Constraints (2018) 23:123–153
https://doi.org/10.1007/s10601-017-9279-9

Mixed model line balancing with parallel stations, zoning constraints, and ergonomics

Anas Alghazi[1] · **Mary E. Kurz**[2]

Published online: 29 November 2017

Abstract Assembly lines are cost efficient production systems that mass produce identical products. Due to customer demand, manufacturers use mixed model assembly lines to produce customized products that are not identical. To stay efficient, management decisions for the line such as number of workers and assembly task assignment to stations need to be optimized to increase throughput and decrease cost. In each station, the work to be done depends on the exact product configuration, and is not consistent across all products. In this paper, we propose a mixed model line balancing integer program (IP) that considers parallel workers, zoning, task assignment, and ergonomic constraints with the objective of minimizing the number of workers. Upon observing the limitation of the IP, a Constraint Programming (CP) model is developed to solve larger assembly line balancing problems. Data from an automotive OEM are used to assess the performance of both the MIP and CP models, including sensitivity analysis to measure the computational cost of enabling the different constraints. To the best of our knowledge, we are the first paper to incorporate the different realistic mixed model assembly line constraints and develop a CP model based on the scheduling module of the IBM ILOG Optimizations Studio. Using the OEM data, we show that the CP model outperforms the IP model for bigger problems.

Keywords Assembling line balancing · Mixed model assembly lines · Mixed integer programs

✉ Mary E. Kurz
mkurz@clemson.edu

Anas Alghazi
alghazi@kfupm.edu.sa

[1] System Engineering Department, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia

[2] Department of Industrial Engineering, Clemson University, Clemson, SC 29634-0920, USA

1 Introduction

An assembly line is a flow-oriented production system that is composed of productive areas called stations arranged in a serial manner. The final product starts as a workpiece that moves down the line passing through all stations on a transportation system such as a conveyor belt. At each station, assembly operations are performed on the workpiece. The amount of time each workpiece is within the boundaries of a station depends on the conveyor speed. The interval between launching workpieces is called the cycle time. Generally, the total amount of assembly operations done on the workpiece by a single operator at any given station cannot exceed this cycle time. The assembly operations have some precedence constraints because of technological and organizational conditions. The decision problem of assigning the assembly operations to stations with respect to some objective while satisfying the cycle time limitation and the precedence requirement between operations is referred to as the assembly line balancing problem.

Initially, assembly lines were used as a cost-efficient way for mass production of identical products. However, to respond to customer needs, companies added the ability to customize the products ordered. Automation and multipurpose machines made it easier to produce customized products on the same line in a cost-effective way. Assembly lines are associated with considerable investment costs, which makes the configuration of these lines an important factor.

Most research in the configuration planning of assembly systems is focused on line balancing. The base problem, with the smallest set of restrictions that differentiates this problem from the bin-packing problem, is the simple assembly line balancing problem (SALBP) [1]. Some later studies extended the simple problem by incorporating practical aspects such as parallel stations and zoning constraints; this type of research is categorized under the general assembly line balancing name [2]. Despite this effort in extending the problem to be more realistic, there is still a gap between academic papers and practical implementation. To this day, many automotive assembly organizations still line balance manually. Boysen et al. [6] hypothesized some reasons behind this gap between the academia and practice. The first reason is that researchers have not considered the “true” real-world problem yet. The second reason is that the problem was studied but no satisfactory solution was found to it. Finally, the third reason is that scientific results could not be translated into something practical. Our experience with industry provides another explanation: published research sometimes provides too narrow of a solution that cannot be flexible for the changing business environment or requires the maintenance of too much data that is not otherwise needed. The classification scheme proposed in [6] was the first step to close this gap between academia and implementation.

Becker and Scholl [3] consider the problem of balancing assembly lines with variable parallel workplaces to minimize the number of workplaces needed. A workplace is the combination of a worker at a station. The model proposed is based on the assumption of mass production of a homogenous product. In this work, we extend the model of [3] to the mixed model environment and account for ergonomic constraints, usually ignored in line balancing models. We develop both an integer programming (IP) model and a constraint programming (CP) model for the problem at hand and assess the performance of both models using data from industry.

In this paper, the problem description is given first, followed by an example to illustrate the different constraints of the problem and the solution structure. Next, a literature review is given in Section 3, followed by motivation in Section 4. In Section 5, the mathematical program is given, followed by the constraint programming model in Section 6. Computational experiments and results discussion are presented in Section 7. The models are exercised in a sensitivity study in Section 8. Finally, concluding remarks and future possible research directions are given in Section 9.

2 Problem description

To achieve realism, we incorporate several conditions and limitations that are faced in an industry's real assembly line. Various constraints that make a line balancing problem applicable in real world industrial setting are explained in detail in [12]. Our problem is based on the assembly line balancing problem with flexible parallel workstations that was introduced by [3] and observed in an automotive OEM. The objective is to minimize the total number of workers on the line subject to several constraints.

Our problem has the following SALBP characteristics: serial line layout, fixed cycle time (paced line), and deterministic task times that are less than the cycle time. In addition, the following characteristics are included: mixed model assembly, parallel workers, zoning constraints, assignment restrictions, and ergonomic risk constraints.

2.1 Mixed model environment

In a mixed model environment, different base models that may include a variety of options are assembled on the same line. Automakers who provide their customers with flexibility in customizing cars usually operate under a make-to-order environment. The number of options lead to a huge number of possible customized cars to be assembled on the same line. The installation of different options typically leads to variations in assembly task times; for example, the installation of a power liftgate requires a different amount of time when compared to the manual liftgate installation.

There are two primary approaches to deal with variation in task durations when enforcing the cycle time constraint. The first is enforcing the sum of *average* task times assigned to a worker to not exceed the cycle time. The drawback of this method is having some models exceed the cycle time at some stations. The second method is enforcing the cycle time on all workpieces that may be processed by each worker, which unfortunately requires a higher cycle time, which is translated into a lower production rate. Moreover, it requires the line balancers to know every combination of tasks that might be required of all possible workpieces. Pil and Holweg [23] describe the number of model varieties that are available from different car manufacturers; for some German car manufacturers, there could be over 10ˆ24 potentially unique workpieces. Boysen et al. [7] reviews the different approaches used to handle the mixed model problem in the literature. In this paper, the first approach is used in which the average task time is guaranteed to be less than the cycle time. The option-based precedence graph introduced by [8] is used in this research based on the proportion p_h of products (cars) that require task h with duration d_h . This proportion is derived using the forecasted demand of the volume of cars that will require the execution of that task divided by the total production volume. The weighted task time for task h, $\bar{d}_h$ is calculated as follows $\bar{d}_h = p_h d_h \forall h$.

2.2 Zoning constraints

Zoning constraints are used to avoid worker interference in a station and reduce non-productive walking time between mounting positions. Each task is done in a specific location on the product. This attribute is called the mounting position and is one of the different positions available in a product zone map. Figure 1 shows an example of a product zone map in which there are 9 different mounting positions.

Each task h has a mounting position q_h from the 9 positions on the product zone in which the task takes place. Mapping each task to a mounting position avoids overlap in the scheduling of two tasks hl with the same mounting position $q_h = q_l$ at any station.

2.3 Parallel workers

It is common to have more than one operator working within the same station in assembly lines for products such as cars and trucks because of the relatively large parts and space to accommodate multiple workers. We assume that each worker is not fixed to one work area inside the station and can be assigned different tasks that have different mounting positions. Furthermore, we provide detailed scheduling of the tasks assigned to each worker to avoid conflicts between workers in the same station. Infeasibility could occur even if the sum of all task times assigned within a station is less than the cycle time. This is because there exist precedence relations between tasks that could lead to idle time if one operator is waiting for another to finish a task.

