Flexible Services and Manufacturing Journal
https://doi.org/10.1007/s10696-025-09635-4

Integrated model assignment and multi-line balancing in human–robot collaborative mixed-model assembly lines

Oktay Yilmaz[1] · Nezir Aydin[1,2] · Ibrahim Kucukkoc[3]

Accepted: 19 August 2025

Abstract

The adoption of collaborative robots in assembly lines is increasingly widespread due to their ability to work alongside human operators. This study introduces a simultaneous model-line assignment and robotic mixed-model multiple assembly line balancing (MLA-RMMALB) problem, formulated as a multi-objective mathematical model. The objectives are to (i) minimize production costs, (ii) reduce the energy consumption of cobots, and (iii) evenly distribute the physical workload among operators. A numerical example with 21 tasks is used to analyze the impact of these objectives. The example is first solved separately for each objective and then as a combined multi-objective function. A numerical example with 21 tasks is used to evaluate the effects of the objective functions, first solving each separately and then as a multi-objective function. The results indicate that the model provides high-quality solutions for small instances, with solution times directly influenced by the type of objectives and the problem size. Model assignments, the number of active lines and workstations, and the allocation of operators and cobots vary significantly depending on the objective function. Notably, all models can be produced on a single line with a slight increase in cycle time. To assess the model's performance, benchmark problems with increasing task numbers are analyzed. Findings reveal that due to the problem's complexity -especially under strict cycle times and limited CPU capacity- optimal solutions are not always attainable even for small instances, and no feasible solutions may emerge for medium-sized problems. The proposed approach offers valuable managerial insights by enabling decision-makers to simultaneously optimize cost, energy consumption, and workload distribution. Furthermore, the effective assignment and scheduling of heterogeneous operators and cobots enhance production flexibility and resource utilization. Consequently, this study contributes to the literature by integrating model-line assignment with mixed-model robotic assembly line balancing while considering resource heterogeneity, limited resource availability, and collaboration between cobots and operators.

Extended author information available on the last page of the article

Published online: 09 September 2025

Keywords Multiple assembly lines · Model-line assignment · Human-robot collaboration · Mixed-model · Mixed integer linear programming · Cobots

1 Introduction

The problem of optimal planning for flow-oriented mass production activities with specific rules is an important concept that has been used since ancient times and remains relevant (Li et al. 2024). A series of workstations connected by conveyor belts form assembly lines, which have been used for many years for this purpose (Kucukkoc and Zhang 2016; Ozcan et al. 2010).

Assembly lines are used for producing a wide range of products, from the smallest mass-produced items to large-scale products such as cars and airplanes (Abdous et al. 2023; Li et al. 2023a, b). Problems related to obtaining the best solution by assigning the tasks that must be done in the production process to workstations under certain constraints, considering precedence relationships, are called Assembly Line Balancing Problems (ALBP).

Assembly line balancing problems were first mathematically modeled by Salveson (1955) through the assignment of tasks to workstations. Following this study, numerous studies on assembly lines, including approaches such as exact solution methods, heuristics and meta-heuristics, have been added to the literature (Aguilar et al. 2020; Meng et al. 2023). There are various models and solution approaches in the literature to address this problem. These models and approaches are categorized according to characteristics such as industrial aim, product variety, line type, and task duration (Battaïa and Dolgui 2013; Li et al. 2023a, b).

With the rapid development of technology, the use of robots in production is becoming increasingly widespread and has become an indispensable component of today's smart manufacturing systems (Ponnambalam et al. 2023). Robots contribute to mitigating certain drawbacks, such as human-induced flexibility, safety and sensitivity in production (Soysal Kurt et al. 2024). However, robots have disadvantages, such as energy consumption and environmental impact. Robots, especially collaborative robots, can simultaneously work alongside humans on the same assembly line. Additionally, robots vary according to their technical features and capabilities. It is becoming easier for manufacturing companies to choose robots, especially to reduce uncertainty in task durations and gain a competitive advantage (Sahin and Tural 2023).

Although there are many studies on the assembly line balancing problem in the literature, there are few studies on the multiple assembly line balancing problem. In most of these studies, it is predetermined which model will be produced on which line. In other words, these studies do not address model-line assignment problems, and mixed models are not produced on the lines. In addition to this gap in the literature, the objectives of the problem have begun to diversify, especially with technological advancements and the increasing use of robots in assembly lines. Objectives such as minimizing energy consumption, balancing workstation workloads, reducing total idle time, lowering costs, ensuring ergonomic balance at workstations, and optimizing line efficiency have gained prominence. These gaps in the literature serve as the motivation for this study, which aims to address the model-line assignment problem and the assembly line balancing problem simultaneously while solving this complex problem with multiple objective functions.

In this paper, decisions are made regarding which mixed models will be produced on which assembly line and these multiple lines are balanced in accordance with the defined objectives. In other words, the model-line assignment problem and the multiple assembly line balancing problem in the literature are jointly addressed. The objectives of the problem are (i) minimizing the total cost, including line and workstation opening costs and cobot (collaborative robot) and operator costs, (ii) reducing the energy consumption of cobots, and (iii) distributing the physical workload evenly among operators.

The contributions of this study are summarized as follows:

- The model-line assignment problem and multiple assembly line balancing problem are solved simultaneously.
- Different product models are produced simultaneously on multiple independent assembly lines.
- The resources (operators and cobots) are limited and heterogeneous. Not all resource types can perform every task, and the time required to complete a task varies depending on resource capabilities.
- Each workstation allows one operator and one cobot to work in parallel. Task scheduling is conducted based on start and end times to ensure compliance with the precedence diagram.
- The problem is solved using a multi-objective approach, simultaneously optimizing (i) production cost, (ii) cobot energy consumption, and (iii) physical workload distribution among operators.

The rest of the paper is organized as follows. Section 2 reviews the related literature, considering relevant studies. Section 3 presents the mathematical model, initially focusing on cost minimization. Subsequently, the second objective function (minimizing cobot energy consumption) and the third (distributing the physical load among operators) are incorporated, with changes to the sets, indices and constraints explained. Section 4 presents an illustrative example, experimental tests, and their results. Finally, Sect. 5 concludes the work and outlines future research directions.

2 Literature review

Assembly lines are widely used in flow-oriented production systems designed to produce high-quality, low-cost, and standardized homogeneous products (Kucukkoc and Zhang 2014, 2015). In determining the subject, the "Future Directions" sections of survey papers were first examined to investigate gaps in the literature (Aguilar et al. 2020; Battaïa and Dolgui 2013; Boysen et al. 2022; Chutima 2023; Kheirabadi et al. 2022; Lusa 2008; Otto and Battaïa, 2017; Sivasankaran and Shahabudeen 2014; Tasan and Tunali 2008). Aguilar et al. (2020) highlight several research gaps in their "Future Directions" section. These include the need for more studies on assigning models to production lines, especially for mixed models, and the necessity of increasing research on heterogeneous operators and resource constraints such as robots and equipment. They also call for more work on energy efficiency, ergonomic considerations, and semi-automatic workstations that involve collaboration between cobots and operators.

By taking advantage of the gaps in the literature mentioned above, this study focuses on the model-line assignment problem and mixed-model multiple robotic assembly line balancing problem, aiming to produce mixed models on a fixed number of assembly lines. No study has been found in the literature where more than one type of model is produced on multiple lines, the decision of which models to assign to each line is made by the algorithm, and mixed models are produced on each assembly line.

Cobots are preferred because they can work alongside operators and increase system efficiency without reducing flexibility. One of the most significant advantages of cobot usage is its potential to improve operators' ergonomic conditions. However, the use of cobots also introduces challenges, such as resource selection, task planning (scheduling), and task assignment (Chutima 2023; Kheirabadi et al. 2022).

When studies in the literature involving cobots are examined, Levitin et al. (2006) minimized cycle time by assigning a limited number of heterogeneous robots with different capabilities to workstations, which contained only robots. Similarly, Gao et al. (2009) reconfigured a single-model production line using only robots, aiming to minimize cycle time. Unlike Levitin et al. (2006), the robots in their study were homogeneous. Weckenborg and Spengler (2019) stated that cobots reduce production costs and focus on minimizing production costs based on cycle time. In this study, the cost data from Weckenborg and Spengler (2019) is used to optimize the total cost, including line and station opening costs as well as cobot and operator costs (obj_1). Weckenborg et al. (2020) addressed a single-model assembly line using a homogeneous and limited number of cobots. Their findings indicate that cobots enhance production efficiency but complicate operations due to their collaboration with operators at the same station. Due to the same reason, the model presented in this study becomes more complex with the scheduling problem. Sikora and Weckenborg (2023) extended the work of Weckenborg et al. (2020) by incorporating a Benders decomposition algorithm, allowing for a comparative analysis of their results. Cil et al. (2020) proposed a model aimed at minimizing the cycle time of each model. Their approach assigned heterogeneous cobots to the same workstation as operators, although cobots and operators could not work in parallel, nor could tasks be divided. Boschetti et al. (2021) presented a model that aims for optimal task assignment to resources. Task times may vary depending on the resource type; not every resource can perform every task. Li et al. (2021) introduced a multi-objective model that minimized cycle time and cobot purchasing costs, incorporating cobot alternatives with different purchasing expenses. Similar to Li et al. (2021), Nourmohammadi et al. (2022) examined a single-model straight assembly line balancing problem. However, unlike Li et al. (2021), their approach simultaneously minimized cycle time and the number of resources, including operators and cobots.

Zacharia et al. (2024) considered stochastic task durations for operators, using a fuzzy inference method to calculate task times based on a fitness function. Mao et al.

(2023) focused on a single-model U-type assembly line, incorporating homogeneous operators and cobots who could be assigned to the same workstation. A key distinction of their study is that the assigned operator and cobot could work both in parallel and cooperatively. Keshvarparast et al. (2023) addressed the production of mixed models across multiple straight lines, aiming to minimize the cycle time of each line. However, in their study, the assignment of models to lines was predetermined rather than optimized.

One of the new problem areas that arise with the use of cobots in assembly lines is minimizing energy consumption costs (Chutima 2023). Several studies in the literature have explored energy costs associated with cobots. Ab Rashid et al. (2022) proposed a single-model straight assembly line that balances workstation loads while minimizing cobot energy consumption. They assumed idle energy consumption to be 10% of the energy consumed during task execution, a widely adopted assumption in the literature. Belkharroubi and Yahyaoui (2022) investigated a mixed-model straight robotic assembly line exclusively utilizing heterogeneous robots. Chutima and Khotsaenlee (2022) proposed a mixed-model parallel U-type assembly line, incorporating homogeneous cobots alongside multiple operator types. Their model simultaneously optimized five objectives, including cobot energy consumption. This study also adopts the energy consumption model from Chutima and Khotsaenlee (2022) to minimize cobot energy consumption (obj_2).

In the study of Soysal-Kurt et al. (2024), the energy consumption of cobots consists of energy consumption in working and standby modes, similar to this study. Huang et al. (2024) proposed a mixed-model, two-sided straight assembly line setup to minimize ergonomic risk, energy consumption, and cycle time. Their study is one of the most comprehensive in the literature, offering high-quality solutions by simultaneously focusing on production efficiency, energy consumption cost, and ergonomic risks.

Factors such as the repetitive nature of the work, the frequent need for force application, and static postures expose operators on assembly lines to the risk of musculoskeletal system disorders. Although ergonomics may initially seem solely related to operator health, it is also an important factor for companies due to its effects on operator performance, line stoppages or slowdowns, and decreased production quality and productivity (Lovato et al. 2024). To measure the physical demands and environmental factors of tasks, various methods exist, including OSHA (Occupational Safety and Health Administration), RULA (Rapid Upper Limb Assessment), OCRA (Occupational Repetitive Action), NIOSH (National Institute for Occupational Safety and Health), REBA (Rapid Entire Body Assessment), PMES (Predetermined Motion Energy System). In mathematical models, ergonomics is incorporated into objective functions to balance operator workloads and is controlled by constraints to ensure ergonomic limits are not exceeded for a given period or task (Otto and Battaïa 2017).

