0.00

Though the comparison of the results presented in Table 6 with those in Table 4 reveals similarity, the line efficiencies of rMILP and MINLP algorithms are higher when $RP = 2$ compared to the case when $RP = 1$. This is a result of expanded solution space due to the increase in the number of operators that can be assigned to a workstation. An analysis of the line efficiency values presented in Table 6 reveals that the rMILP algorithm consistently yields results that are either equivalent to or superior to those of the MINLP algorithm except one case, i.e. P-83 Arcus. Although the line efficiency found by the rMILP algorithm is 0.3% less than that by the MINLP algorithm for P-83 Arcus, the solution speed of rMILP is 87.77% better. In this instance, once more, the discrepancy in the line efficiencies appears to be tolerable, given the substantial enhancement in the time required to ascertain the solution. For the two test cases, i.e. P-70 Tongue and P-111 Arcus, rMILP obtains better solutions than MINLP within considerably shorter times. Consequently, as the problem becomes more complex, the advantages of the proposed algorithm become more apparent. Another advantage of the proposed rMILP algorithm is the capacity to furnish planners with greater flexibility by offering solutions that encompass a variety of cycle time and station number combinations. These advantages can also be extended with the ease of implementation and maintenance as compared to other methods especially metaheuristics.

5. Discussion and conclusions

This study deals with the mixed-model type-E assembly line rebalancing problem while restricting the total number of task assignment relocations to a certain value. By this restriction, decision makers are capable of utilising a desired line balancing solution based on their choice of task relocation limitations. For this purpose, a mixed integer nonlinear programming model is first developed. Due to the complex structure of the problem, the mathematical model developed cannot obtain a good solution in large data sets in a reasonable time. Therefore, an iterative heuristic algorithm (rMILP) is integrated into the developed model. With the proposed algorithm, the objective function of the type-E problem is linearised by giving minimum and maximum bounds to the cycle time. The developed model is run iteratively for values in the range of minimum and maximum cycle times, providing flexibility to the decision maker. Starting from the given minimum cycle time, the model is run iteratively until it reaches the maximum cycle time. At the same time, in this study, the use of multi-operator stations is also included in the problem of assigning tasks to stations. The proposed approach has been tested with large data sets of different task numbers and it effectively provided competitive solutions in a reasonable time. Therefore, this study provides decision makers with advantages such as the ability to respond quickly and flexibly to evolving markets and special demands, efficient use of assembly lines, thereby increasing customer satisfaction.

The main limitation of the proposed algorithm might be determining the value of step size especially when the difference between C_{min} and C_{max} is large. Since running the algorithm for every possible cycle time value requires excessive computational times, step size could be increased. However, increasing the step size may cause missing the cycle time that might give the optimal line efficiency, as not all values between C_{min} and C_{max} will be checked.

In future work, our model can be extended for various line layouts, including U-shaped, two-sided, parallel and robotic lines. The model can also be tested with real-world datasets by considering more complex scenarios involving hundreds of jobs. In order to make the use of our algorithm more effective and efficient on different datasets, machine learning algorithms can be integrated to learn which step size, C_{step}, might give better results for the specific dataset, taking into account the size of the dataset and the cycle times. The algorithm can be improved by adding different constraints such as zoning constraints and comparisons can be made. At the same time, meta-heuristics can also be incorporated to solve larger-scale problems.

Acknowledgments

None.

Funding

None.

Conflict of interest

There is no conflict of interest to disclose.

Author contributions

Damla Camli: Conceptualization, Methodology, Software, Investigation, Validation, Writing - original draft, Writing - review & editing. **Ibrahim Kucukkoc:** Conceptualization, Methodology, Investigation, Supervision, Writing - original draft, Writing - review & editing. **Zixiang Li:** Conceptualization, Methodology, Supervision, Writing - original draft, Writing - review & editing.

Declaration of using AI tools

The authors declare that they have not used any type of generative artificial intelligence for the writing of this manuscript, nor for the creation of images, graphics, tables, or their corresponding captions.

References

[1] I. Baybars, "A Survey of Exact Algorithms for the Simple Assembly Line Balancing Problem," (in English), *Management Science,* vol. 32, no. 8, pp. 909-932, 1986.

[2] I. Kucukkoc and D. Z. Zhang, "Mathematical Model and Agent Based Solution Approach for the Simultaneous Balancing and Sequencing of Mixed-Model Parallel Two-Sided Assembly Lines," *International Journal of Production Economics,* vol. 158, pp. 314-333, 2014, doi: http://dx.doi.org/10.1016/j.ijpe.2014.08.010.