2.4 Assignment restrictions

Some tasks are restricted to be assigned to some stations due to different reasons. For example, a task that requires a resource that is available only on some stations or a task that is done below the car and requires the car to be elevated. These restrictions can be categorized as follows

2.4.1 Resource constraints

This constraint is used when a task requires a certain resource such as a tool that is not available in all stations. For example, some tools are heavy or expensive so they will be fixed to certain stations. Tasks that require this tool should be only assigned to this station. A task h that requires a certain resource not available in all stations has a set of eligible stations U_h. If a tool is available in only one station, tasks that need this tool are assigned to this station in the preprocessing step.

2.4.2 Adjacency constraints

This constraint is for tasks that need to be executed consecutively by the same worker. For example, a task for picking up a part must be followed by a task to assemble it. In this

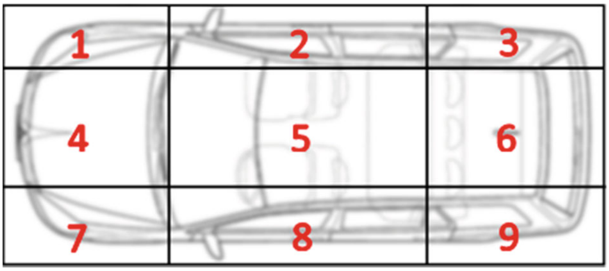

Fig. 1 Product zone map

case, the worker is not open to any other tasks since he has the part in his hands. Adjacency constraints are treated in the preprocessing phase by combining the adjacent set of tasks into one task. Some manufacturers may store information about their tasks in such a manner that this preprocessing is required in practice.

2.4.3 Same worker constraints

Some tasks need to be executed by the same worker but not necessarily consecutively. An example would be inspection or quality checks that should be done by a worker after finishing a task. Since the inspection may involve several tasks or intervening tasks are allowed, the same worker constraint is used instead of the adjacency constraint. All tasks constrained to be done by the same worker are collected in sets labeled by R^{sw}.

2.4.4 Incompatible worker constraint

A set of tasks may not be allowed to be assigned to the same worker. This constraint is mainly used to prohibit unproductive walking by limiting the assignment of tasks with opposite mounting positions to the same worker. An example for the product zone in Fig. 1 would be to limit any worker from being assigned a pair of tasks hl with mounting positions $\{(q_h, q_l) = (1, 9)\ (7, 3)\ (4, 6)\}$. That is, if a worker is assigned a task with mounting position 1 the assignment of all tasks with mounting position 9 is prohibited for him since it would be unproductive to walk all the way from product zones 1 to 9. All tasks that are incompatible worker are collected in sets labeled by R^{nw}.

2.4.5 Same station constraint

This constraint is used to indicate that a set of tasks needs to be done on the same station but not necessarily done by the same worker. An example would be a large piece that needs to be installed by two workers via two different tasks. Both tasks need to be done on the same station so that each worker would be assigned one of them. All tasks that need to be assigned to the same station are collected in sets labeled by R^{ss}.

2.4.6 Not the same station

This constraint prohibits a set of tasks from being assigned to the same station. An example for using this constraint would be for a task to lubricate a movable part and task to install the seat. These two tasks can't be assigned to the same station to avoid the danger of soiling the seat with the lubrication material. All tasks that are station incompatible are collected in sets labeled by R^{ns}.

2.5 Ergonomics risks constraints

Ergonomic risks are treated in two different ways, directly and indirectly. An indirect way of treating ergonomic risk is divided to station-related and worker-related methods. An example for dealing with an indirect station related ergonomic risk is the relation between tasks and the position of the car on the line. Some tasks that require installing parts above the car need the car to be tilted to avoid injury for the workers executing them. These tasks should be restricted to station in which the car can be tilted. Furthermore, tackling the worker interference issue and unproductive walking are indirect worker-lever ergonomic risk constraints.

On the other hand, the direct application of ergonomic risk mitigation can be accomplished through a weighted ergonomic risk score $\bar{g}_h$ associated with each task h to ensure that each worker is assigned tasks with a total weighted ergonomic risk score less than a predefined level G.

2.6 Pre-processing

The pre-processing phase is important to prepare the input before it is used with the mathematical program. For the mixed model problem, this involves computing the weighted task durations and the weighted ergonomic scores. Some of the available information is useful in lowering the number of decision variables by fixing some of them. It is possible to remove unnecessary constraints by reducing the set of variables the constraint applies to. This helps in getting a tighter feasible solution space and helps in reducing the combinatorial explosion. The pre-processing phase involves the following:

- Reduce the number of tasks by combining the tasks in an adjacency set into one task.
- Defining the feasible stations set F for all tasks by using project management techniques to calculate the time window defined by the early start and late finish times for each task. By calculating this window, the feasible stations set for each task can be populated by stations that overlap in time with the calculated window.
- Defining the eligible stations set U for all tasks that require a specific resource that is not available in all stations. If this resource is available in one station only, all tasks that require it are pre-assigned to this station.
- Calculating the weighted task duration $\bar{d}$ and the weighted ergonomic score $\bar{g}$ for all tasks.
- Populating the assignment constraints sets R^{sw}, R^{nw}, R^{ss}, and R^{ns}.

2.7 An example

Consider the example with 12 tasks and the option-based task precedence graph in Fig. 2. Assume that 10 cars are to be assembled, each with different options that require different tasks. The volume and proportion of each task are given in Table 1 along with the

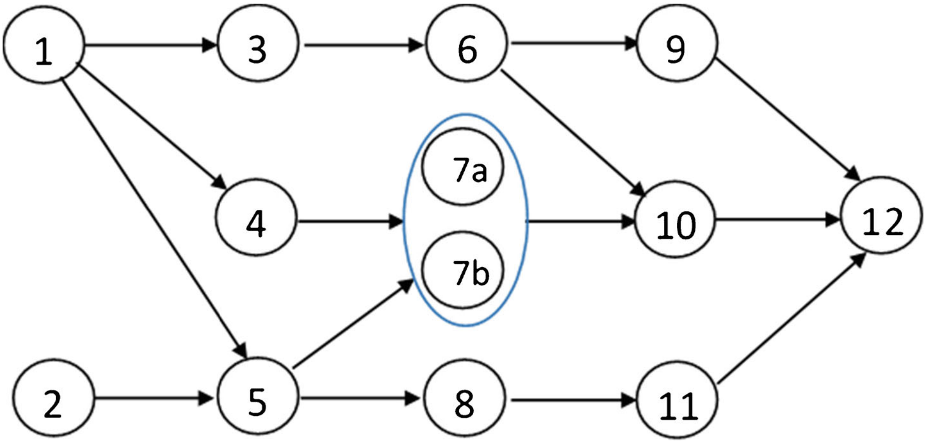

Fig. 2 Option-based joint precedence graph

Table 1 Example data for the line balancing problem

Task	Volume	Proportion	Mounting position	Duration	ErgoS	Weighted duration	Weighted ergoS
1	10	1	1	50	50	50	50
2	10	1	2	30	40	30	40
3	3	0.3	3	80	70	24	21
4	5	0.5	3	50	40	25	20
5	7	0.7	4	70	80	49	56
6	10	1	3	70	90	70	90
7a	10	1	2	20	40	20	40
7b	10	1	2	20	30	20	30
8	8	0.8	1	30	30	24	24
9	10	1	2	50	70	50	70
10	2	0.2	3	70	80	14	16
11	5	0.7	3	50	60	35	42
12	10	1	4	30	30	30	30

task duration and ergonomic score. The weighted values of both the task duration and the ergonomic score are calculated based on the volume of cars that require that task. Recall that the ergonomic and cycle time constraints are enforced on the weighted (average) values only. Figure 3 shows the mounting position for each task. Since the number of stations is predefined, the early start and late finish for each task are calculated and shown in the Gantt chart (Fig. 4) with each task scheduled at its early start time. The Gantt chart helps reduce the feasible station set for each task, for example tasks 1, 6, and 12 can only be assigned to stations 1, 2, and 3 respectively. The assignment constraints are given in Table 2. Since tasks 7a and 7b are in an adjacency set, they are joined together as task 7 in the preprocessing phase. The "incompatible worker" assignment constraint is used in this example to avoid unproductive walking between opposite mounting positions. For example, a worker who is assigned a task with mounting position 1 will not be assigned another task with mounting position 4 and vice versa but he can be assigned tasks with mounting positions 2 and 3. An optimal solution to this example is shown in Fig. 5 in which all 6 potential workers are