Regardless of whether they involve robots/cobots, studies on ergonomic considerations in the literature include the work of Kara et al. (2014), who proposed a model that minimizes production costs while incorporating ergonomic and resource constraints. Their model ensured that the total psychological and physical loads of the tasks assigned to an operator did not exceed predefined limits. Battini et al. (2015) proposed both a multi-objective and a single-objective model to balance operators' tasks in terms of time and ergonomics. In their study, the energy consumed by operators during tasks was calculated using formulas from Garg et al. (1978), while the model with a single objective function utilized formulas from Rohmert (1973).

Barathwaj et al. (2015) proposed a multi-objective model that optimized cycle time and ergonomic risk level using the RULA method for ergonomic risk assessment and presenting a genetic algorithm as an alternative solution. Bautista et al. (2016) developed a multi-objective model that optimized ergonomic risk by considering cycle time, required workspace size, and assigned tasks.

Battini et al. (2016) presented a multi-objective model that simultaneously minimized cycle time and balanced physical workload among operators. Their study introduced the PMES method, a relatively new approach in the literature, to calculate operators' physical loads. In this study, the PMES method and the workload distribution model from Battini et al. (2016) are used to distribute the physical load among operators (obj_3). Similar to Battini et al. (2016), Bortolini et al. (2017) proposed a multi-objective model that simultaneously optimized cycle time and physical workload distribution using the REBA method for ergonomic calculations. Zhang et al. (2020) proposed a multi-objective model that optimized both cycle time and ergonomic risk, using OCRA method for a single-model U-type assembly line with heterogeneous operators. Keshvarparast et al. (2022) examined scenarios where operators and cobots can work at the same station. In their study, cobots were homogeneous, whereas operators had varying experience levels and limited physical capacities. Unlike previous studies, this research allows operators and cobots to perform tasks sequentially, simultaneously, and supportively.

Deniz and Ozcelik (2024) proposed a robotic multi-objective model that minimizes operator costs and ergonomic risks associated with tasks. Junior et al. (2023) examined a classical assembly line balancing problem aimed at evenly distributing operators' physical workloads. The OCRA model was used to calculate task-related physical loads. Dalle Mura and Dini (2023) emphasized that ergonomic risks should be assessed not only based on physical workload intensity but also on environmental factors such as noise levels.

Although there are many studies on assembly line balancing, research on multiple assembly line balancing remains limited. In most of these studies, model-to-line assignments are predetermined, eliminating the need for optimization. Among them, the study by Scholl and Boysen (2009) stands out, as it employs an algorithm to determine which model is produced on which line. Their research aimed to produce multiple models on parallel assembly lines, where the algorithm assigns models to lines, but only one model type can be produced per line. Additionally, workstations are shared between lines, but tasks can only be performed on one side of the line. No previous study has simultaneously addressed the production of multiple model types on multiple lines, where model-line assignments are determined by an algorithm and mixed models can be produced within each assembly line.

Considering the literature, this study is distinguished primarily by its model-line assignment approach. Additional contributions include the ability to produce mixed models on each line, the heterogeneity of operators and cobots, their capability to work in parallel at the same station, and the simultaneous optimization of total production cost, cobot energy consumption, and the even distribution of operators' physical workloads. A summary of the literature is presented in Table 1.

3 Problem statement

This study proposes a simultaneous model-line assignment and robotic mixed-model multiple assembly line balancing (MLA-RMMALB) problem to produce mixed-models simultaneously on more than one line while minimizing cost and energy consumption and ensuring ergonomic efficiency. Unlike parallel assembly lines, multiple assembly lines are completely separate, with no common workstations. Cycle times depend on production demands; therefore, changes in production demands directly impact cycle times. Additionally, capacity constraints are enforced since task times depending on both models and resources are considered in the mathematical model. The assumptions of this research are defined as follows:

- The algorithm assigns two or more product models (i.e., mixed models) to more than one assembly line in accordance with the objective functions.
- The workstations are synchronous and utilized lines are paced.
- A common precedence diagram is used for models produced on the same line. This ensures that common tasks between similar models on the same line are not assigned to different workstations or resources (operators and cobots), maximizing resource utilization.
- A limited number of heterogeneous resources are available, and the algorithm assigns these resources to workstations based on the objective functions.
- Not all types of resources can perform all tasks; therefore, a capability matrix defines which resources can execute specific tasks. Additionally, task durations may vary depending on resource capabilities.
- The processing times of tasks are deterministic and known for each resource type.
- Tasks cannot be assigned to more than one workstation or resource. Thus, each task for each model must be assigned to only one resource at one workstation.
- The total time of tasks assigned to a resource at each workstation cannot exceed the defined cycle time.
- Each task can only be assigned and completed after all its prerequisite tasks have been completed, in accordance with precedence constraints.
- At most, one cobot and one operator can be assigned to a workstation, allowing them to work together simultaneously.
- Total cost includes line opening costs, workstation opening costs, cobot costs, and operator costs. Since cobots and operators are heterogeneous, the unit cost of each type varies.

3.1 Mathematical model

In the presented problem, a mixed integer linear programming model is developed with three objective functions: minimizing line costs, reducing cobot energy con-

Table 1 Summary of the literature

Author(s) (Year)	Line layout	Model variety	Model-line assignment	Objective function	Robotic	Energy consumption by robots	Ergonomics
Levitin et al. (2006)	Straight	Single	–	Cycle time	✓	–	–
Gao et al. (2009)	Straight	Single	–	Cycle time	✓	–	–
Scholl and Boysen (2009)	Parallel	Single	✓	Station number	–	–	–
Kara et al. (2014)	Straight	Single	–	Cost	–	–	✓
Battini et al. (2015)	Straight	Single	–	Smoothness index, physical workload among workers	–	–	✓
Barathwaj et al. (2015)	Straight	Mixed	–	Station number, ergonomic risk	–	–	✓
Bautista et al. (2016)	Straight	Mixed	–	Cycle time, space, ergonomic risk	–	–	✓
Battini et al. (2016)	Straight	Mixed	–	Cycle time, energy expenditure	–	–	✓
Bortolini et al. (2017)	Straight	Single	–	Takt time, ergonomic risk	–	–	✓
Weckenborg and Spengler (2019)	Straight	Single	–	Cost	✓	✓	✓
Zhang et al. (2020)	U-Type	Single	–	Cycle time, ergonomic risk	–	–	✓
Weckenborg et al. (2020)	Straight	Single	–	Cycle time	✓	–	–
Cil et al. (2020)	Straight	Mixed	–	Cycle time	✓	–	–
Boschetti et al. (2021)	Straight	Single	–	Makespan	✓	–	–
Li et al. (2021)	Straight	Single	–	Cycle time, cost	✓	–	–
Belkharroubi and Yahyaoui (2022)	Straight	Mixed	–	Energy	✓	✓	–
Ab Rashid et al. (2022)	Straight	Single	–	Smoothness index, energy	✓	–	–
Chutima and Khotsaenlee (2022)	Parallel U-Type	Single	–	Efficiency, energy, energy expenditure among workers, tax deduction	✓	✓	✓
Nourmohammadi et al. (2022)	Straight	Single	–	Cycle time, resource number	✓	–	–
Keshvarparast et al. (2022)	Straight	Single	–	Cycle time, physical workload of operators	–	–	✓
Keshvarparast et al. (2023)	Multi-Straight	Mixed	–	Cycle time	✓	–	–
Sikora and Weckenborg (2023)	Straight	Single	–	Cycle time	✓	–	–
Dalle Mura and Dini (2023)	Straight	Mixed	–	Cost, energy and noise variance among workers	–	–	✓

Table 1 (continued)

Author(s) (Year)	Line layout	Model variety	Model-line assignment	Objective function	Robotic	Energy consumption by robots	Ergonomics
Mao et al. (2023)	U-Type	Single	–	Cycle time	✓	–	–
Junior et al. (2023)	Straight	Single	–	Cycle time, ergonomic risk	–	–	✓
Deniz and Ozcelik (2024)	Straight	Single	–	Cost, hazard to the worker	–	–	✓
Zacharia et al. (2024)	Straight	Single	–	Cycle time, smoothness index	✓	–	–
Soysal-Kurt et al. (2024)	Parallel	Mixed	–	Cycle time, energy	✓	✓	–
Huang et al. (2024)	Straight	Mixed	–	Cycle time, energy, energy expenditure of workers	✓	✓	✓
Our work	Multi-Straight	Mixed	✓	Cost, energy, workload distribution	✓	✓	✓

sumption, and evenly distributing the physical workload among operators. The mathematical model is divided into three sub-models:

- Model-1 focuses only on minimizing line costs.
- Model-2 examines only the energy consumption of cobots. At this stage, additional parameters, decision variables, and constraints are introduced to extend Model-1.
- Model-3 aims to balance the physical workload among operators. The necessary parameters, decision variables, and constraints are added to the previous models.

Before solving the multi-objective function, which combines line costs, cobot energy consumption, and physical workload distribution, Model-1, Model-2, and Model-3 are solved separately. The minimum and maximum values obtained from these solutions are then normalized, after which the multi-objective function is solved. The normalization follows the formula $(obj - obj_{min}) / (obj_{max} - obj_{min})$ (Aydin and Cetinkale 2022; Li et al. 2021).

Due to its ease of use and short solution time, the weighted sum method is applied (Marler and Arora 2010). After normalization, the multi-objective function is formulated as $(obj_{Multi} = w_1 \times obj_1 + w_2 \times obj_2 + w_3 \times obj_3)$. Decision-makers can assign different weight coefficients to each objective function based on specific needs. However, in this study, equal weight coefficients are used for all three objective functions. Figure 1 illustrates a feasible solution for the proposed mathematical model.

In Model-1, only the cost of production lines is considered. Information about workstation cost, cobot cost, and operator cost is taken from the study of Weckenborg and Spengler (2019). Weckenborg and Spengler (2019) generated their data based on certain assumptions in their paper. In their study, the machine life is assumed to be five years, annual working days are calculated as 230 days, and

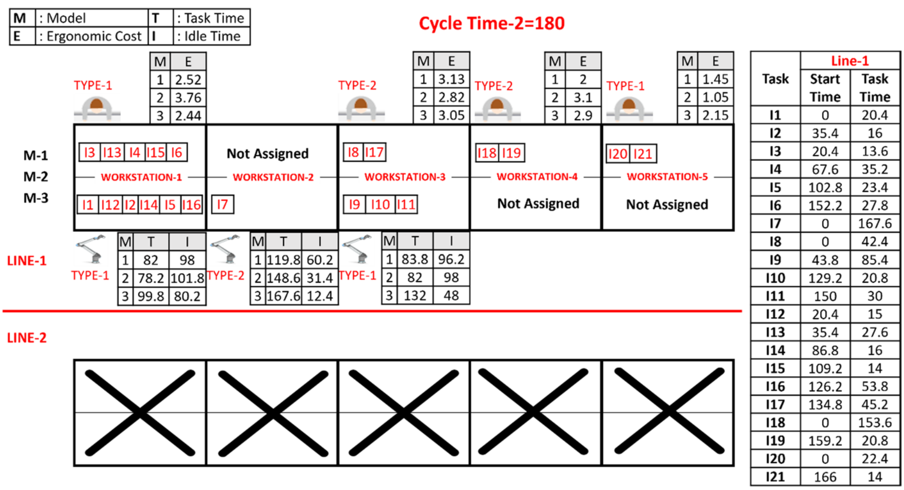


Fig. 1 An illustration of a feasible solution

daily working hours are calculated as eight hours. Thus, the amortization period is $5 \times 230 \times 8 \times 60 = 552{,}000$ min and all costs are calculated in minutes based on this period. The cost of a workstation equipped with the necessary equipment is assumed to be 35,000 euros, with a cost per minute of 35,000/552,000 = 0.06 euro/min. The cost of a cobot, including all its equipment, is 45,000/552,000 = 0.08 euro/min. The operator (labor) cost is assumed to be 35/60 = 0.6 euro/min, based on an hourly wage of 35 euros. As a result, unit operator costs are observed to be 7.5 times higher than unit cobot costs (0.6/0.08). Consequently, the mathematical model will prioritize cobots unless constraints, such as capability limitations or resource shortages, arise.