[3] O. Battaïa and A. Dolgui, "A taxonomy of line balancing problems and their solution approaches," *International Journal of Production Economics,* vol. 142, no. 2, pp. 259-277, 2013.

[4] I. Kucukkoc and D. Z. Zhang, "Mixed-model parallel two-sided assembly line balancing problem: A flexible agent-based ant colony optimization approach," *Computers & Industrial Engineering,* vol. 97, pp. 58-72, 2016, doi: 10.1016/j.cie.2016.04.001.

[5] O. Yilmaz, N. Aydin, and I. Kucukkoc, "Minimization of Manufacturing Cost in Mixed-model Robotic Assembly Lines," in *The Third International Conference on Applied Mathematics in Engineering (ICAMΣ'24)*, Ayvalik, Balikesir, Turkiye, 2024, p. 141.

[6] N. Boysen, P. Schulze, and A. Scholl, "Assembly line balancing: What happened in the last fifteen years?," *European Journal of Operational Research,* vol. 301, no. 3, pp. 797-814, 2022/09/16/ 2022, doi: https://doi.org/10.1016/j.ejor.2021.11.043.

[7] N. Boysen, M. Fliedner, and A. Scholl, "A classification of assembly line balancing problems," (in English), *European Journal of Operational Research,* vol. 183, no. 2, pp. 674-693, Dec 1 2007, doi: DOI 10.1016/j.ejor.2006.10.010.

[8] O. Battaïa and A. Dolgui, "Hybridizations in line balancing problems: A comprehensive review on new trends and formulations," *International Journal of Production Economics,* Article vol. 250, 2022, Art no. 108673, doi: 10.1016/j.ijpe.2022.108673.

[9] Z. Li, I. Kucukkoc, and Q. Tang, "A comparative study of exact methods for the simple assembly line balancing problem," *Soft Computing,* vol. 24, no. 15, pp. 11459-11475, 2020/08/01 2020, doi: 10.1007/s00500-019-04609-9.

[10] N. T. Thomopoulos, "Line Balancing-Sequencing for Mixed-Model Assembly," *Management Science,* vol. 14, no. 2, pp. B-59-B-75, 1967, doi: 10.1287/mnsc.14.2.B59.

[11] N. T. Thomopoulos, "Mixed Model Line Balancing with Smoothed Station Assignments," *Management Science,* vol. 16, no. 9, pp. 593-603, 1970/05/01 1970, doi: 10.1287/mnsc.16.9.593.

[12] H. Gokcen and E. Erel, "A goal programming approach to mixed-model assembly line balancing problem," *International Journal of Production Economics,* vol. 48, no. 2, pp. 177-185, 1997/01/24/ 1997, doi: https://doi.org/10.1016/S0925-5273(96)00069-2.

[13] S. Emde, N. Boysen, and A. Scholl, "Balancing mixed-model assembly lines: a computational evaluation of objectives to smoothen workload," (in English), *International Journal of Production Research,* vol. 48, no. 11, pp. 3173-3191, 2010, doi: Doi 10.1080/00207540902810577.

[14] L. Belkharroubi and K. Yahyaoui, "A Hybrid approach for the Mixed-Model Assembly Line Balancing problem Type-II," in *2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, 22-25 Sept. 2021 2021, vol. 1, pp. 329-332, doi: 10.1109/IDAACS53288.2021.9661039.

[15] L. Belkharroubi and K. Yahyaoui, "Solving the mixed-model assembly line balancing problem type-I using a Hybrid Reactive GRASP," *Production & Manufacturing Research,* vol. 10, no. 1, pp. 108-131, 2022/12/31 2022, doi: 10.1080/21693277.2022.2065380.

[16] T. Mönch, A. Huchzermeier, and P. Bebersdorf, "Variable takt times in mixed-model assembly line balancing with random customisation," *International Journal of Production Research,* vol. 59, no. 15, pp. 4670-4689, 2021/08/03 2021, doi: 10.1080/00207543.2020.1769874.

[17] K. Meng, Q. Tang, L. Cheng, and Z. Zhang, "Mixed-model assembly line balancing problem considering preventive maintenance scenarios: MILP model and cooperative co-evolutionary algorithm," *Applied Soft Computing,* vol. 127, p. 109341, 2022/09/01/ 2022, doi: https://doi.org/10.1016/j.asoc.2022.109341.