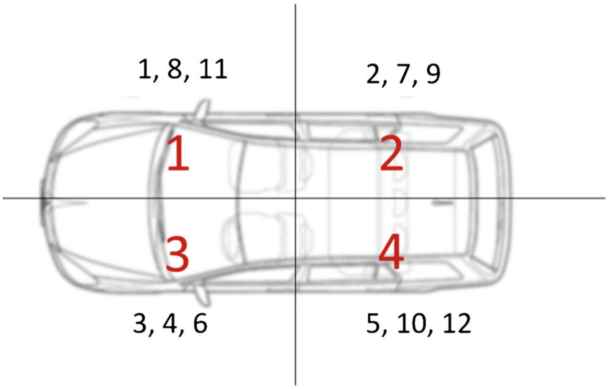

Fig. 3 Mounting position for each task

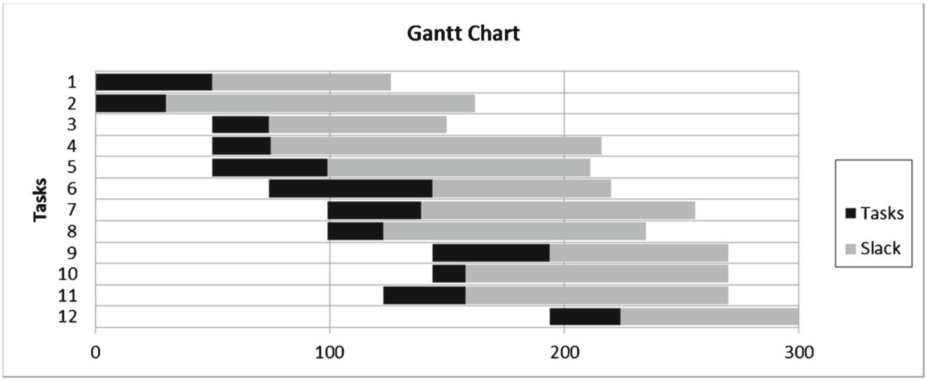

Fig. 4 Gantt chart with tasks at early start times

assigned tasks. In Fig. 5, the tasks are represented by blocks in which the length denotes the weighted duration of the task and the height of the block denotes the weighted ergonomic score. Since task 6 has a high weighted ergonomic score it is the only scheduled task for worker (2,1). The scheduling of tasks to workers helps avoid an infeasible solution because of the precedence relations. For example, worker (1,1) has an unavoidable idle gap since he had to wait for task 1 to finish before starting task 5 because of the precedence relation between the two tasks.

3 Related work

The basic mixed model assembly line balancing problem (MALBP) is based on the same assumptions as the SALBP. The main difference is having multiple models each with its own precedence graph and task times. In the MALBP literature, two basic approaches are used to model and solve the problem: reduction to single model problem and horizontal balancing [2]. The MALBP can be transformed to the SALBP through a joint precedence graph. This graph is constructed by computing the expected processing times based on the probability of each model in the expected model mix, reducing to a single model case to which traditional single model techniques can be applied.

Gokcen and Erel [13] develop a binary goal programming model for the MALBP. The authors claim that they are the first to apply the multiple criteria decision making approach. Although the optimal solution may not be identified through this solution procedure, a satisfactory result that compromises the different conflicting objectives is provided. In addition, Gokcen and Erel [14] propose a binary integer programming model for the MALBP along

Table 2 Assignment constraints

Assignment constraint	Tasks
Adjacency set	(7a,7b)
Same worker	(9,12)
Same station	(4,5)
Not the same worker	$\{(hl) : q_h q_l = (1,4)\,, (2,3)\}$
Not the same station	(10,11)

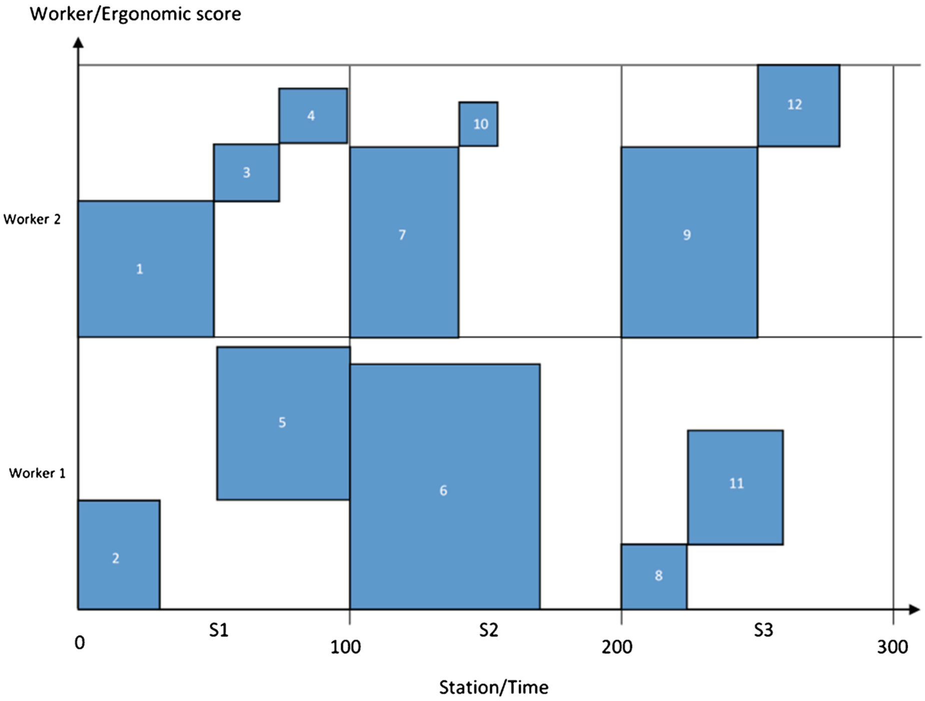

Fig. 5 An optimal solution to the example

with some computational properties of the model. They claim that their model is superior to the other in the literature in terms of decision variables numbers and constraints. Furthermore, a shortest route formulation for the MALBP is presented by [11], requiring the reduction of the mixed model to a single model by the combined precedence graph.

Some related work that is beyond the scope of this paper is the inclusion of scheduling (meaning what ordered workpiece is released when) or consideration of multiple objectives. Sawik [24] compares two approaches to solve the combined balancing and scheduling problem for the flexible assembly line. In the monolithic approach, the balancing and assembly decisions are made simultaneously using a mixed integer program. For the hierarchical approach, the station workloads are balanced first and then the assembly process is scheduled by solving a permutation flow shop problem. Öztürk et al. [18] also study the problem of balancing and scheduling in mixed model assembly lines with parallel stations. They formulate the problem as a mixed integer program and then propose a decomposition scheme to be applied for large scale applications. Kara et al. [15] solve a MALBP for model mixes that have precedence conflicts with a binary mathematical model with a single objective and then extend the model to incorporate three objectives into a single objective model. Mahdavi et al. [16] propose a fuzzy multi-objective linear programming model for solving multi-objective MALBP.