Moreover, there are multiple types of operators and cobots, and the multiple assembly line problem is studied in addition to the data taken from the study of Weckenborg and Spengler (2019). Therefore, there is a need for alternative operator and cobot costs, as well as the opening cost of an assembly line. These data are derived to be compatible with the data from the study of Weckenborg and Spengler (2019).

In this study, operators and cobots can be assigned to the same workstation and perform different tasks simultaneously. This situation makes it difficult to check compliance with the predecessor-successor relationship while performing the tasks. To overcome this difficulty, for tasks assigned to the same workstation -especially when assigned to different resources- that have a predecessor-successor relationship, it is necessary to ensure that the previous task is completed before the next task starts. Therefore, a decision variable (st_{il}) representing the start times of the tasks is added to the mathematical model to schedule the tasks.

The notation used in Model-1 is presented below, followed by the objective function and constraints utilized.

Model-1:

Sets and Indices	
I	Set of tasks $(i, j \in I)$
W	Set of workstations $(w \in W)$
M	Set of models $(m \in M)$
L	Set of assembly lines $(l \in L)$
H	Set of operators $(h \in H)$
C	Set of cobots $(c \in C)$
R	Set of resources $(R = H \cup C)$ and $(h, c, r \in R)$
Parameters	
cp_{ri}	Capability of resource r to perform task , which equals 1 if resource r can perform task ; 0, otherwise
$tfrs_{irm}$	Required time for resource r to execute task for model m
P_{ij}	Precedence relationship between tasks and j, where $(i, j) \in P$ if task is an immediate predecessor of task j
nor_r	Quantity for each type of resource r
cor_r	Cost per minute for each type of resource r
cos	Cost per minute of opened workstations
col	Cost per minute of opened lines
CT	Cycle time
$BIGM$	A large enough positive number
Decision Variables	

x_{iwlr}	1, if task is assigned to resource r in workstation w on line ; 0, otherwise
y_{ml}	1, if model m is assigned to line ; 0, otherwise
z_{wl}	1, if workstation w on line is opened; 0, otherwise
ln_l	1, line is opened; 0, otherwise
ra_{wlr}	1, if resource type r is assigned to workstation w on line ; 0, otherwise
ssa_{ijwl}	1, if task is before task j in workstation w on line ; 0, otherwise
st_{il}	Start time of task on line

Objective Function

The first objective function given in (1) corresponds to minimizing the total cost of opened lines, workstations and used cobots and operators.

$$obj_1 = \sum_{l \in L} (ln_l \times col \times CT) + \sum_{w \in W} \sum_{l \in L} (z_{wl} \times cos \times CT) + \sum_{w \in W} \sum_{l \in L} \sum_{r \in R} (ra_{wlr} \times cor_r \times CT) \quad (1)$$

Constraints

Constraints (2) and (3) ensure that each task is assigned to one of the open (utilized) assembly lines.

$$\sum_{w \in W} \sum_{r \in R} x_{iwlr} \leq ln_l, \forall i \in I; l \in L \quad (2)$$

$$\sum_{w \in W} \sum_{r \in R} x_{iwlr} \geq 1 - BIGM \times (1 - ln_l), \forall i \in I; l \in L \quad (3)$$

Constraint (4) prevents tasks from being assigned to resources that are incapable of performing them.

$$x_{iwlr} \leq cp_{ri}, \forall i \in I; w \in W; l \in L; r \in R \quad (4)$$

Constraint (5) ensures that each product model is assigned to exactly one assembly line.

$$\sum_{l \in L} y_{ml} = 1, \forall m \in M \quad (5)$$

Constraints (6) and (7) ensure that a line with no model assigned is not opened, whereas a line with at least one model assigned is opened.

$$ln_l \leq \sum_{m \in M} y_{ml}, \forall l \in L \quad (6)$$

$$y_{ml} \leq ln_l, \forall m \in M; l \in L \quad (7)$$

Constraint (8) prevents the opening of workstations to which no resources have been assigned.

$$z_{wl} \leq \sum_{r \in R} ra_{wlr}, \forall w \in W; l \in L \tag{8}$$

Constraints (9) and (10) ensure that, at most, one operator and one cobot are assigned to each workstation on the lines.

$$\sum_{h \in H} ra_{wlh} \leq z_{wl}, \forall w \in W; l \in L \tag{9}$$

$$\sum_{c \in C} ra_{wlc} \leq z_{wl}, \forall w \in W; l \in L \tag{10}$$

Constraint (11) ensures that no resources are present at the workstation unless they are assigned a task suitable for their capabilities.

$$ra_{wlr} \leq \sum_{i \in I} x_{iwlr}, \forall w \in W; l \in L; r \in R \tag{11}$$

Constraint (12) ensures that a task is not assigned to a workstation without at least one suitable resource.

$$\sum_{i \in I} x_{iwlr} \leq ra_{wlr} \times |I|, \forall w \in W; l \in L; r \in R \tag{12}$$

Constraint (13) ensures that the number of resources of each type used across all lines does not exceed the total number available.

$$\sum_{w \in W} \sum_{l \in L} ra_{wlr} \leq nor_r, \forall r \in R \tag{13}$$

Constraint (14) ensures that, for tasks with a precedence relationship, the preceding task is assigned to the same or an earlier workstation than the next task.

$$\sum_{w \in W} \sum_{r \in R} (w \times x_{iwlr} - w \times x_{jwlr}) \leq 0, \forall i, j \in P_{ij}; l \in L \tag{14}$$

Constraints (15) and (16) ensure that lines and workstations are opened, respectively, as needed.

$$z_{w+1l} \leq z_{wl}, \forall w \in W; l \in L \tag{15}$$

$$ln_{l+1} \leq ln_l, \forall l \in L \tag{16}$$

Constraints (17) and (18) ensure that the sum of the task start time and its execution time does not exceed the cycle time. They also prevent assigning start times to tasks on lines that are not opened.

$$st_{il} + \sum_{w \in W} \sum_{r \in R} (x_{iwlr} \times tfrs_{irm}) + BIGM \times (y_{ml} - 1) \leq CT, \forall i \in I; l \in L; m \in M \tag{17}$$

$$st_{il} \leq ln_l \times CT, \forall i \in I; l \in L \tag{18}$$

For tasks with a precedence relationship, constraint (19) ensures that the sum of the start time and execution time of the preceding task is less than or equal to the start time of the other task.

$$\begin{aligned} st_{il} + \sum_{r \in R} (x_{iwlr} \times tfrs_{irm}) + BIGM \times (y_{ml} - 1) + BIGM \times (ssa_{ijwl} - 1) \\ \leq st_{jl}, \forall i, j \in I; w \in W; l \in L; m \in M \end{aligned} \tag{19}$$

Constraint (20) ensures that when tasks with a precedence relationship (P_{ij}) are assigned to the same workstation, a priority-subsequence relationship (ssa_{ijwl}) exists between them, regardless of the resource to which they are assigned.

$$\begin{aligned} 1 - BIGM \times \left(1 - \sum_{r \in R} x_{iwlr}\right) - BIGM \times \left(1 - \sum_{r \in R} x_{jwlr}\right) \\ \leq ssa_{ijwl}, \forall i, j \in P_{ij}; w \in W; l \in L \end{aligned} \tag{20}$$

Constraint (21) guarantees that a priority-subsequence relationship (ssa_{ijwl}) exists between all tasks assigned to the same resource at the same workstation.

$$\begin{aligned} 1 - BIGM \times (1 - x_{iwlr}) - BIGM \times (1 - x_{jwlr}) \\ \leq ssa_{ijwl} + ssa_{jiwl}, \forall i, j \in I, i \neq j; w \in W; l \in L; r \in R \end{aligned} \tag{21}$$

Constraint (22) ensures that no priority-subsequence relationship (ssa_{ijwl}) exists between tasks that are not assigned to the same workstation.

$$ssa_{ijwl} \leq \left(\sum_{r \in R} x_{iwlr} + \sum_{r \in R} x_{jwlr}\right) /2, \forall i, j \in I; w \in W; l \in L \tag{22}$$

Constraints (23)–(29) identify the types of decision variables.

$$x_{iwlr} \in \{0,1\}, \forall i \in I, w \in W, l \in L, r \in R \tag{23}$$

$$y_{ml} \in \{0,1\}, \forall m \in M, l \in L \tag{24}$$

$$z_{wl} \in \{0,1\}, \forall w \in W, l \in L \tag{25}$$

$$ln_l \in \{0,1\}, \forall l \in L \tag{26}$$

$$ra_{wlr} \in \{0{,}1\}, \forall w \in W, l \in L, r \in R \tag{27}$$

$$ssa_{ijwl} \in \{0{,}1\}, \forall i, j \in I, w \in W, l \in L \tag{28}$$

$$st_{il} \geq 0, \forall i \in I, l \in L \tag{29}$$

Model-2:

In Model-2, the objective function focuses solely on minimizing the energy consumed by the cobots assigned to the workstations. The calculation of the energy consumption during the working and standby phases inspired by the work of Chutima and Khotsaenlee (2022). In their study, one of the objective functions is to minimize the energy consumed by cobots. They define the energy consumed during the standby phase as 10% of the energy used during the working phase. This ratio is also applied in this study.

Given the presence of multiple types of cobots in this study, energy consumption is derived according to the type of cobot. The assumption is that cobot types that complete the same task in less time consume more energy per unit of time. Below are the additional parameters, decision variables, and constraints that need to be incorporated into Model-1 to form Model-2.

Additional Parameters	
OPC_c	Energy consumption per minute for each cobot type c during operation
SPC_c	Energy consumption per minute for each cobot type c during standby ($SPC_c = 0{,}1 \times OPC_c$)
PMR	PM2.5 average energy consumption emission coefficient (g/kWh)
s	A sufficiently small positive number
Additional Decision Variables	
TTM_{lwrm}	Total task time for model m of tasks assigned to line workstation w and resource type r
TIM_{lwrm}	Total idle time for model m of tasks assigned to line workstation w resource type r

Objective Function

The second objective function given in (30) minimizes the total energy consumed by different types of cobots assigned to workstations.

$$obj_2 = \sum_{l \in L} \sum_{w \in W} \sum_{c \in C} \sum_{m \in M} (TTM_{lwcm} \times OPC_c \times PMR) + \sum_{l \in L} \sum_{w \in W} \sum_{c \in C} \sum_{m \in M} (TIM_{lwcm} \times SPC_c \times PMR) \tag{30}$$

Additional Constraints

Constraints (31) and (32) determine the total task and idle times of each resource assigned to the lines, for each model.

$$\sum_{i \in I} (x_{iwlr} \times tfrs_{irm}) + BIGM \times (y_{ml} - 1) \leq TTM_{lwrm}, \forall l \in L; w \in W; r \in R; m \in M \tag{31}$$

$$CT - \sum_{i \in I} (x_{iwlr} \times tfrs_{irm}) + BIGM \times (y_{ml} - 1) + BIGM \times (ra_{wlr} - 1) \\ \leq TIM_{lwrm}, \forall l \in L; w \in W; r \in R; m \in M \tag{32}$$

Constraint (33) ensures that the sum of task and idle times does not exceed the cycle time.

$$TTM_{lwrm} + TIM_{lwrm} \leq CT \times ra_{wlr}, \forall l \in L; w \in W; r \in R; m \in M \tag{33}$$

Constraint (34) ensures that no task or idle time is calculated for a model that is not assigned to the line.

$$TTM_{lwrm} + TIM_{lwrm} \leq CT \times y_{ml}, \forall l \in L; w \in W; r \in R; m \in M \tag{34}$$

Constraint (35) ensures that the total time of the tasks assigned to each resource is greater than zero.

$$\sum_{m \in M} TIM_{lwrm} + s \leq \sum_{m \in M} y_{ml} \times CT + BIGM \times (1 - ra_{wlr}), \\ \forall l \in L; w \in W; r \in R \tag{35}$$

Constraints (36) and (37) define the type of decision variables.

$$TTM_{lwrm} \geq 0, \forall l \in L, w \in W, r \in R, m \in M \tag{36}$$

$$TIM_{lwrm} \geq 0, \forall l \in L, w \in W, r \in R, m \in M \tag{37}$$

Model-3:

In Model-3, the objective function is solely concerned with reducing risk among workstations by distributing the physical load as evenly as possible among operators. This model employs the PMES method, which was introduced in the literature and used by Battini et al. (2016) to calculate the physical loads of operators based on their physical characteristics. The PMES method calculates the total energy expenditure of an operator based on movements such as lifting, walking, carrying loads, and arm movements. The operator's physical characteristics and the number of repetitions of these movements are used as input data.