[18] Z. Zhang, M. Chica, Q. Tang, Z. Li, and L. Zhang, "A multi-objective co-evolutionary algorithm for energy and cost-oriented mixed-model assembly line balancing with multi-skilled workers," *Expert Systems with Applications,* vol. 236, p. 121221, 2024/02/01/ 2024, doi: https://doi.org/10.1016/j.eswa.2023.121221.

[19] A. Nourmohammadi, A. H. C. Ng, M. Fathi, J. Vollebregt, and L. Hanson, "Multi-objective optimization of mixed-model assembly lines incorporating musculoskeletal risks assessment using digital human modeling," *CIRP Journal of Manufacturing Science and Technology,* vol. 47, pp. 71-85, 2023/12/01/ 2023, doi: https://doi.org/10.1016/j.cirpj.2023.09.002.

[20] Y. Li, D. Liu, and I. Kucukkoc, "Mixed-model assembly line balancing problem considering learning effect and uncertain demand," *Journal of Computational and Applied Mathematics,* vol. 422, p. 114823, 2023/04/01/ 2023, doi: https://doi.org/10.1016/j.cam.2022.114823.

[21] Y. Delice, E. K. Aydoğan, S. Himmetoğlu, and U. Özcan, "Integrated mixed-model assembly line balancing and parts feeding with supermarkets," *CIRP Journal of Manufacturing Science and Technology,* vol. 41, pp. 1-18, 2023/04/01/ 2023, doi: https://doi.org/10.1016/j.cirpj.2022.12.002.

[22] M. D. Mura and G. Dini, "Improving ergonomics in mixed-model assembly lines balancing noise exposure and energy expenditure," *CIRP Journal of Manufacturing Science and Technology,* vol. 40, pp. 44-52, 2023/02/01/ 2023, doi: https://doi.org/10.1016/j.cirpj.2022.11.005.

[23] Z. A. Çil, Z. Li, S. Mete, and E. Özceylan, "Mathematical model and bee algorithms for mixed-model assembly line balancing problem with physical human–robot collaboration," *Applied Soft Computing,* vol. 93, p. 106394, 2020/08/01/ 2020, doi: https://doi.org/10.1016/j.asoc.2020.106394.

[24] H. M. Alakaş and B. Toklu, "Problem Specific Variable Selection Rules for Constraint Programming: A Type II Mixed Model Assembly Line Balancing Problem Case," *Applied Artificial Intelligence,* vol. 34, no. 7, pp. 564-584, 2020/06/06 2020, doi: 10.1080/08839514.2020.1731782.

[25] L. Yuwei, C. Jian'e, Z. Feng, and Z. Shijing, "A Research of Multi-Constrained Two-sided Mixed-model Assembly Line Balancing Problem Based on Genetic Algorithm," in *2020 IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA)*, 16-21 April 2020 2020, pp. 377-381, doi: 10.1109/ICIEA49774.2020.9102104.

[26] S. Yaphiar, C. Nugraha, and A. Ma'ruf, "Mixed Model Assembly Line Balancing for Human-Robot Shared Tasks," in *iMEC-APCOMS 2019*, Singapore, M. N. Osman Zahid, R. Abd. Aziz, A. R. Yusoff, N. Mat Yahya, F. Abdul Aziz, and M. Yazid Abu, Eds., 2020// 2020: Springer Singapore, pp. 245-252.

[27] M. Lalaoui and A. E. Afia, "A versatile generalized simulated annealing using type-2 fuzzy controller for the mixed-model assembly line balancing problem," *IFAC-PapersOnLine,* vol. 52, no. 13, pp. 2804-2809, 2019/01/01/ 2019, doi: https://doi.org/10.1016/j.ifacol.2019.11.633.

[28] P. Sadeghi, R. D. Rebelo, and J. S. Ferreira, "Balancing mixed-model assembly systems in the footwear industry with a variable neighbourhood descent method," *Computers & Industrial Engineering,* vol. 121, pp. 161-176, 2018/07/01/ 2018, doi: https://doi.org/10.1016/j.cie.2018.05.020.

[29] B. Naderi, A. Azab, and K. Borooshan, "A realistic multi-manned five-sided mixed-model assembly line balancing and scheduling problem with moving workers and limited workspace," *International Journal of Production Research,* vol. 57, no. 3, pp. 643-661, 2019/02/01 2019, doi: 10.1080/00207543.2018.1476786.

[30] I. Kucukkoc, Z. Li, A. D. Karaoglan, and D. Z. Zhang, "Balancing of mixed-model two-sided assembly lines with underground workstations: A mathematical model and ant colony optimization algorithm," *International Journal of Production Economics,* vol. 205, pp. 228-243, 2018/11/01/ 2018, doi: https://doi.org/10.1016/j.ijpe.2018.08.009.