3.1 Exact methods

Most of the literature on exact solution methods is focused on the SALBP. To use the single model solution methods on the mixed model problem, the mixed model need to be reduced to a single model through a combined precedence graph. Sewell and Jacobson [27] present an exact algorithm for the SALBP named the branch, bound and remember algorithm that combines branch and bound with dynamic programming. The authors claim that this algorithm manages to find the optimal solution for all combined benchmark problems of Hoffmann, Talbot, and Scholl. Scholl and Becker [26] review state of the art exact and heuristic methods used to solve the SALBP. Although solving the average SALBP guarantees that the line balance will not violate the cycle time on average, the optimal solution might still have significant inefficiencies when put to practice based on the exact sequence of workpieces that are encountered. Scholl [25] point out that there are some necessary modifications that need to be done on any SALBP exact solution when applying it to a MALBP.

Ege et al. [10] propose a branch and bound algorithm to solve the assembly line balancing problem with station paralleling. Boysen and Fliedner [5] propose a two-stage graph algorithm (Avalanche) that to solve GALBP with relevant constraints. The authors claim that this approach can be easily modified to include extensions such as parallel work stations, processing alternatives, zoning restrictions, stochastic processing times and U-shaped assembly lines.

Vilà-and-Pereira [28] study the assembly line worker assignment and balancing problem. They provide an exact enumeration procedure that is based on the branch-bound and remember algorithm presented by [27]. The authors develop several lower bounds, reductions, and dominance rules for the problem.

Wilhelm and Gadidov [29] develop two models to address different tooling requirements in the multiple product assembly system design problem. The authors propose a branch and cut approach that employs a facet generation procedure to generate cuts in the search tree by exploiting some special structures of the problem. The authors demonstrate the use of the approach through experiments and compare it to the available solution procedures.

One of the assumptions for the MALBP states that identical tasks are assigned to the same station for all models. The problem can be decomposed into several SALBP (one for each model) by relaxing this assumption. Bukchin and Rabinowitch [9] relax the assumption of assigning identical tasks across different models to the same station, which allows a common task to be assigned to different stations for different models. The authors develop an integer program formulation and an exact solution method based on a backtracking branch and bound algorithm. In their model, some costs are associated with assigning an identical task to different stations. However, according to [2], this relaxation is not desired in practice for several reasons such as the complex production control, facility requirements, loss of specialization effects and setup inefficiencies.

3.2 Constraint programming models

There has not been much research in using CP to solve the assembly line balancing problem in the past although it is one of the most successful tools in solving combinatorial problems. Bockmayr and Pisaruk [4] propose a hybrid approach for solving the SALBP by combining CP and IP. The main contribution in their paper is developing a branch and cut solver for the SALBP and to show how it can make use of the CP solver in pruning the search tree. Pastor et al. [21] compare the performance of impulse variables based models, step variable based models, CP models, and IP models in solving the SALBP with different objectives. The authors conclude that the CP formulation performs better and is faster than IP but the best results were for the impulse variable based models.

Most of the research done using CP in the area of assembly line balancing (combined with sequencing) can be attributed to one research group: Öztürk et al. [17–20]. Öztürk et al. [17] develop MIP and CP formulation to solve the simultaneous balancing and scheduling of flexible mixed model assembly lines with sequence-dependent setup times. Results shows that CP formulation is promising when compared to MIP. In addition, Öztürk et al. [19] propose a MIP and a CP model to solve the flexible mixed model assembly line balancing and sequencing problems. They compare the performance of the MIP, the complete and various decomposition methods, and CP, finding that the CP outperforms all the other approaches over all sizes of the test instances. Recently, Öztürk et al. [20] develop a new constraint programming model formulation for simultaneous balancing and cyclic scheduling of flexible mixed model assembly lines with parallel stations with the objective of minimizing the cycle time of models. This is an extension the authors' previous work with the non-cyclic problem. The authors employ assignment based symmetry breaking constraints and customized search strategy. The CP outperform the MIP and MIP based decomposition heuristics. To the best of our knowledge there is no other published research that uses CP in solving the mixed model assembly line balancing problem (without sequencing) with parallel stations, zoning constrains and ergonomics.

4 Motivation

The bulk of the research in the solution procedures for the MALBP is based on heuristics and meta-heuristics, perhaps because the SALBP is a bin packing problem when the precedence constraints are removed. The bin packing problem is a known NP-hard problem, thus MALBP is NP-hard since it is based on SALBP. More details on the complexity of the problem are given in [25]. There is a gap between the research and the industry in line balancing. Falkenauer [12] discussed how the SALBP solution methods discussed in the literature have little or no application to the real-world assembly line balancing problem. Furthermore, the extensions in the GALBP literature are often not applicable to industry. Each extension may tackle a generalization in one direction, while other research might consider other generalizations in another direction; meanwhile, industry needs a way to deal with its own unique set of generalizations at the same time. This study includes many generalizations seen in practice and the models proposed within can readily be adapted as new constraints are discovered or created, perhaps through the creation of new business rules. The main characteristics needed to capture realism and make the solution applicable are summarized as follows:

- The majority of line balancing takes place at a previously-built assembly plant so the number of stations is fixed and re-balancing is what is needed. However, the number of workers can be impacted by effective re-balancing.
- Each station has its own identity with the tooling and restrictions of the tasks that can be done on it.
- There are zoning constraints that should be considered in line balancing.
- Since the assembly plants are already built, elimination of stations is not feasible.
- Smoothing the workload across stations should be an important objective of the line balance.
- Multiple operators usually work in a station, so a schedule is needed to increase utilization and remove idle time.

- Station level ergonomic risks should be considered and might lead to assignment restrictions. Also incorporating worker level ergonomic risks such as avoiding work area interference between workers should be incorporated.
- In a multi-product line, care should be taken in computing the average task times, horizontal balancing should be applied, and handling of drifting operations.

To the best of our knowledge, no published research has considered these characteristics simultaneously. Moreover, Pearce [22] studied the general assembly line balancing problem. The author proposed an IP model and several heuristics to solve the problem. In his work, the IP outperformed all the proposed heuristics. This has motivated us to compare the performance of the IP to the proposed CP solution method.

5 The mathematical model

The integer program formulation for this problem is based on the formulation by [3] for the single model variable parallel workstations assembly line balancing problem. In our model, it’s assumed that tasks have a duration that is less than the cycle time so there is no need to divide tasks between stations. Also, mounting positions are not assigned to workers so several workers in the same station can be assigned tasks that share the same mounting position. Detailed scheduling is used to avoid worker interference.