To maintain the model's linearity, additional decision variables are introduced to minimize the difference between the maximum physical load on the line and the physical loads of other workstations. These physical loads are calculated only for workstations assigned to operators and only for models assigned to those lines. The calculations assume that the operator type who completes the task in less time consumes more energy per unit of time.

The parameters, decision variables, and constraints that need to be added to Model-2 to form Model-3 are presented below.

Additional Parameter	
$ecfh_{ihm}$	Physical load of operator h for model m during task
Additional Decision Variables	
TEC_{lwm}	Total physical load of the operator for model m assigned to workstation w on line
BEC_{lm}	The maximum physical load (TEC_{lwm}) calculated for line for model m
DEC_{lwm}	The difference between the highest physical load (BEC_{lm}) calculated for line for model m and the physical loads (TEC_{lwm}) of workstations assigned operators on that line

Objective Function

The third objective function given in (38) seeks to distribute the physical load evenly among operators during task performance.

$$obj_3 = \sum_{l \in L} \sum_{w \in W} \sum_{m \in M} DEC_{lwm} \tag{38}$$

Additional Constraints

Constraints (39)–(41) determine the total physical loads of operators for each model.

$$\sum_{i \in I} \sum_{h \in H} (x_{iwlh} \times ecfh_{ihm}) + BIGM \times (y_{ml} - 1) \leq TEC_{lwm}, \forall l \in L; w \in W; m \in M \tag{39}$$

$$TEC_{lwm} \leq \sum_{i \in I} \sum_{h \in H} (x_{iwlh} \times ecfh_{ihm}), \forall l \in L; w \in W; m \in M \tag{40}$$

$$TEC_{lwm} \leq BIGM \times y_{ml}, \forall l \in L; w \in W; m \in M \tag{41}$$

Constraint (42) determines the maximum of the total physical loads calculated for each line and model.

$$TEC_{lwm} \leq BEC_{lm}, \forall l \in L; w \in W; m \in M \tag{42}$$

Constraints (43)–(45) determine the difference between the largest physical load and the total physical load calculated for each line and model.

$$BEC_{lm} - TEC_{lwm} + BIGM \times \left(\sum_{h \in H} ra_{wlh} - 1 \right) \leq DEC_{lwm}, \forall l \in L; w \in W; m \in M \tag{43}$$

$$DEC_{lwm} \leq BEC_{lm} - TEC_{lwm}, \forall l \in L; w \in W; m \in M \tag{44}$$

$$DEC_{lwm} \leq \sum_{h \in H} BIGM \times ra_{wlh}, \forall l \in L; w \in W; m \in M \tag{45}$$

Constraints (46)–(48) identify the type of decision variables.

$$TEC_{lwm} \geq 0, \forall l \in L, w \in W, m \in M \tag{46}$$

$$BEC_{lm} \geq 0, \forall l \in L, m \in M \tag{47}$$

$$DEC_{lwm} \geq 0, \forall l \in L, w \in W, m \in M \tag{48}$$

4 Experimental study

This section begins with a numerical example, followed by a series of comprehensive experimental tests using the proposed mathematical model developed in this research. In Sect. 4.1, data acquisition and results of the numerical example generated for a detailed explanation of the proposed model are examined. To test the performance of the mathematical model, test problems commonly used in the literature, with a gradually increasing number of tasks, are employed. Section 4.2 analyzes the results obtained.

4.1 Numerical example

The numerical example is solved under two different cycle time constraints (Cycle Time-1 = 170 and Cycle Time-2 = 180 time-units). One of the aims of varying cycle time is to observe the change in the number of assembly lines, workstations and resources used with this relief. Another aim is to test the sensitivity of the mathematical model by examining the effects of changing the objective function on the prioritized resource type and task assignment strategy. The parameters of the numerical example are presented in Table 2.

A sample containing three models and 21 tasks (Barathwaj et al. 2015) is used to create the numerical example. The task times for the three models specified in the sample are considered for Operator Type-1. Since this study has two types of operators and two types of cobots, the task times for other resource types are derived from the times assigned to Operator Type-1. Information about these tasks is provided in Table 3.

Since the resources are heterogeneous, not every resource can perform every task. For this reason, a capability matrix has been created. Additionally, a common prece-

Table 2 Parameters used in the numerical example

Data type	Value
Number of tasks	21
Number of Lines	2
Maximum number of workstations on each line	5
Number of models	3
Number of operators (Type 1)	10
Number of operators (Type 2)	10
Number of cobots (Type 1)	10
Number of cobots (Type 2)	10

Table 3 Task times for resources

Task	Operator Type-1 model (1/2/3)			Operator Type-2 model (1/2/3)			Cobot Type-1 model (1/2/3)			Cobot Type-2 model (1/2/3)		
1	25.4	23.6	9.2	28.4	26.6	12.2	20.4	18.6	4.2	23.4	21.6	7.2
2	21.0	16.4	11.6	24.0	19.4	14.6	16.0	11.4	6.6	19.0	14.4	9.6
3	13.6	0.0	0.0	16.6	0.0	0.0	8.6	0.0	0.0	11.6	0.0	0.0
4	0.0	35.2	30.4	0.0	39.2	33.4	0.0	30.2	25.4	0.0	34.2	28.4
5	21.4	22.2	28.4	24.4	25.2	31.4	16.4	17.2	23.4	19.4	20.2	26.4
6	0.0	6.8	27.8	0.0	9.8	30.8	0.0	1.8	22.8	0.0	4.8	25.8
7	121.8	150.6	169.6	124.8	153.6	172.6	116.8	145.6	164.6	119.8	148.6	167.6
8	32.2	39.4	22.6	35.2	42.4	25.6	27.2	34.4	17.6	30.2	37.4	20.6
9	58.8	59.2	90.4	61.8	62.2	93.4	53.8	54.2	85.4	56.8	57.2	88.4
10	0.0	0.0	25.8	0.0	0.0	28.8	0.0	0.0	20.8	0.0	0.0	23.8
11	35.0	32.8	30.8	38.0	35.8	33.8	30.0	27.8	25.8	33.0	30.8	27.8
12	19.4	20.0	0.0	22.4	23.0	0.0	14.4	15.0	0.0	17.4	18.0	0.0
13	23.2	27.6	0.0	26.2	30.6	0.0	18.2	22.6	0.0	21.2	25.6	0.0
14	19.8	21.0	16.8	22.8	24.0	19.8	14.8	16.0	11.8	17.8	19.0	14.8
15	11.4	14.0	0.0	14.4	17.0	0.0	6.4	9.0	0.0	9.4	12.0	0.0
16	0.0	0.0	58.8	0.0	0.0	62.8	0.0	0.0	53.8	0.0	0.0	57.8
17	42.2	36.0	23.6	45.2	39.0	26.6	37.2	31.0	18.6	40.2	34.0	21.6
18	105.8	149.0	150.6	108.8	152.0	153.6	100.8	144.0	145.6	103.8	147.0	148.6
19	15.2	17.8	14.6	18.2	20.8	17.6	10.2	12.8	9.6	13.2	15.8	12.6
20	17.4	11.8	22.4	20.4	14.8	25.4	12.4	6.8	17.4	15.4	9.8	20.4
21	10.6	10.2	14.0	13.6	13.2	17.0	5.6	5.2	9.0	8.6	8.2	12.0

dence diagram is used to produce the three models. The capability matrix is presented in Table 4 and the common precedence diagram is presented in Table 5.

Data to be used for cost calculation (obj_1) are given in Table 6, while the data to be used in calculating the total energy consumed by the cobots (obj_2) and distributing the physical load among operators (obj_3) are presented in Tables 7 and 8, respectively.

The proposed mathematical model is solved via Python Gurobi 10.0 on an Intel(R) Core (TM) i7-8565U CPU 2.0 GHz with 16 GB RAM, Windows 10, and a 64-bit operating system. The solutions used in the normalization calculation of the multi-objective function, obtained by solving Model-1, Model-2, and Model-3 separately, are presented in Table 9, and the optimal solutions after normalization are presented in Table 10. Additionally, the optimal line-balancing solutions for two different cycle times of the numerical example are shown in Figs. 2 and 3.

The results of the numerical example under two different cycle time conditions are presented above. Upon examining the results of the numerical example:

- Solution times of objective functions are longer for Cycle Time-2 compared to Cycle Time-1. This is likely due to the expansion of the feasible region and the increase in solution alternatives resulting from the longer cycle time.
- For Cycle Time-1, the solutions of obj_1 and obj_{Multi} converge. This is likely because the feasible region and optimal solution alternatives are constrained due to the cycle time being at its limit.
- For Cycle Time-2, the solution of obj_1 requires one fewer operator but two more cobots compared to the solution of obj_{Multi}. Despite the increase in total resources, the overall cost decreases. Since the cost of the cobot is less than the operator cost, the model prioritizes the use of cobots in the solution of obj_1.
- With a roughly 5% increase in cycle time, three models can be produced on a single line for Cycle Time-2. As a result, the production cost (obj_1) decreases significantly (45%).
- In the solution of one of the objective functions, one of the models which are produced on the same assembly line can be grouped with different models in different assembly lines in the solution of another objective function, if it is more suitable. Since the change in resource assignment combinations affects physical loads and energy consumption depending on the resource type, the model-line assignment process is also affected by this change.
- For both cycle times, in the solution of obj_2, which focuses on the energy consumption of cobots, the mathematical model prioritizes operator usage and reduces cobot usage as much as possible.
- In the solution of obj_3, which focuses on distributing the physical loads of the operators who are assigned to workstations, the mathematical model balances physical loads by distributing tasks/operators to workstations, regardless of whether it affects the number of resources used or not.