[31] F. Tanhaie, M. Rabbani, and N. Manavizadeh, "Simultaneous balancing and worker assignment problem for mixed-model assembly lines in a make-to-order environment considering control points and assignment restrictions," *Journal of Modelling in Management,* vol. 15, no. 1, pp. 1-34, 2019/09/23/ 2019, doi: https://doi.org/10.1108/JM2-04-2018-0052.

[32] T. T. M. Anh and N. Van Hop, "Matheuristics for mixed-model assembly line balancing problem with fuzzy stochastic processing time," *Applied Soft Computing,* vol. 160, p. 111694, 2024/07/01/ 2024, doi: https://doi.org/10.1016/j.asoc.2024.111694.

[33] D. Schibelbain, T. C. Lopes, and L. Magatão, "A method to balancing robotic mixed-model assembly lines: Practical constraints, computational challenges, and performance estimation," *Computers & Industrial Engineering,* vol. 197, p. 110595, 2024/11/01/ 2024, doi: https://doi.org/10.1016/j.cie.2024.110595.

[34] C. G. S. Sikora, "Balancing mixed-model assembly lines for random sequences," *European Journal of Operational Research,* vol. 314, no. 2, pp. 597-611, 2024/04/16/ 2024, doi: https://doi.org/10.1016/j.ejor.2023.10.008.

[35] X. Zhang *et al.*, "Human-Robot Collaboration in Mixed-Flow Assembly Line Balancing under Uncertainty: An Efficient Discrete Bees Algorithm," *Journal of Industrial Information Integration,* vol. 41, p. 100676, 2024/09/01/ 2024, doi: https://doi.org/10.1016/j.jii.2024.100676.

[36] S. El Machouti, M. Hlyal, A. Babay, and J. E. Alami, "Optimisation of Simple Assembly Line Balancing Problem Type E: A Systematic Literature Review," *Management Systems in Production Engineering,* vol. 32, no. 2, pp. 162-173, 2024, doi: 10.2478/mspe-2024-0016.

[37] J. Plans and A. Corominas, "Modelling and solving the SALB-E problem," in *Proceedings of the 1999 IEEE International Symposium on Assembly and Task Planning (ISATP'99) (Cat. No.99TH8470)*, 24-24 July 1999 1999, pp. 356-360, doi: 10.1109/ISATP.1999.782984.

[38] N.-C. Wei and I. M. Chao, "A solution procedure for type E simple assembly line balancing problem," *Computers & Industrial Engineering,* vol. 61, no. 3, pp. 824-830, 2011/10/01/ 2011, doi: https://doi.org/10.1016/j.cie.2011.05.015.

[39] P. T. Zacharia and A. C. Nearchou, "A meta-heuristic algorithm for the fuzzy assembly line balancing type-E problem," *Computers & Operations Research,* vol. 40, no. 12, pp. 3033-3044, 2013/12/01/ 2013, doi: https://doi.org/10.1016/j.cor.2013.07.012.

[40] R. Esmaeilbeigi, B. Naderi, and P. Charkhgard, "The type E simple assembly line balancing problem: A mixed integer linear programming formulation," *Computers & Operations Research,* vol. 64, pp. 168-177, 2015/12/01/ 2015, doi: https://doi.org/10.1016/j.cor.2015.05.017.

[41] M. Jusop and M. F. F. Ab Rashid, "Optimisation of Assembly Line Balancing Type-E with Resource Constraints Using NSGA-II," *Key Engineering Materials,* vol. 701, pp. 195-199, 2016, doi: 10.4028/www.scientific.net/KEM.701.195.

[42] M. N. A. Khalid, U. K. Yusof, and H. Iida, "An Artificial Immune System Approach for Solving Type-E Assembly Line Balancing Problem with Problem-Specific Information," in *2020 IEEE 16th International Conference on Control & Automation (ICCA)*, 9-11 Oct. 2020 2020, pp. 1590-1595, doi: 10.1109/ICCA51439.2020.9264382.

[43] N. Manavizadeh, M. Rabbani, D. Moshtaghi, and F. Jolai, "Mixed-model assembly line balancing in the make-to-order and stochastic environment using multi-objective evolutionary algorithms," (in English), *Expert Systems with Applications,* vol. 39, no. 15, pp. 12026-12031, Nov 1 2012, doi: DOI 10.1016/j.eswa.2012.03.044.