Each product assembled on the line starts with a workpiece that is launched down the line and have assembly tasks done on it until the workpiece exits the final station K as a finished product. The time between the launch of two workpieces is called the cycle time c and is equivalent to the time spent by each workpiece at each station. The workpiece enters the first station at time $t = 0$ and leaves the last station at time $t = Kc$. For each of the $i = 1, \dots K$ stations there are $j = 1, \dots, W$ potential workers available, so worker (ij) represents the potential worker j at station i. The binary variable $y_{i,j}$ is associated with each potential worker (i, j) and is equal to 1 if the potential worker is assigned and 0 otherwise. Figure 6 shows an example profile for a line with 3 stations and 4 potential workers per station in which the greyed blocks represent the assignment of potential worker (i, j) when

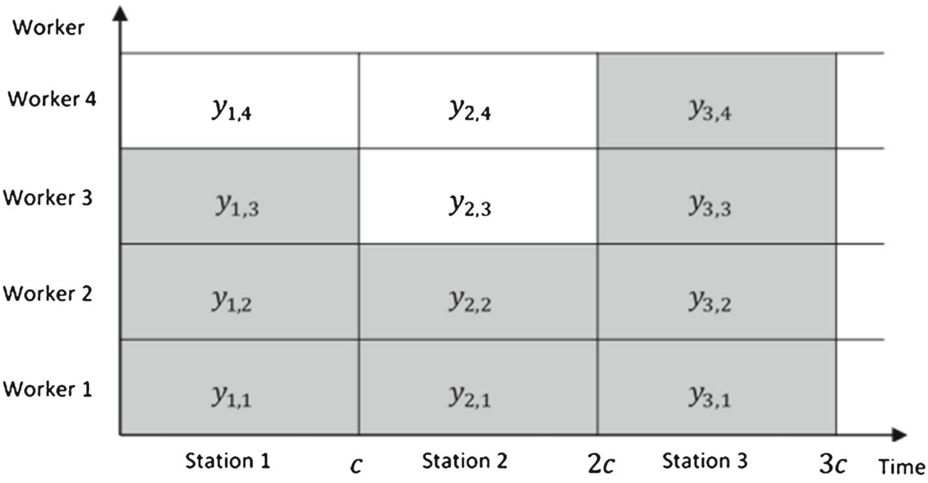

Fig. 6 Example for a worker/station profile

$y_{i,j} = 1$. In this example 3, 2, and 4 workers are assigned to stations 1, 2, and 3 respectively. The model's notation are shown in Table 3.

Decision Variables:

$$x_{ijh} = \begin{cases} 1 & \textit{if task h is assigned to station i and worker j} \\ 0 & \textit{otherwise} \end{cases}$$

$$y_{ij} = \begin{cases} 1 & \textit{if potential worker j at station i is assigned} \\ 0 & \textit{otherwise} \end{cases}$$

$$v_{hl} = \begin{cases} 1 & \textit{if task h is executed before task l} \\ 0 & \textit{otherwise} \end{cases}$$

$$s_h = \textit{starting time of task h}$$

Objective:

$$\textit{minimize } Z = \sum_i \sum_j y_{ij}$$

The objective is to minimize the total number of workers assigned.

Subject to the following constraints:

- Task assignment to worker: each task is assigned to only one worker.

$$\sum_{i \in F_h} \sum_j x_{ijh} = 1 \qquad \forall h \tag{C5.1}$$

Table 3 Line balancing problem's notation

K	Maximum number of stations indexed $i = 1, 2, 3, \ldots, K$
W	Maximum Number of potential workers per station indexed $j = 1, 2, 3, \ldots, W$
N	Number of tasks indexed $hl = 1, 2, 3, \ldots, N$
c	Cycle time
A_i	Starting time of station i $\quad A_i = (i - 1) * c$
t_h^s	Early start time for task h
t_h^f	Late finish for task h
q_h	Mounting position of task h
p_h	Proportion of models that will require task h in the production run
d_h	Duration of task h
$\bar{d}_h$	Weighted duration of task h
g_h	Ergonomic score for task h
$\bar{g}_h$	Weighted ergonomic score for task h
G	Ergonomic risk maximum weighted score limit
F_h	The set of feasible stations that task h is assignable to
U_h	The set of stations with the required resources to perform task h
O_h	The set of immediate predecessors of task h
R^{ss}	The set of tasks that must be done on the same station
R^{ns}	The set of tasks that cannot be done on the same station
R^{sw}	The set of tasks that must to be done by the same worker
R^{nw}	The set of tasks that cannot be done by the same worker

- Cycle time: the total weighted duration assigned to a worker should not exceed the cycle time.

$$\sum_h x_{ijh}\bar{d}_h \leq c y_{ij} \qquad \forall (i \in F_h, j) \tag{C5.2}$$

- Station time: each task assigned to a worker should be scheduled between the worker's station start and finish times.

$$s_h \geq \sum_{i \in F_h} \sum_j A_i x_{ijh} \qquad \forall h \tag{C5.3}$$

$$s_h + \bar{d}_h \leq \sum_{i \in F_h} \sum_j (A_i + c) x_{ijh} \qquad \forall h \tag{C5.4}$$

- Precedence relations: a task can only start when all predecessors are finished.

$$s_h + \bar{d}_h \leq s_l \qquad \forall h, l \in O_l \tag{C5.5}$$

- Tasks overlap (sequencing): all tasks assigned to a worker should not overlap. That is, a worker cannot perform two tasks at the same time. For example, variable v_{hl} is equal to 1 when task h starts and ends before the start of task l.

$$v_{hl} + v_{lh} \geq x_{ijh} + x_{ijl} - 1 \qquad \forall j, h \neq l, \ and \ i \in F_h \cap F_l \tag{C5.6}$$

$$s_h + t_h \leq s_l + (1 - v_{hl})\,(t_h^f - t_h^s) \qquad \forall h \neq l \tag{C5.7}$$

- Worker interference: tasks that share the same mounting position should not overlap. When two workers are assigned two tasks that share the same mounting position, one of the tasks should start and end before the start of the other task.

$$v_{hl} + v_{lh} \geq \sum_j (x_{ijh} + x_{ijl}) - 1 \qquad \forall i \in F_h \cap F_l, h \neq l, \ and \ q_h = q_l \tag{C5.8}$$

- Ergonomic risk: the total weighted ergonomic risk for each worker should be within the maximum limit.

$$\sum_h x_{ijh}\bar{g}_h \leq G \qquad \forall (i \in F_h, j) \tag{C5.9}$$

- Assignment restrictions: pairs of tasks with same (station/worker) or not the same (station/worker) restrictions.

$$\sum_j x_{ijh} = \sum_j x_{ijl} \qquad \forall i \in F_h \cap F_l \ and \ (h,l) \in R^{ss} \tag{C5.10}$$

$$x_{ijh} = x_{ijl} \qquad \forall (i \in F_h \cap F_l, j) \ and \ (h,l) \in R^{sw} \tag{C5.11}$$

$$\sum_j (x_{ijh} + x_{ijl}) \leq 1 \qquad \forall i \in F_h \cap F_l \ and \ (h,l) \in R^{ns} \tag{C5.12}$$

$$x_{ijh} + x_{ijl} \leq 1 \qquad \forall (i \in F_h \cap F_l, j) \ and \ (h,l) \in R^{nw} \tag{C5.13}$$

- Resource requirement: tasks that require a resource should only be assigned to eligible stations.

$$x_{ijh} = 0 \qquad \forall i \notin U_h, j, \ and \ h \tag{C5.14}$$

- Breaking symmetry.

$$y_{ij} \geq y_{i(j+1)} \qquad \forall (i, j \leq W - 1) \tag{C5.15}$$

- Variables domain.

$$x_{ijh}, y_{ij}, v_{hl} \in \{0, 1\}, s_h \in Z^+ \qquad \forall i, j, h, \text{ and } l \tag{C5.16}$$

The cycle time constraint (C5.2) provides a lower bound for the integer program that is equivalent to the capacity bound (LB1) presented in [3]. The lower bound achieved by this constraint gives the lowest possible number of workers needed to perform all tasks which is equivalent to rounding up the sum of all weighted task durations divided by the cycle time.

$$LB = \left\lceil \sum\nolimits_h \bar{d}_h \,/\, c \right\rceil \tag{5.1}$$

To avoid worker interference, constraint (C5.8) is added to make sure that no two tasks that share the same mounting position overlap at any given time in any station. Constraints (C5.10) and (C5.13) are for the same station and same worker assignment restriction constraints. The original model uses the worker incompatible and station incompatible constraints only. Constraint (C5.14) helps in reducing the feasible space by fixing some decision variables using the eligible station set for each task. Since all workers are identical, constraint (C5.15) reduces the problem's feasible region by removing the symmetry from the problem in an effort to reduce the computation time. For example, the constraint enforces the assignment of tasks to worker 1 before assigning worker 2.