Table 4 Capability matrix for resource types

Resource type	Task																				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Operator Type-1	1	0	1	1	0	1	0	1	1	1	0	1	1	0	1	0	1	1	0	1	1
Operator Type-2	0	1	1	1	0	1	0	1	1	0	1	1	1	0	1	0	1	1	1	0	0
Cobot Type-1	1	1	1	0	1	0	1	0	1	1	1	1	0	1	0	1	0	1	1	1	0
Cobot Type-2	1	1	0	1	1	0	1	0	1	1	1	0	1	1	0	1	0	1	0	0	0

Table 5 Common Precedence Diagram

	Task																				
Task	1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Successor	3	12	5	4	5	6	7	8	9	10	11	18	13	14	15	16	17	18	19	20	21

Table 6 Data for cost

Cost type	Cost (Euro/Time-unit)
Line	0.1
Workstation	0.06
Operator Type-1	0.7
Operator Type-2	0.6
Cobot Type-1	0.1
Cobot Type-2	0.08

Table 7 Energy consumption data of cobots

Data type	Value
Energy Consumption During Operation (Type 1/Type 2)	0.43/0.38 (PM2.5)
Energy Consumption During Stand By (Type 1/Type 2)	0.043/0.038 (PM2.5)
PM2.5 average energy consumption emission coefficient	2.4956 (g/kWh)

Table 8 Task-based physical load of operators

	Operator Type-1 Model (1/2/3)			Operator Type-2 Model (1/2/3)		
1	0.95	0.85	0.45	0.92	0.82	0.43
2	0.90	0.75	0.58	0.88	0.72	0.56
3	0.75	0	0	0.73	0	0
4	0	1.45	1.32	0	1.40	1.30
5	0.97	1.03	1.23	0.95	1	1.15
6	0	0.42	1.12	0	0.39	1.10
7	1.65	2.13	2.62	1.58	2.10	2.50
8	1.60	1.73	2.38	1.53	1.62	2.25
9	2.30	2.32	3.30	2.15	2.15	3.13
10	0	0	1.32	0	0	1.30
11	1.62	1.60	1.58	1.52	1.50	1.48
12	0.97	0.97	0	0.95	0.95	0
13	1.02	1.04	0	1	1.01	0
14	0.97	1.09	0.68	0.93	1.03	0.63
15	0.75	0.85	0	0.70	0.80	0
16	0	0	2.03	0	0	2.00
17	1.65	1.23	0.83	1.60	1.20	0.80
18	1.32	2.32	2.35	1.30	2.30	2.30
19	0.73	0.83	0.63	0.70	0.80	0.60
20	0.83	0.43	1.33	0.80	0.40	1.30
21	0.62	0.62	0.82	0.60	0.60	0.80

4.2 Performance tests

In addition to the numerical example, which is used for a detailed explanation of the mathematical model above, test problems in the literature are also used to test the model's performance (https://assembly-line-balancing.de/). Jackson (11), Roszieg

Table 9 Independent Solutions of the Objective Functions

Objective	Solutions with Cycle Time-1			Solutions with Cycle Time-2		
	1	2	3	1	2	3
obj_1	**897.61**	1295.41	1230.81	**493.2**	1288.81	1137.61
obj_2	1482.68	**864.02**	1632.72	1611.61	**739.61**	1440.09
obj_3	7.91	29.54	**3.39**	6.08	22.66	**2.3**

(25), Heskia (28), Lutz (32), Kilbrid (45), Hahn (53), and Kim (61) test problems, whose number of tasks gradually increases, are developed in accordance with the problem to produce three or four models.

The times retrieved from these test problems (except for Kim (61)) are considered for Product Model-A and Operator Type-1. Afterward, the task times of Model 2, Model 3, and Model 4 of Operator Type-1 are randomly derived from being within +—10% of the original values. Since Kim (61) is a mixed-model data set containing four models, there is no need to produce additional data for Model 2, Model 3, and Model 4. After the data of Operator Type-1 is determined, the task times of other resources are randomly derived such that Operator Type-2 is 5%-10% longer than Operator Type-1, Cobot Type-1 is 10%-15% shorter than Operator Type-1, and Cobot Type-2 is 5%-10% longer than Cobot Type-1. To enhance realism, task times for different product models across all resource types have a 5% probability of being set to zero. Precedence diagrams in test problems are used as common precedence diagrams. Notably, there is no predefined ratio between operators and cobots, as this does not inherently impact the mathematical model's performance. The purpose is to replicate real-world conditions and promote diversity by simply generating data.

The test problems are initially solved with challenging/limited cycle times, followed by alternative cycle times that are 30% longer. Three different time limits are applied as Python Gurobi 10.0 solver parameters: 1800 (30 min), 3600 (1 h) and 14,400 (4 h), and absolute and relative GAP are equal to zero in all solutions.

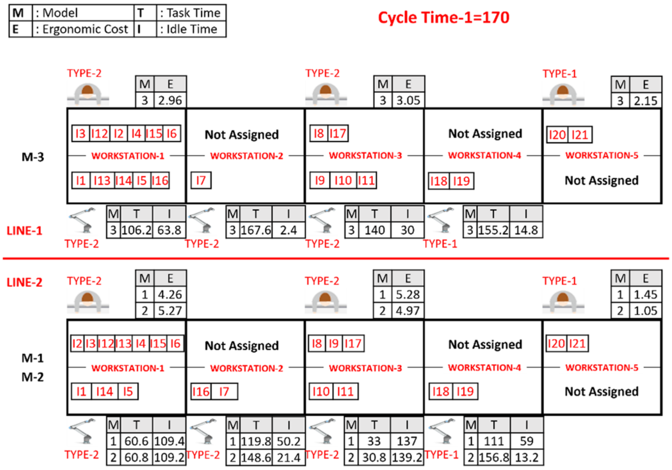
As described above, Jackson (11), Roszieg (25), Heskia (28), Lutz (32), Kilbrid (45), Hahn (53) and Kim (61) test problems were solved with two different cycle times and three different time-limit combinations. The solutions obtained are presented in Table 11. Note that "INF" in the column "Objective Value" means that no solution was reached within the defined time limit. If an "INF" value exists for any objective functions 1, 2, or 3, multi-objective cannot be calculated for that sample.

When the solutions regarding the test problems in the literature are examined:

- Since the initial cycle times are challenging/limited in the mathematical model, some solutions exhibit a nonzero GAP even for the small cases (e.g., Heskia—28). As the test problem size increases, the GAP increases further. For Kim (61), some instances yield no results.
- Given the complexity of the problem, solutions with an 1800-s time limit sometimes fail to yield results, starting from Lutz (32). Consequently, solving medium- and large-scale problems within acceptable time limits becomes increasingly difficult.
- Overall, particularly from Heskia (28) onward, the CPU times of the objective functions follow an increasing order: obj_1, obj_2 and obj_3. Solution times are

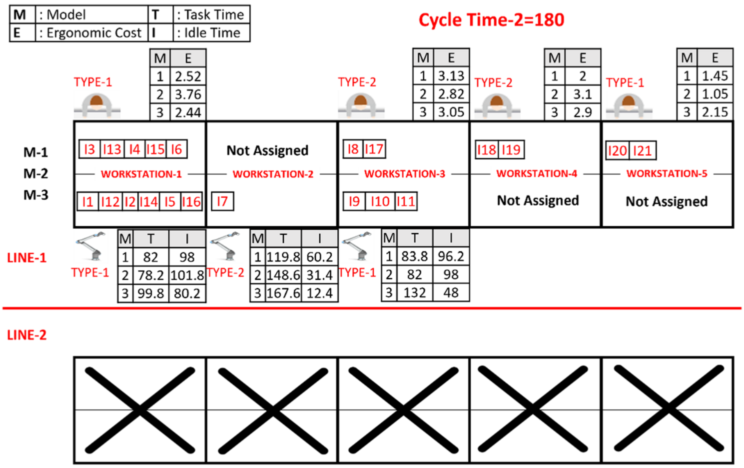
Table 10 Optimal solutions

Data	Obj	Objective Value	CPU (sec)	Number of Lines Opened	Number of Opened Workstations (Line 1/ Line 2)	Models Assigned to Lines (Line 1/Line 2)	Number of Operators Used (Type 1/Type 2)	Number of Cobots Used (Type 1/Type 2)
Cycle Time-1	obj_1	897.61	87.25	2	5/5	3/1,2	2/4	2/6
	obj_2	864.02	118.96	2	5/5	3/1,2	7/2	4/4
	obj_3	3.39	209.85	2	5/5	1,3/2	7/1	7/3
	obj_{Multi}	0.9 **(897.61- 1348.68- 10.36)**	149.18	2	5/5	3/1,2	2/4	2/6
Cycle Time-2	obj_1	493.2	118.25	1	5/0	1,2,3/–	1/2	2/3
	obj_2	739.61	176.71	2	5/5	2,3/1	6/3	2/2
	obj_3	2.3	295.04	2	5/5	2/1,3	5/2	5/4
	obj_{Multi}	0.86 **(590.41- 1077.95- 9.39)**	131.51	1	5/0	1,2,3/–	2/2	2/1



Task	Line-1 Start Time	Line-1 Task Time	Line-2 Start Time	Line-2 Task Time
I1	0	7.2	0	23.4
I2	7.2	14.6	0	24
I3	7.2	0	24	16.6
I4	21.8	33.4	100.8	39.2
I5	55.2	26.4	140	20.2
I6	81.6	30.8	160.2	9.8
I7	2.4	167.6	0	148.6
I8	0	25.6	0	42.4
I9	25.6	88.4	42.4	62.2
I10	118.4	23.8	104.6	0
I11	142.2	27.8	137	33
I12	7.2	0	40.6	23
I13	7.2	0	70.2	30.6
I14	7.2	14.8	121	19
I15	55.2	0	143.2	17
I16	81.6	57.8	0	0
I17	25.6	26.6	104.6	45.2
I18	0	145.6	0	144
I19	160.4	9.6	157.2	12.8
I20	133.6	22.4	142	17.4
I21	156	14	159.4	10.6

Fig. 2 Optimal line balance representation for cycle Time-1



Task	Line-1 Start Time	Line-1 Task Time
I1	0	20.4
I2	35.4	16
I3	20.4	13.6
I4	67.6	35.2
I5	102.8	23.4
I6	152.2	27.8
I7	0	167.6
I8	0	42.4
I9	43.8	85.4
I10	129.2	20.8
I11	150	30
I12	20.4	15
I13	35.4	27.6
I14	86.8	16
I15	109.2	14
I16	126.2	53.8
I17	134.8	45.2
I18	0	153.6
I19	159.2	20.8
I20	0	22.4
I21	166	14

Fig. 3 Optimal line balance representation for cycle Time-2

directly influenced by the type of decision variables (binary, linear, etc.) in the objective functions and the number of parameters affecting them. In addition, the constraints that enable the assembly lines and workstations to be opened sequentially also shorten the solution time of obj_1.

- The assignment of models to lines, the number of opened lines and workstations, and the allocation of operators and cobots vary depending on the objective function. While obj_1 focuses on using as few resources as possible because it is cost-oriented, obj_2 prioritizes operator usage, and obj_3 achieves results by distributing a large number of operators to workstations if it is beneficial for physical load balance. As a result, even when performing the same tasks, the number of resources

Table 11 Solutions to test problems

Data Set/ Max. Station Number	Number of Tasks	Cycle Time (sec)	Time Limit (sec)	Objective	Objective Value	CPU (sec)	GAP	Number of Lines Opened	Number of Opened W.Stations (Line 1/ Line 2)	Models Assigned to Lines (Line 1/ Line 2)	Number of Operators Used (Type 1/ Type 2)	Number of Cobots Used (Type 1/ Type 2)
Jackson/ 3	n=11	10	1800	obj_1	34.4	5,38	0	2	3/3	3/1,2	0/4	0/6
				obj_2	43.55	9,6	0	2	3/3	1,2/3	2/4	1/4
				obj_3	0.17	6,26	0	2	3/3	1,2/3	1/3	2/4
				obj_{Multi}	0.87	7,18	0	2	3/3	3/1,2	0/4	1/5
			3600	obj_1	34.4	5,44	0	2	3/3	3/1,2	0/4	0/6
				obj_2	43.55	9,58	0	2	3/3	1,2/3	2/4	1/4
				obj_3	0.17	6,27	0	2	3/3	1,2/3	1/3	2/4
				obj_{Multi}	0.87	6,68	0	2	3/3	3/1,2	0/4	1/5
			14,400	obj_1	34.4	5,33	0	2	3/3	3/1,2	0/4	0/6
				obj_2	43.55	9,47	0	2	3/3	1,2/3	2/4	1/4
				obj_3	0.17	6,32	0	2	3/3	1,2/3	1/3	2/4
				obj_{Multi}	0.87	6,73	0	2	3/3	3/1,2	0/4	1/5
		13	1800	obj_1	14.82	0,6	0	1	3/0	1,2,3/–	0/1	1/2
				obj_2	21.53	5,97	0	2	3/3	3/1,2	3/3	0/3
				obj_3	0	0,33	0	2	3/3	2,3/1	2/0	3/3
				obj_{Multi}	0.76	4,07	0	1	3/0	1,2,3/–	1/2	0/2
			3600	obj_1	14.82	0,63	0	1	3/0	1,2,3/–	0/1	1/2
				obj_2	21.53	6,02	0	2	3/3	3/1,2	3/3	0/3
				obj_3	0	0,35	0	2	3/3	2,3/1	2/0	3/3
				obj_{Multi}	0.76	4,18	0	1	3/0	1,2,3/–	1/2	0/2
			14,400	obj_1	14.82	0,62	0	1	3/0	1,2,3/–	0/1	1/2
				obj_2	21.53	6,1	0	2	3/3	3/1,2	3/3	0/3
				obj_3	0	0,32	0	2	3/3	2,3/1	2/0	3/3
				obj_{Multi}	0.76	4,16	0	1	3/0	1,2,3/–	1/2	0/2