[44] P. Su, N. Wu, and Z. Yu, "A Petri net-based heuristic for mixed-model assembly line balancing problem of Type-E," *International Journal of Production Research,* vol. 52, no. 5, pp. 1542-1556, 2014/03/04 2014, doi: 10.1080/00207543.2013.849010.

[45] A. Legesse, E. Tesfaye, and E. Berhan, "Multi-objective optimization of mixed model assembly line balancing in an assemble-to-order industry with stochastic environment," *International Journal of Engineering, Science and Technology,* vol. 12, no. 3, pp. 90-107, 2020.

[46] G. Fan, H. Zheng, Z. Jiang, J. Liu, and S. Lou, "Multi-objective assembly line rebalancing problem based on complexity measurement in green manufacturing," *Engineering Applications of Artificial Intelligence,* vol. 132, p. 107884, 2024/06/01/ 2024, doi: https://doi.org/10.1016/j.engappai.2024.107884.

[47] T. Çimen, A. Baykasoğlu, and S. Demirkol Akyol, "A detailed review and analysis of assembly line rebalancing problems," *Assembly Automation,* vol. 42, no. 6, pp. 742-760, 2022.

[48] K. Ağpak, "An approach to find task sequence for re-balancing of assembly lines," *Assembly Automation,* vol. 30, no. 4, pp. 378-387, 2010, doi: 10.1108/01445151011075834.

[49] C. Yang, J. Gao, and L. Sun, "A multi-objective genetic algorithm for mixed-model assembly line rebalancing," *Computers & Industrial Engineering,* vol. 65, no. 1, pp. 109-116, 2013/05/01/ 2013, doi: https://doi.org/10.1016/j.cie.2011.11.033.

[50] F. Serin, S. Mete, and E. Çelik, "An efficient algorithm for U-type assembly line re-balancing problem with stochastic task times," *Assembly Automation,* vol. 39, no. 4, pp. 581-595, 2019, doi: 10.1108/AA-07-2018-106.

[51] F. S. Oliveira, K. Vittori, R. M. O. Russel, and X. L. Travassos, "Mixed assembly line rebalancing: A binary integer approach applied to real world problems in the automotive industry," *International Journal of Automotive Technology,* vol. 13, no. 6, pp. 933-940, 2012/10/01 2012, doi: 10.1007/s12239-012-0094-4.

[52] F. Makssoud, D. Y. C. D. Maurizio Faccio, O. Battaïa, A. Dolgui, K. Mpofu, and O. Olabanji, "Re-balancing problem for assembly lines: new mathematical model and exact solution method," (in english), *Emerald,* 2015 2015.

[53] I. Kucukkoc and D. Z. Zhang, "*Type-E Parallel Two-Sided Assembly Line Balancing Problem: Mathematical Model and Ant Colony Optimisation based Approach with Optimised Parameters*," *Computers & Industrial Engineering,* vol. DOI: 10.1016/j.cie.2014.12.037, 2015, doi: 10.1016/j.cie.2014.12.037.

[54] D. Camli and I. Kucukkoc, "Rebalancing of multi-manned assembly lines with model variability," in *The Third International Conference on Applied Mathematics in Engineering (ICAMΣ'24)*, Ayvalik, Balikesir, Turkiye, 2024, p. 125.

[55] Y. Haimes, "On a bicriterion formulation of the problems of integrated system identification and system optimization," *IEEE transactions on systems, man, and cybernetics,* no. 3, pp. 296-297, 1971.

[56] I. Kucukkoc, S. Finco, M. Peron, and G. Aydin Keskin, "Including mechanical requirements in a bi-objective nesting and scheduling model for additive manufacturing," *European Journal of Operational Research,* vol. 325, no. 3, pp. 416-432, 2025/09/16/ 2025, doi: https://doi.org/10.1016/j.ejor.2025.03.022.

[57] O. Yilmaz, N. Aydin, and I. Kucukkoc, "Bi-objective optimization of human-robot collaborative mixed-model multiple assembly lines considering model assignment and energy consumption," *Journal of Computational and Applied Mathematics,* vol. 473, p. 116876, 2026/02/01/ 2026, doi: https://doi.org/10.1016/j.cam.2025.116876.

[58] D. Çamlı, "Karışık modelli montaj hatlarında hat etkinliğinin maksimizasyonu ve bir uygulama," MSc Thesis, Balikesir University (Turkey), 2024.

[59] N. Boysen, M. Fliedner, R. Klein, and A. Scholl. "Assembly Line Balancing Problem Benchmark Data Sets." http://assembly-line-balancing.de/ (accessed 16 Dec 2023).