6 The constraint programming model

Since it is anticipated that the mathematical programming model will face difficulties in finding a solution for larger problem instances, we propose a constraint programming (CP) model. CP is proven to be a useful technique for solving scheduling problems efficiently. Several tools exist for developing CP models; ILOG CPLEX optimization studio (https://www.ibm.com/us-en/marketplace/ibm-ilog-cplex) is one of the most used for scheduling problems because of the CP Optimizer (ILOG Scheduler) engine that was developed specifically with several constraint propagation and search algorithms that exploit the structure of scheduling problems. CP Optimizer is used in this work to model the mixed model line balancing problem. In the proposed CP model, interval variables and interval sequence variables are used along with specialized constraints such as alternative, no overlap, precedence relation, and logical constraints. These specialized constraints state some relation between different variables in the model. Although these relations could be modeled using traditional logical constraints, the use of the specialized constraint increases the efficiency of the solver by exploiting the structure of the problem with a specialized filtering algorithm.

6.1 The constraint programming model formulation

Decision variables:

$Task_h$: Interval variable associated with task h with a size equal to $\bar{d}_h$

$TaskAssign_{h,i,j}$: Interval variable associated with task h that is performed by worker j on station i, limited to lie within $\left[t_h^s, t_h^f\right] \cap [A_i, A_i + c]$

$Worker_{i,j}$: Sequence variable associated with worker j at station i

Objective:

Minimize $\sum_i \sum_j \max\limits_h presenceOf(TaskAssign_{h,i,j})$

Subject to:

- Task assignment to worker: defining the alternative interval variable such that each task interval variable has an alternative variable for each worker at each station. Only one of the alternatives will be present; in other words, each task is assigned to only one worker.

$$alternative(Task_h, TaskAssign_{\forall i,\forall j}) \quad \forall h \tag{CP2.1}$$

- Precedence relations: a task can start when all predecessors are finished.

$$endBeforeStart(Task_h, Task_l) \qquad \forall h, l \in O_l \tag{CP2.2}$$

- Tasks overlap: tasks that are assigned to one worker should not overlap.

$$noOverlap(Worker_{i,j}) \qquad \forall i, j \tag{CP2.3}$$

- Worker interference: tasks that share the same mounting position should not overlap.

$$noOverlap(Task_h) \qquad \forall h : q_h = 1, 2, 3, \ldots, Q \tag{CP2.4}$$

- Ergonomic risk: the total weighted ergonomic risk for each worker should be within limits.

$$\sum_h presenceOf(TaskAssign_{h,i,j})\bar{g} \leq G \qquad \forall i, j \tag{CP2.5}$$

- Assignment restrictions: pairs of tasks with same (station/worker) or not the same (station/worker) restrictions.

$$\sum_j presenceOf(TaskAssign_{h,i,j}) = \sum_j presenceOf(TaskAssign_{l,i,j})$$
$$\forall j \ and \ (h,l) \in R^{ss} \tag{CP2.6}$$
$$presenceOf(TaskAssign_{h,i,j}) = presenceOf(TaskAssign_{l,i,j})$$
$$\forall i, j, \ and \ (h,l) \in R^{sw} \tag{CP2.7}$$
$$\sum_j (presenceOf(TaskAssign_{h,i,j}) + \sum\nolimits_j presenceOf(TaskAssign_{l,i,j}))$$
$$\leq 1 \forall j \ and \ (h,l) \in R^{ns} \tag{CP2.8}$$
$$presenceOf(TaskAssign_{h,i,j}) + presenceOf(TaskAssign_{l,i,j}) \leq 1$$
$$\forall i, j, \ and \ (h,l) \in R^{nw} \tag{CP2.9}$$

- Resource requirement: tasks that require a resource should only be assigned to eligible stations.

$$presenceOf(TaskAssign_{h,i,j}) == 0 \qquad \forall i \notin U_h, j, \ and \ h \tag{CP2.10}$$

- Breaking symmetry:

$$presenceOf(TaskAssign_{h,i,j+1}) \leq presenceOf(TaskAssign_{h,i,j})$$
$$\forall (i, j \leq W - 1) \tag{CP2.11}$$

7 Computational experiments

We compare the performance of the CP and IP in solving industrial sized assembly line balancing problems using data from the automotive industry. The data belongs to three

Table 4 Number of tasks in each band

Band	Original number of tasks	Number of tasks after pre-processing
Band 1	395	209
Band 26	317	174
Band 47	408	178

different bands of actual assembly lines. The data used in this study was collected over several years and at various stages of facility redesign, and does not conform to any current setting at the time of the research reported in this paper. However, the OEM confirmed the validity of all constraints contained in the models presented. Details on the size of each band before and after the preprocessing phase are provided in Table 4. Four instances are generated from each band's data with different number of tasks (50, 100, 150, and all tasks) and different number of stations. Details on test instances from all bands are provided on Table 5.

Both the Gurobi and CPLEX solvers were tested to choose the best solver to be used for this problem. Our experimental results were uniformly better when using Gurobi so results presented are based upon the IP model as coded in AMPL and solved using Gurobi 6 solver. The CP is coded under the ILOG CPLEX Optimization Studio 12.6 and is solved using the CP Optimizer engine with the default search strategy. Data sets and model files are available from the authors. The computer has an Intel i7-3770 CPU with 3.4 GHz clock speed and 32 GB of memory. The solution time is limited to one hour; if the optimal solution is not provably found, the best solution is recorded along with the time at which the best solution was reached. For CP, the lower bound is similar to the IP one which is equivalent to LB in (5.1). The search in CP is terminated when the solution is equal to the lower bound. Since CP search is based on random number generation, 10 runs were done for each instance with different seeds and the average solution time is recorded. The default inference level is set to "extended" in the CP along with the precedence, interval sequence, and no overlap inference levels.

Table 6 summarizes the results of both the IP and CP subject to a one-hour time limit. Since 10 runs were executed for each instance in CP, the number of runs in which each

Table 5 Test instances

Instance	Number of tasks	Number of stations
B1.1	50	5
B1.2	100	10
B1.3	150	10
B1.4	209	13
B26.1	50	5
B26.2	100	6
B26.3	150	10
B26.4	174	10
B47.1	50	5
B47.2	100	7
B47.3	150	11
B47.4	178	12

Table 6 Solution time under time limit

Instance	LB	IP (3600 sec time limit)		CP (3600 sec time limit)		
		Solution	Solution time (sec.)	Solution	# found	Avg. solution time (sec.)
B1.1	7	7	34.36	7	10	34.96
B1.2	12	15	3208	13	10	327.24
B1.3	16	–	–	1617	64	2263.94 468.03
B1.4	22	–	–	23	10	1684.64
B26.1	5	5	13.47	5	10	34.30
B26.2	9	11	2030	9	10	104.98
B26.3	12	20	2552	12	10	534.73
B26.4	13	–	–	13	10	354.43
B47.1	6	6	17.83	6	10	28.45
B47.2	11	12	720	11	10	136.38
B47.3	16	–	–	16	10	555.20
B47.4	18	–	–	18	10	842.93

solution was found is shown. As expected the IP faced difficulties as the size of the instance increases. The optimal solution is found only in 3 small sized instances while no feasible solutions were found on the largest instance of each band within the one-hour time limit. On the other hand, the CP managed to find optimal solutions in 10 out of the 12 instances with only a small gap in the solution of the other two instances. In CP instance B1.3 two different solutions were found for different seed runs, of which one is optimal. Band 1 instances were difficult for both the CP and IP solvers. Table 7 presents the processing time needed by the IP solver to reach the optimal solution for different instances under no time limits. Results show that CP clearly outperforms the IP in terms of processing time making it more applicable in industry. Table 7 shows the time it takes the IP to reach the optimal solution when there is no time limit, in which some instances utilized all memory before finding an optimal solution.