Table 11 (continued)

Data Set/ Max. Station Number	Number of Tasks	Cycle Time (sec)	Time Limit (sec)	Objective	Objective Value	CPU (sec)	GAP	Number of Lines Opened	Number of Opened W.Stations (Line 1/ Line 2)	Models Assigned to Lines (Line 1/ Line 2)	Number of Operators Used (Type 1/ Type 2)	Number of Cobots Used (Type 1/ Type 2)
Roszieg/ 4	n=25	20	1800	obj_1	130	83,93	0	2	4/4	2/1,3,4	3/5	4/4
				obj_2	197.41	115,89	0	2	4/4	1,3,4/2	6/2	4/4
				obj_3	3.98	99,71	0	2	4/4	2/1,3,4	6/2	4/4
				obj_{Multi}	0.37	93,69	0	2	4/4	1,4/2,3	3/5	4/4
			3600	obj_1	130	95,37	0	2	4/4	2/1,3,4	3/5	4/4
				obj_2	197.41	114,16	0	2	4/4	1,3,4/2	6/2	4/4
				obj_3	3.98	99,65	0	2	4/4	2/1,3,4	6/2	4/4
				obj_{Multi}	0.37	96,12	0	2	4/4	1,4/2,3	3/5	4/4
			14,400	obj_1	130	94,66	0	2	4/4	2/1,3,4	3/5	4/4
				obj_2	197.41	116,85	0	2	4/4	1,3,4/2	6/2	4/4
				obj_3	3.98	99,34	0	2	4/4	2/1,3,4	6/2	4/4
				obj_{Multi}	0.37	96,49	0	2	4/4	1,4/2,3	3/5	4/4
		26	1800	obj_1	67.6	71,41	0	1	4/0	1,2,3,4/–	1/2	2/2
				obj_2	132.15	513,68	0	2	4/4	1,3,4/2	6/2	2/3
				obj_3	1.08	585,31	0	2	4/4	1,4/2,3	4/3	5/3
				obj_{Multi}	0.79	234,38	0	1	4/0	1,2,3,4/–	2/2	2/2
			3600	obj_1	67.6	64,25	0	1	4/0	1,2,3,4/–	1/2	2/2
				obj_2	132.15	441,71	0	2	4/4	1,3,4/2	6/2	2/3
				obj_3	1.08	557,5	0	2	4/4	1,4/2,3	4/3	5/3
				obj_{Multi}	0.79	215,73	0	1	4/0	1,2,3,4/–	2/2	2/2
			14,400	obj_1	67.6	66,27	0	1	4/0	1,2,3,4/–	1/2	2/2
				obj_2	132.15	474,3	0	2	4/4	1,3,4/2	6/2	2/3
				obj_3	1.08	587,32	0	2	4/4	1,4/2,3	4/3	5/3
				obj_{Multi}	0.79	241,45	0	1	4/0	1,2,3,4/–	2/2	2/2

Table 11 (continued)

Data Set/ Max. Station Number	Number of Tasks	Cycle Time (sec)	Time Limit (sec)	Objective	Objective Value	CPU (sec)	GAP	Number of Lines Opened	Number of Opened W.Stations (Line 1/ Line 2)	Models Assigned to Lines (Line 1/ Line 2)	Number of Operators Used (Type 1/ Type 2)	Number of Cobots Used (Type 1/ Type 2)
Heskia/ 5	n=28	115	1800	obj_1	646.31	1039,11	0	2	5/5	2/1,3,4	3/3	6/4
				obj_2	1677.79	1800	0.0605	2	5/5	1,3,4/2	8/2	5/3
				obj_3	1.33	1800	0.9162	2	5/5	1,3,4/2	6/3	8/2
				obj_{Multi}	0.79	1800	1.5635	2	5/4	1,3,4/2	3/5	5/4
			3600	obj_1	646.3	742,95	0	2	5/5	2/1,3,4	3/3	6/4
				obj_2	1677.79	3600	0.0032	2	5/5	1,3,4/2	8/2	5/3
				obj_3	1.33	3600	0.4277	2	5/5	1,3,4/2	6/3	8/2
				obj_{Multi}	0.79	3600	0.2129	2	5/4	1,3,4/2	3/5	5/4
			14,400	obj_1	646.3	757,16	0	2	5/5	2/1,3,4	3/3	6/4
				obj_2	1677.79	8254,61	0	2	5/5	1,3,4/2	8/2	5/3
				obj_3	1.31	14,400	0.0023	2	5/5	1,3,4/2	5/3	6/4
				obj_{Multi}	0.79	5688,72	0	2	4/5	2/1,3,4	3/5	5/4
		150	1800	obj_1	324	306,37	0	1	5/0	1,2,3,4/–	1/1	3/2
				obj_2	1053.39	1800	0.3804	2	5/5	3/1,2,4	6/4	2/3
				obj_3	0.11	1800	1	2	5/5	2,3/1,4	4/1	6/4
				obj_{Multi}	0.67	1800	0.4018	1	5/0	1,2,3,4/–	3/1	2/1
			3600	obj_1	324	327,6	0	1	5/0	1,2,3,4/–	1/1	3/2
				obj_2	1053.39	3600	0.3804	2	5/5	3/1,2,4	6/4	2/3
				obj_3	0.11	3600	1	2	5/5	2,3/1,4	4/1	6/4
				obj_{Multi}	0.67	3600	0.1477	1	5/0	1,2,3,4/–	3/1	2/1
			14,400	obj_1	324	308,43	0	1	5/0	1,2,3,4/–	1/1	3/2
				obj_2	1053.39	14,400	0.0709	2	5/5	3/1,2,4	6/4	2/3
				obj_3	0.04	14,400	1	2	5/5	2,3/1,4	5/0	7/3

Table 11 (continued)

Data Set/ Max. Station Number	Number of Tasks	Cycle Time (sec)	Time Limit (sec)	Objective	Objective Value	CPU (sec)	GAP	Number of Lines Opened	Number of Opened W.Stations (Line 1/ Line 2)	Models Assigned to Lines (Line 1/ Line 2)	Number of Operators Used (Type 1/ Type 2)	Number of Cobots Used (Type 1/ Type 2)
				obj_{Multi}	0.67	2524,24	0	1	5/0	1,2,3,4/–	3/1	2/2
Lutz/ 5	n=32	1800	1800	obj_1	14,688	1543,68	0	2	5/5	2,4/1,3	5/5	3/7
				obj_2	INF	1800	–	–	–	–	–	–
				obj_3	6.38	1800	1	2	5/5	1,3/2,4	7/3	7/3
				obj_{Multi}	INF	1800	–	–	–	–	–	–
			3600	obj_1	14,688	1406,06	0	2	5/5	2,4/1,3	5/5	3/7
				obj_2	23,927.16	2411,75	0	2	5/5	2,4/1,3	6/4	3/7
				obj_3	6.38	2601,92	0	2	5/5	1,3/2,4	7/3	7/3
				obj_{Multi}	0.95	1677,05	0	2	5/5	2,4/1,3	5/5	5/5
			14,400	obj_1	14,688	1547,83	0	2	5/5	2,4/1,3	5/5	3/7
				obj_2	23,927.16	2659,79	0	2	5/5	2,4/1,3	6/4	3/7
				obj_3	6.38	3051,94	0	2	5/5	1,3/2,4	7/3	7/3
				obj_{Multi}	0.95	2028,7	0	2	4/5	2,4/1,3	5/5	5/5
		2340	1800	obj_1	6411.6	1637,37	0	1	5/0	1,2,3,4/–	1/2	2/3
				obj_2	15,328.03	1800	0.3841	2	5/5	3/1,2,4	5/5	1/7
				obj_3	0.55	1800	1	2	5/5	2,3/1,4	6/2	6/4
				obj_{Multi}	0.69	1800	1.0804	1	5/0	1,2,3,4/–	3/1	2/3
			3600	obj_1	6411.6	1410,27	0	1	5/0	1,2,3,4/–	1/2	2/3
				obj_2	15,328.03	3600	0.2988	2	5/5	3/1,2,4	5/5	1/7
				obj_3	0.37	3600	1	2	5/5	3/1,2,4	5/3	6/4
				obj_{Multi}	0.68	3600	0.603	1	5/0	1,2,3,4/–	3/2	1/3
			14,400	obj_1	6411.6	1414,4	0	1	5/0	1,2,3,4/–	1/2	2/3
				obj_2	15,328.03	14,400	0.1192	2	5/5	3/1,2,4	5/5	1/7
				obj_3	0.37	14,400	1	2	5/5	3/1,2,4	5/3	6/4

Table 11 (continued)

Data Set/ Max. Station Number	Number of Tasks	Cycle Time (sec)	Time Limit (sec)	Objective	Objective Value	CPU (sec)	GAP	Number of Lines Opened	Number of Opened W.Stations (Line 1/ Line 2)	Models Assigned to Lines (Line 1/ Line 2)	Number of Operators Used (Type 1/ Type 2)	Number of Cobots Used (Type 1/ Type 2)
				obj_{Multi}	0.68	14,400	0.3897	1	5/0	1,2,3,4/–	3/2	1/3
Kilbrid/ 5	n=45	57	1800	obj_1	491.34	1800	0.5929	2	5/5	3/1,2,4	9/1	6/4
				obj_2	INF	1800	–	–	–	–	–	–
				obj_3	INF	1800	–	–	–	–	–	–
				obj_{Multi}	INF	1800	–	–	–	–	–	–
			3600	obj_1	474.24	3600	0.5494	2	5/5	3/1,2,4	6/4	6/4
				obj_2	INF	3600	–	–	–	–	–	–
				obj_3	INF	3600	–	–	–	–	–	–
				obj_{Multi}	INF	3600	–	–	–	–	–	–
			14,400	obj_1	474.24	14,400	0.4914	2	5/5	3/1,2,4	6/4	6/4
				obj_2	INF	14,400	–	–	–	–	–	–
				obj_3	INF	14,400	–	–	–	–	–	–
				obj_{Multi}	INF	14,400	–	–	–	–	–	–
		74	1800	obj_1	INF	1800	–	–	–	–	–	–
				obj_2	776.1	1800	0.6287	2	5/5	2,3,4/1	6/4	5/3
				obj_3	2.16	1800	1	1	5/0	1,2,3,4/–	3/1	4/1
				obj_{Multi}	INF	1800	–	–	–	–	–	–
			3600	obj_1	254.56	3600	0.3026	1	5/0	1,2,3,4/–	2/2	2/3
				obj_2	695.27	3600	0.5855	2	5/5	2,3,4/1	8/2	2/5
				obj_3	0.72	3600	1	2	5/5	2/1,3,4	7/2	8/2
				obj_{Multi}	0.38	3600	0.8188	1	5/0	1,2,3,4/–	3/2	2/2
			14,400	obj_1	210.16	14,400	5.4E–07	1	5/0	1,2,3,4/–	2/1	2/3
				obj_2	692.71	14,400	0.5839	2	5/5	2,3,4/1	8/2	2/5
				obj_3	0.24	14,400	1	2	5/5	2/1,3,4	6/2	7/3

Table 11 (continued)