Table 7 IP solution time under no time limit

Instance	IP time to optimal solution (days, HH:MM:SS)
B1.1	00:00:34
B1.2	1 days, 21:43:19
B1.3	1 days, 19:01:49
B1.4	–
B26.1	00:00:13
B26.2	10:28:52
B26.3	12:37:00
B26.4	–
B47.1	00:00:17
B47.2	05:05:43
B47.3	3 days, 01:19:43
B47.4	–

8 Case study

This section explores the IP and CP models developed for the assembly line balancing problem through an increased sensitivity analysis using the case study data supplied by an automaker. In generating the data for the case study, the research team learned that some data is relatively easier or harder to obtain and maintain, and some constraints may not be discovered until a potential solution is presented that violates the constraints. Requiring an OEM to maintain large databases that are needed for only application to line balancing may increase the potential cost. In practice, satisfying the late-discovered constraints can sometimes be accomplished by a small permutation of the current solution, or the decision maker may determine that this or another constraint can be violated. The nature of the mathematical programming techniques used in this work is to enforce all constraints provided, even if the decision maker considers some of them to be less-critical. The motivation is to build intuition about the relative cost of modeling, data collection and maintenance required to enforce some of the realistic constraints in the mixed model line balancing problem.

Some of the line balancing constraints define the problem at a fundamental level and cannot be ignored or relaxed. It maybe that some elements that are described as constraints are in fact preferences, and can be accommodated partially. If such constraints, when included in the model, are computationally difficult, it may be acceptable to remove them and manually accommodate them later. Moreover, some constraints may not be acknowledged until a balance is proposed, at which time a manual readjustment is made, instead of reformulating and resolving. By studying the effects of relaxing or removing some of these constraints on the model behavior, we gain more insight on the complexity each constraint contributes to the model so that decision makers can make better use of it.

We classify the line balancing constraints as hard or soft constraints. A constraint is labeled "hard" if removing it will either violate any of the SALBP assumptions or fundamentally affect the way the problem is modeled. Otherwise, it is a "soft" constraint.

Hard constraints:

1. Task to worker assignment:
   Every task is assigned to one worker only.
2. Cycle time:
   The total expected task durations assigned to a worker does not exceed the cycle time.
3. Station time:
   Every tasks must be completed on exactly one station.
4. Precedence relation:
   Without the precedence constraint, the problem can be transformed into a bin-packing problem in which the goal is to assign (pack) as many tasks to every station (bin) such that the number of stations is minimized.
5. Tasks overlap:

Two tasks that are assigned to a worker cannot overlap in time.
Soft constraints:

1. Same mounting position worker interference (mounting):
   This constraint prohibits schedules in which two workers work on the same mounting position at the same time. That is, no two tasks that share the same mounting position overlap in time. This constraint can be relaxed assuming that the

interference can either be avoided by manually changing the station's schedule or by communication between workers in the same station. These constraints are either enforced or not.

2. Task assignment restrictions constraints:
   This includes same worker/station and not same worker/station constraints in addition to the task/station eligibility constraints. In our experiment, the soft constraints are either enforced or not.
3. Ergonomic risk:
   This constraint limits the maximum possible weighted ergonomic score for the tasks assigned to any worker. This constraint can take different levels of maximum ergonomic score, or be completely eliminated.

Experimental configuration:

The experiments use a subset of the data in Section 7. The summary of the data sets used with the number of tasks along with the number of different constraints is given in Table 8.

It should be noted that because of the IP's limitation, the maximum number of tasks is limited to 100 to reach a solution in reasonable time. The solution time of the IP is limited to 24 hours and 3600s for the CP; if no optimal solution is found the best solution is recorded along with the time it took the IP or CP to reach that solution. Due to the use of random numbers in CP, the CP experiments are run 10 times for each instance with random number seeds 0 to 9 and the average is recorded along with the number of times the solution was found.

There are two experiments to assess the models' sensitivity. The first experiment varies the maximum weighted ergonomic score allowed for each worker and records both the solution and computation time. The second experiment is conducted by disabling a subset of constraints and recording the solution and computation time. Both of these experiments are done on an i7-4790 CPU with 32 GB on Windows 10 platform.

For the IP, AMPL Version 20161220 is used along with Gurobi 7.0.2 solver. For the CP, IBM ILOG CPLEX Optimization Studio Version 12.7.0 is used.

8.1 Experiment 1: Sensitivity to maximum ergonomic score

The maximum ergonomic score allowed is varied depending on the tasks' ergonomic scores for each band. Tables 9, 10 and 11 show results for bands 1, 26, and 47 respectively. In these tables, the maximum ergonomic score used is shown along with the IP's lower bound (LB) on the solution, the IP solution (optimal is bolded), the computation time to reach this

Table 8 Test data sets: Types of constraints included

Band	# tasks	Number of stations/workers	Task assignment constraints					Mounting zone constraints
			Same station	Same worker	Not same station	Not same worker	Eligibility	
Band 1	50	5/25	0	9	0	102	8	640
Band 26	75	5/25	2	2	0	88	34	1896
Band 47	100	7/35	0	0	2	332	14	650

Table 9 Sensitivity to maximum ergonomic score in Band 1

Exp (Band1)	Max Ergo.	IP			CP		
		LB	Solution	Time (HH:MM:SS)	Solution	# found	Avg time (HH:MM:SS)
E1.1	155	9	9*	00:03:59	9	10	00:00:49
E1.2	160	8	8	00:00:45	8	10	00:00:50
E1.3	165	8	8	00:01:01	8	10	00:00:53
E1.4	170	8	8	00:00:33	8	10	00:00:47
E1.5	175	7	8	00:00:11	7	7	00:38:56
					8	3	00:00:38
E1.6	180	7	7	00:00:33	7	10	00:01:33
E1.7	No limit	7	7	00:00:07	7	10	00:00:36

*Optimal solution is bolded

IP solution, the CP solution(s) (optimal is bolded), the number of time the CP solution is found, and the average CP computation time.

The range of possible maximum ergonomic scores for the different bands increases as the problem size increases. This range might be limited by the possible number of workers in each station. It should be noted that for each band there is a setting for the maximum ergonomic score in which the IP fails to reach the optimal solution even though values below and above this setting allows optimal solutions to be found. For these hard instances, the CP finds the optimal solution in some of the runs with increased computational time. This indicates that the problem becomes harder for some imposed maximum ergonomic levels as the solution pool becomes smaller, perhaps because of the exact values in the data sets. Moreover, relaxing the maximum ergonomic score does not always help in reducing the computation time.