Data Set/ Max. Station Number	Number of Tasks	Cycle Time (sec)	Time Limit (sec)	Objective	Objective Value	CPU (sec)	GAP	Number of Lines Opened	Number of Opened W.Stations (Line 1/ Line 2)	Models Assigned to Lines (Line 1/ Line 2)	Number of Operators Used (Type 1/ Type 2)	Number of Cobots Used (Type 1/ Type 2)
				obj_{Multi}	0.43	14,400	0.355	1	5/0	1,2,3,4/–	3/2	2/2
Hahn/ 5	n = 53	2190	1800	obj_1	17,520	1800	1.6E–08	2	5/5	2/1,3,4	5/5	3/5
				obj_2	21,970.17	1800	0.2828	2	5/5	2/1,3,4	7/3	5/4
				obj_3	44.18	1800	0.0515	2	5/5	2,4/1,3	10/0	6/3
				obj_{Multi}	0.59	1800	0.3525	2	5/5	2,4/1,3	5/5	4/5
			3600	obj_1	17,520	2875,05	0	2	5/5	2/1,3,4	5/5	3/5
				obj_2	21,629.85	2860,05	0	2	5/5	2,4/1,3	6/4	4/5
				obj_3	44.18	2901,53	0	2	5/5	2,4/1,3	10/0	6/3
				obj_{Multi}	0.65	2803,02	0	2	5/5	1,3/2,4	5/5	4/5
			14,400	obj_1	17,520	2406,22	0	2	5/5	2/1,3,4	5/5	3/5
				obj_2	21,629.85	2423,77	0	2	5/5	2,4/1,3	6/4	4/5
				obj_3	44.18	2465,5	0	2	5/5	2,4/1,3	10/0	6/3
				obj_{Multi}	0.65	2395,29	0	2	5/5	1,3/2,4	5/5	4/5
		2847	1800	obj_1	INF	1800	–	–	–	–	–	–
				obj_2	13,696.3	1800	0.8634	2	5/5	2,4/1,3	6/4	3/4
				obj_3	1.874	1800	1	1	5/0	1,2,3,4/–	4/0	4/1
				obj_{Multi}	INF	1800	–	–	–	–	–	–
			3600	obj_1	9281.22	2152,43	0	1	5/0	1,2,3,4/–	2/2	1/2
				obj_2	12,299.73	3600	0.8298	2	5/5	3/1,2,4	6/4	3/5
				obj_3	0.57	3600	1	2	5/5	2,4/1,3	4/4	6/4
				obj_{Multi}	INF	3600	–	–	–	–	–	–
			14,400	obj_1	9281.22	2340,24	0	1	5/0	1,2,3,4/–	2/2	1/2
				obj_2	12,299.73	14,400	0.3069	2	5/5	3/1,2,4	6/4	3/5
				obj_3	0.46	14,400	1	2	5/5	2,4/1,3	4/4	8/2

Table 11 (continued)

Data Set/ Max. Station Number	Number of Tasks	Cycle Time (sec)	Time Limit (sec)	Objective	Objective Value	CPU (sec)	GAP	Number of Lines Opened	Number of Opened W.Stations (Line 1/ Line 2)	Models Assigned to Lines (Line 1/ Line 2)	Number of Operators Used (Type 1/ Type 2)	Number of Cobots Used (Type 1/ Type 2)
				obj_{Multi}	0.57	14,400	1.025	1	5/0	1,2,3,4/–	2/3	1/3
Kim/ 5	n=61	18	1800	obj_1	INF	1800	–	–	–	–	–	–
				obj_2	INF	1800	–	–	–	–	–	–
				obj_3	INF	1800	–	–	–	–	–	–
				obj_{Multi}	INF	1800	–	–	–	–	–	–
			3600	obj_1	INF	3600	–	–	–	–	–	–
				obj_2	INF	3600	–	–	–	–	–	–
				obj_3	INF	3600	–	–	–	–	–	–
				obj_{Multi}	INF	3600	–	–	–	–	–	–
			14,400	obj_1	59.76	14,400	0.4077	1	5/0	1,2,3,4/–	1/3	1/4
				obj_2	INF	14,400	–	–	–	–	–	–
				obj_3	2.23	14,400	1	2	5/5	1,3,4/2	8/2	8/2
				obj_{Multi}	INF	14,400	–	–	–	–	–	–
		23	1800	obj_1	INF	1800	–	–	–	–	–	–
				obj_2	INF	1800	–	–	–	–	–	–
				obj_3	INF	1800	–	–	–	–	–	–
				obj_{Multi}	INF	1800	–	–	–	–	–	–
			3600	obj_1	INF	3600	–	–	–	–	–	–
				obj_2	INF	3600	–	–	–	–	–	–
				obj_3	INF	3600	–	–	–	–	–	–
				obj_{Multi}	INF	3600	–	–	–	–	–	–
			14,400	obj_1	61.18	14,400	0.3119	1	5/0	1,2,3,4/–	1/2	2/2
				obj_2	70.69	14,400	0.162	2	5/5	1,2/3,4	3/7	0/4
				obj_3	0.14	14,400	1	2	5/5	1,3/2,4	6/1	5/5

Table 11 (continued)

Data Set/ Max. Station Number	Number of Tasks	Cycle Time (sec)	Time Limit (sec)	Objective	Objective Value	CPU (sec)	GAP	Number of Lines Opened	Number of Opened W.Stations (Line 1/ Line 2)	Models Assigned to Lines (Line 1/ Line 2)	Number of Operators Used (Type 1/ Type 2)	Number of Cobots Used (Type 1/ Type 2)
				obj_{Multi}	0.93	14,400	1.002	1	5/0	1,2,3,4/–	2/2	2/2

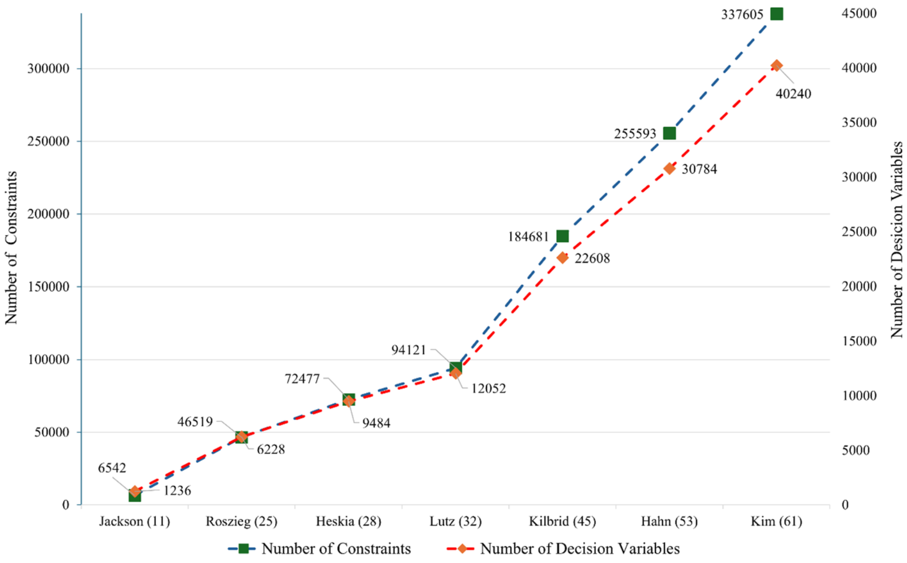


Fig. 4 Comparison of the number of constraints and decision variables

used in the optimal solution of each objective function can differ.

- Among the test problems from Heskia (28) onward, each includes two different cycle times. However, only for the relatively shorter cycle times of Lutz (32) and Hahn (53) was the optimal result (GAP=0) achieved across all objective functions (obj_1, obj_2, obj_3, and obj_{Multi}). This outcome serves as an indicator of the problem's complexity.

To further illustrate the increasing complexity of the test problems and the computational challenges faced by the proposed model, an analysis of the growth in the number of decision variables and constraints as the number of tasks increases is conducted. This analysis is particularly important in understanding the scalability and practical limitations of the approach. A graphical illustration is provided in Fig. 4. As depicted, both the number of decision variables and constraints increase significantly in a nonlinear fashion, which directly contributes to the sharp rise in CPU time and computational difficulty.

5 Conclusions

The mixed-model robotic multiple assembly line balancing problem, where decisions involve assigning models to assembly lines, represents a significant gap in the literature. Additionally, studies that simultaneously address production cost, cobot energy consumption, and balanced physical load distribution among operators remain limited. This paper attempts to fill in this gap by introducing a simultaneous MLA-RMMALB problem to produce multiple product models concurrently on multiple lines. The multi-objective mathematical model simultaneously optimizes (i) production cost, (ii) energy consumption by cobots, and (iii) physical loads on operators. The problem allows mixed-model production on each line, considers diverse operators and cobots with varying capabilities and processing times, and permits one operator and one cobot to work in parallel at each workstation under a fixed cycle time.

A numerical example with 21 tasks is solved to describe the fundamental characteristics of the problem and solution method. The example is first solved separately for each objective function and then for the multi-objective function to assess the impact of different objectives on multiple assembly lines. Results indicate that the mathematical model provides optimal solutions aligned with the given objectives.

This is followed by a comprehensive computational study based primarily on data retrieved from the literature. To evaluate model performance, test problems with an increasing number of tasks –Jackson (11), Roszieg (25), Heskia (28), Lutz (32), Kilbrid (45), Hahn (53), and Kim (61)—are developed and solved to produce three or four product models. The findings reveal a rapid and nonlinear increase in CPU time as the problem size grows. Specifically, the CPU times required to solve the multi-objective function for the first (challenging) cycle time in Jackson (11), Roszieg (25), and Lutz (32) are approximately 7, 96, and 2028s, respectively, confirming this exponential trend. To highlight the scalability issue, the increasing number of decision variables and constraints with growing task sizes has also been examined. This increase directly contributes to the computational complexity observed in larger test problems.

While this study makes valuable contributions, it has certain limitations. The complexity of the model restricts the size of problems that can be efficiently solved. As the number of tasks, resources, and overall problem complexity increase, finding optimal solutions within reasonable time limits becomes increasingly difficult. Moreover, in real-world scenarios, operator task times fluctuate due to factors like fatigue or skill level, whereas this study assumes deterministic processing times.

To address these limitations, future research might focus on developing meta-heuristic solution methods to enhance computational efficiency, particularly for medium- and large-sized problems. Moreover, conducting sensitivity analyses by varying key parameters, such as resource capabilities and cycle time, would provide deeper insights into the model's adaptability across different production environments.

This study provides valuable managerial insights for industries employing mixed-model robotic assembly lines. First, the proposed approach enables decision-makers to simultaneously optimize production cost, energy consumption, and workload distribution, fostering more efficient and sustainable production planning. Second, by considering heterogeneous resources and their capabilities, the model allows for a more effective allocation of operators and cobots, thereby enhancing overall productivity. Finally, the findings emphasize the trade-offs between solution accuracy and computational effort, suggesting that heuristic or hybrid approaches may be more practical for large-scale applications. These insights can aid manufacturers in designing more adaptive, resource-efficient, and flexible assembly systems capable of responding to dynamic production environments.

Furthermore, the proposed model can be applied to different production settings. In small- and medium-scale environments, it supports cost-effective planning and flexible task assignments on a single line. In contrast, large-scale operations, such as those in the automotive or electronics industries, can benefit from its ability to balance multiple lines, assign diverse product models, and optimize operator-cobot collaboration. In such settings, energy efficiency and workload balance become even more critical. The model's flexibility in handling heterogeneous resources also makes it suitable for modern smart factories. These differences illustrate how decision-making priorities may vary depending on production scale and structure, highlighting the model's practical relevance.

Acknowledgements The authors would like to acknowledge that this paper is presented as a partial fulfillment of the Ph.D. degree requisites at Yildiz Technical University. This study is produced within the scope of Oktay Yilmaz's Ph.D. dissertation.

Data availability Data used in the experimental tests can be accessed through Harvard Dataverse, https://doi.org/10.7910/DVN/YOXQVL.

Declarations

Conflict of interest The authors confirm that there is no potential competing interest to report.

References

Ab Rashid MFF, Mohamed NMZN, Oumer AN (2022) Modelling and optimization of energy efficient assembly line balancing using modified moth flame optimizer. Int J Integr Eng 14(1):25–39

Abdous M-A, Delorme X, Battini D, Sgarbossa F, Berger-Douce S (2023) Assembly line balancing problem with ergonomics: a new fatigue and recovery model. Int J Prod Res 61(3):693–706. https://doi.org/10.1080/00207543.2021.2015081

Aguilar H, García-Villoria A, Pastor R (2020) A survey of the parallel assembly lines balancing problem. Comput Oper Res 124:105061

Aydin N, Cetinkale Z (2022) Analyses on ICU and non-ICU capacity of government hospitals during the COVID-19 outbreak via multi-objective linear programming: an evidence from Istanbul. Comput Biol Med 146:105562

Barathwaj N, Raja P, Gokulraj S (2015) Optimization of assembly line balancing using genetic algorithm. J Cent South Univ 22:3957–3969

Battaïa O, Dolgui A (2013) A taxonomy of line balancing problems and their solutionapproaches. Int J Prod Econ 142(2):259–277

Battini D, Delorme X, Dolgui A, Sgarbossa F (2015) Assembly line balancing with ergonomics paradigms: two alternative methods. IFAC-PapersOnLine 48(3):586–591

Battini D, Delorme X, Dolgui A, Persona A, Sgarbossa F (2016) Ergonomics in assembly line balancing based on energy expenditure: a multi-objective model. Int J Prod Res 54(3):824–845

Bautista J, Batalla-García C, Alfaro-Pozo R (2016) Models for assembly line balancing by temporal, spatial and ergonomic risk attributes. Eur J Oper Res 251(3):814–829

Belkharroubi L, Yahyaoui K (2022) Solving the energy-efficient robotic mixed-model assembly line balancing problem using a memory-based cuckoo search algorithm. Eng Appl Artif Intell 114:105112

Bortolini M, Faccio M, Gamberi M, Pilati F (2017) Multi-objective assembly line balancing considering component picking and ergonomic risk. Comput Ind Eng 112:348–367

Boschetti G, Faccio M, Milanese M, Minto R (2021) C-ALB (collaborative assembly line balancing): a new approach in cobot solutions. Int J Adv Manuf Technol 116:3027–3042

Boysen N, Schulze P, Scholl A (2022) Assembly line balancing: what happened in the last fifteen years? Eur J Oper Res 301(3):797–814

Chutima P (2023) Assembly line balancing with cobots: an extensive review and critiques. Int J Ind Eng Comput 14(4):785–804

Chutima P, Khotsaenlee A (2022) Multi-objective parallel adjacent U-shaped assembly line balancing collaborated by robots and normal and disabled workers. Comput Oper Res 143:105775

Cil ZA, Li Z, Mete S, Ozceylan E (2020) Mathematical model and bee algorithms for mixed-model assembly line balancing problem with physical human–robot collaboration. Appl Soft Comput 93:106394

Dalle Mura M, Dini G (2023) Improving ergonomics in mixed-model assembly lines balancing noise exposure and energy expenditure. CIRP J Manuf Sci Technol 40:44–52

Deniz N, Ozcelik F (2024) Bi-objective optimization-based multi-criteria decision-making framework for disassembly line balancing and employee assignment problem. Kybernetes 53(3):1073–1091

Gao J, Sun L, Wang L, Gen M (2009) An efficient approach for type II robotic assembly line balancing problems. Comput Ind Eng 56(3):1065–1080

Garg A, Chaffin DB, Herrin GD (1978) Prediction of metabolic rates for manual materials handling jobs. Am Ind Hyg Assoc J 39(8):661–674

Huang Y, Sheng B, Luo R, Lu Y, Fu G, Yin X (2024) Solving human-robot collaborative mixed-model two-sided assembly line balancing using multi-objective discrete artificial bee colony algorithm. Comput Ind Eng 187:109776

Junior MCP, Michels AS, Magatão L (2023) An exact method to incorporate ergonomic risks in assembly line balancing problems. Comput Ind Eng 183:109414

Kara Y, Atasagun Y, Gokcen H, Hezer S, Demirel N (2014) An integrated model to incorporate ergonomics and resource restrictions into assembly line balancing. Int J Comput Integr Manuf 27(11):997–1007

Keshvarparast A, Battaia O, Pirayesh A, Battini D (2022) Considering physical workload and workforce diversity in a collaborative assembly line balancing (C-ALB) optimization model. IFAC-PapersOnLine 55(10):157–162

Keshvarparast A, Katiraee N, Pirayesh A, Battaia O, Berti N (2023) Integrated resource optimization in a multi-product separated line collaborative assembly line balancing problem (MPSLC-ALBP). IFAC-PapersOnLine 56(2):713–718

Kheirabadi M, Keivanpour S, Chinniah Y, Frayret J-M (2022) A review on collaborative robot assembly line balancing problems. IFAC-PapersOnLine 55(10):2779–2784

Kucukkoc I, Zhang DZ (2014) Simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines. Int J Prod Res 52(12):3665–3687. https://doi.org/10.1080/00207543.2013.879618

Kucukkoc I, Zhang DZ (2015) Type-E parallel two-sided assembly line balancing problem: mathematical model and ant colony optimisation based approach with optimised parameters. Comput Ind Eng 84:56–69

Kucukkoc I, Zhang DZ (2016) Integrating ant colony and genetic algorithms in the balancing and scheduling of complex assembly lines. Int J Adv Manuf Technol 82:265–285

Levitin G, Rubinovitz J, Shnits B (2006) A genetic algorithm for robotic assembly line balancing. Eur J Oper Res 168(3):811–825

Li Z, Janardhanan MN, Tang Q (2021) Multi-objective migrating bird optimization algorithm for cost-oriented assembly line balancing problem with collaborative robots. Neural Comput Appl 33(14):8575–8596

Li Y, Liu D, Kucukkoc I (2023a) Mixed-model assembly line balancing problem considering learning effect and uncertain demand. J Comput Appl Math 422:114823. https://doi.org/10.1016/j.cam.2022.114823

Li Z, Kucukkoc I, Tang Q, Zhang Z (2023b) Models and two-phase bee algorithms for multi-objective U-shaped disassembly line balancing problem. Optim Eng 24(1):591–622

Li Z, Sikora CGS, Kucukkoc I (2024) Chance-constrained stochastic assembly line balancing with branch, bound and remember algorithm. Ann Oper Res 335(1):491–516. https://doi.org/10.1007/s10479-023-05809-1

Lovato D, Guillaume R, Thierry C, Battaïa O (2024) Aircraft final assembly line scheduling with user preferences and equity of work distribution. Flex Serv Manuf J. https://doi.org/10.1007/s10696-024-09566-6

Lusa A (2008) A survey of the literature on the multiple or parallel assembly line balancing problem. Eur J Ind Eng 2(1):50–72

Mao Z, Zhang J, Fang K, Huang D, Sun Y (2023) Balancing u-type assembly lines with human–robot collaboration. Comput Oper Res 159:106359

Marler RT, Arora JS (2010) The weighted sum method for multi-objective optimization: new insights. Struct Multidiscip Optim 41(6):853–862. https://doi.org/10.1007/s00158-009-0460-7

Meng K, Tang Q, Zhang Z (2023) Balancing and sequencing of mixed-model assembly line considering preventive maintenance scenarios: mathematical model and a migrating birds optimization algorithm. Flex Serv Manuf J 35(4):1175–1205

Nourmohammadi A, Fathi M, Ng AH (2022) Balancing and scheduling assembly lines with human-robot collaboration tasks. Comput Oper Res 140:105674

Otto A, Battaïa O (2017) Reducing physical ergonomic risks at assembly lines by line balancing and job rotation: a survey. Comput Ind Eng 111:467–480

Ozcan U, Gokcen H, Toklu B (2010) Balancing parallel two-sided assembly lines. Int J Prod Res 48(16):4767–4784

Ponnambalam SG, Chang Q, Zhong RY, Kucukkoc I, Janardhanan MN (2023) Human-robot collaboration in the next generation manufacturing and logistics system. Flex Serv Manuf J 35(4):975–978. https://doi.org/10.1007/s10696-023-09506-w

Rohmert W (1973) Problems of determination of rest allowances Part 2: Determining rest allowances in different human tasks. Appl Ergon 4(3):158–162

Sahin MC, Tural MK (2023) Robotic stochastic assembly line balancing. Flex Serv Manuf J 35(4):1076–1115

Salveson ME (1955) The assembly-line balancing problem. Trans ASME 77(6):939–947

Scholl A, Boysen N (2009) Designing parallel assembly lines with split workplaces: model and optimization procedure. Int J Prod Econ 119(1):90–100

Sikora CGS, Weckenborg C (2023) Balancing of assembly lines with collaborative robots: comparing approaches of the Benders' decomposition algorithm. Int J Prod Res 61(15):5117–5133

Sivasankaran P, Shahabudeen P (2014) Literature review of assembly line balancing problems. Int J Adv Manuf Technol 73:1665–1694

Soysal Kurt H, Isleyen SK, Gokcen H (2024) Balancing and sequencing of mixed-model parallel robotic assembly lines considering energy consumption. Flex Serv Manuf J 1–29

Tasan SO, Tunali S (2008) A review of the current applications of genetic algorithms in assembly line balancing. J Intell Manuf 19:49–69

Weckenborg C, Spengler TS (2019) Assembly line balancing with collaborative robots under consideration of ergonomics: a cost-oriented approach. IFAC-PapersOnLine 52(13):1860–1865

Weckenborg C, Kieckhäfer K, Müller C, Grunewald M, Spengler TS (2020) Balancing of assembly lines with collaborative robots. Bus Res 13(1):93–132

Zacharia PT, Xidias EK, Nearchou AC (2024) The fuzzy human-robot collaboration assembly line balancing problem. Comput Ind Eng 187:109774

Zhang Z, Tang Q, Ruiz R, Zhang L (2020) Ergonomic risk and cycle time minimization for the U-shaped worker assignment assembly line balancing problem: a multi-objective approach. Comput Oper Res 118:104905

Dr. Oktay Yilmaz received his B.Sc. degree in Electronic Engineering from the Turkish National Defense University, Istanbul, Turkey, in 2010. He obtained his M.Sc. degree in Industrial Engineering from Yildiz Technical University, Istanbul, in 2021, and his Ph.D. degree in Industrial Engineering from the same university in 2025. He has been working as a Product Manager at ERD Information Technologies since 2021.

Dr. Nezir Aydin holds a PhD in Industrial and Systems Engineering from Wayne State University/MI/USA and an MSc and BSc in Industrial Engineering from Yildiz Technical University/Istanbul/Turkiye. He conducted over ten research projects and taught graduate and undergraduate-level courses in Supply chain management, Stochastic optimization, Decision-making under uncertainty, Modeling and optimization, Transportation, Simulation and System analysis. He has published over 80 research papers in Scopus and WOS-indexed journals. Dr. Aydin was among the Top 2% of Scientists Worldwide in 2023 & 2024. He supervised several MSc. and Ph.D. students.

Dr. Ibrahim Kucukkoc is a Professor in the Department of Industrial Engineering at Balikesir University, Turkey, where he has been a faculty member since December 2009. He earned his Ph.D. from the University of Exeter, UK, in 2015 with the support of a four-year scholarship from the Turkish Council of Higher Education. His research focuses on the modelling and optimisation of modern manufacturing systems, with particular expertise in assembly line balancing, production planning for additive manufacturing, and the application of advanced algorithms to complex combinatorial optimisation problems.

Authors and Affiliations

Oktay Yilmaz[1] · **Nezir Aydin**[1,2] · **Ibrahim Kucukkoc**[3]

✉ Oktay Yilmaz
oktay.yilmaz@std.yildiz.edu.tr

Nezir Aydin
naydin@hbku.edu.qa

Ibrahim Kucukkoc
ikucukkoc@balikesir.edu.tr

1 Department of Industrial Engineering, Yildiz Technical University, 34349 Besiktas, Istanbul, Turkey

2 College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar

3 Department of Industrial Engineering, Balikesir University, Balikesir, Turkey