Table 10 Sensitivity to maximum ergonomic score in Band 26

Exp (Band26)	Max Ergo.	IP			CP		
		LB	Solution	Time (HH:MM:SS)	Solution	# found	Avg time (HH:MM:SS)
E26.1	150	12	12	00:05:43	12	10	00:00:57
E26.2	170	10	11	00:02:38	10	3	00:23:54
					11	7	00:00:54
E26.3	180	10	10	00:09:20	10	10	00:01:08
E26.4	200	9	9	00:10:07	9	10	00:00:57
E26.5	250	7	8	00:00:49	7	10	00:09:00
E26.6	350	7	7	00:05:13	7	10	00:01:22
E26.7	No limit	7	7	00:08:26	7	10	00:01:08

Table 11 Sensitivity to maximum ergonomic score in Band 47

Exp (Band 47)	Max Ergo.	IP			CP		
		LB	Solution	Time (HH:MM:SS)	Solution	# found	Avg time (HH:MM:SS)
E47.1	90	32	32	06:47:13	32	10	00:02:17
E47.2	100	29	29	08:03:06	29	10	00:02:17
E47.3	110	26	27	05:32:34	26	5	00:02:18
					27	5	00:02:26
E47.4	120	24	24	05:50:04	24	10	00:02:11
E47.5	130	22	22	08:02:31	22	10	00:02:14
E47.6	140	21	21	05:21:45	21	10	00:02:14
E47.7	150	19	19	10:03:30	19	10	00:02:25
E47.8	160	18	18	03:19:44	18	10	00:02:10
E47.9	170	17	17	07:05:31	17	10	00:02:12
E47.10	180	16	16	10:39:56	16	10	00:02:13
E47.11	200	15	15	05:12:00	15	10	00:02:13
E47.12	300	11	11	09:02:07	11	10	00:02:15
E47.13	500	11	11	11:32:06	11	10	00:02:15
E47.14	No limit	11	11	07:51:35	11	10	00:02:18

8.2 Experiment 2: sensitivity to constraints

In this experiment, the effect of disabling one (or more) soft constraints on the solution quality and computation time is studied. For each band, a set of 8 different instances is generated. The first instance is created by disabling all three soft constraints,. In the next three instances, only one soft constraint is disabled. The following three instances study the

Table 12 Band 1 sensitivity to constraints

Exp	Constraints			IP			CP	
	Ergonomics (Max=165)	Mounting	Assignment	LB	Solution	Time (HH:MM:SS)	Solution	Avg time (HH:MM:SS)
S1.1	✗	✗	✗	7	7	00:00:08	7	00:00:36
S1.2	✓	✗	✗	8	8	00:03:08	8	00:00:37
S1.3	✗	✓	✗	7	7	00:01:05	7	00:00:34
S1.4	✗	✗	✓	7	7	00:00:10	7	00:00:37
S1.5	✓	✓	✗	8	8	07:37:56	8	00:00:40
S1.6	✓	✗	✓	8	8	00:00:51	8	00:00:52
S1.7	✗	✓	✓	7	7	00:00:07	7	00:00:36
S1.8	✓	✓	✓	8	8	00:01:01	8	00:00:53

Table 13 Band 26 sensitivity to constraints

Exp	Constraints			IP			CP	
	Ergonomics (Max=200)	Mounting	Assignment	LB	Solution	Time (HH:MM:SS)	Solution	Avg time (HH:MM:SS)
S26.1	✗	✗	✗	7	7	00:04:59	7	00:00:55
S26.2	✓	✗	✗	9	9	02:28:42	9	00:00:56
S26.3	✗	✓	✗	7	8	00:03:58	7	00:01:21
S26.4	✗	✗	✓	7	7	00:00:51	7	00:00:55
S26.5	✓	✓	✗	7	10	00:04:08	7	00:01:02
S26.6	✓	✗	✓	9	9	01:07:20	9	00:00:52
S26.7	✗	✓	✓	7	7	00:08:26	7	00:01:09
S26.8	✓	✓	✓	9	9	00:10:07	9	00:00:57

Table 14 Band 47 sensitivity to constraints

Exp	Constraints			IP			CP	
	Ergonomics (Max=200)	Mounting	Assignment	LB	Solution	Time (HH:MM:SS)	Solution	Avg time (HH:MM:SS)
S47.1	✗	✗	✗	11	11	12:34:17	11	00:02:07
S47.2	✓	✗	✗	15	15	11:53:30	15	00:02:05
S47.3	✗	✓	✗	11	11	26:06:38	11	00:02:12
S47.4	✗	✗	✓	11	11	03:32:52	11	00:02:04
S47.5	✓	✓	✗	15	16	10:56:52	15	00:02:04
S47.6	✓	✗	✓	15	15	01:17:59	15	00:02:02
S47.7	✗	✓	✓	11	11	07:36:31	11	00:02:08
S47.8	✓	✓	✓	15	15	05:12:00	15	00:02:13

Table 15 Number of IP violations in Band 1

Exp	Constraints			Max ergonomic score	Number of violations	
	Ergonomics (Max=165)	Mounting	Assignment		Mounting worker interference(119)	Assignment (640)
V1.1	✗	✗	✗	299.77	12	36
V1.2	✓	✗	✗	163.95	10	32
V1.3	✗	✓	✗	259.96	0	31
V1.4	✗	✗	✓	326.52	13	0
V1.5	✓	✓	✗	164.56	0	27
V1.6	✓	✗	✓	164.47	5	0
V1.7	✗	✓	✓	215.62	0	0
V1.8	✓	✓	✓	164.47	0	0

Table 16 Number of IP violations in Band 26

Exp	Constraints			Max ergonomic score	Number of violations	
	Ergonomics (Max=200)	Mounting	Assignment		Mounting worker interference(1896)	Assignment (126)
V26.1	✗	✗	✗	385.42	67	36
V26.2	✓	✗	✗	199.83	51	33
V26.3	✗	✓	✗	–*	–	–
V26.4	✗	✗	✓	331.25	48	0
V26.5	✓	✓	✗	–	–	–
V26.6	✓	✗	✓	197.95	42	0
V26.7	✗	✓	✓	403.97	0	0
V26.8	✓	✓	✓	199.98	0	0

*No optimal solution obtained

interaction between the soft constraints by disabling two of them at a time. The last instance is the problem with no constraints disabled for comparison. Tables 12, 13 and 14 summarize the constraints sensitivity results for Bands 50, 26, and 47 respectively.

There is an obvious trend in computation time increase for both the IP and CP as the number of tasks increase from 50 to 75, then to 100 across the different bands. Removing the assignment constraints in Band 1 increased the IP time to reach an optimal solution drastically. In addition, the IP was unable to find an optimal solution within a day after removing the assignment constraints in bands 26 and 47. On the contrary, solving the problem with only the mounting constraints increased the computation time in band 47; no optimal solution was reached in band 26. It should be noted that the effect of disabling all constraints in CP is minimal as the standard deviation in computational time is less than 10 seconds across all three bands.

Table 17

Exp	Constraints			Max Ergonomic Score	Number of Violations	
	Ergonomics (Max=200)	Mounting	Assignment		Mounting worker interference(650)	Assignment (362)
V47.1	✗	✗	✗	456.59	40	41
V47.2	✓	✗	✗	199.97	17	28
V47.3	✗	✓	✗	407.03	0	35
V47.4	✗	✗	✓	365.64	23	0
V47.5	✓	✓	✗	–*	–	–
V47.6	✓	✗	✓	199.79	27	0
V47.7	✗	✓	✓	418.28	0	0
V47.8	✓	✓	✓	199.49	0	0

*No optimal solution obtained

Tables 15, 16 and 17 show the number of IP violations in the line balance incurred by disabling the constraints for bands 1, 26, and 47 respectively. We set the maximum ergonomic score for each band as shown in the 2nd column for those experiments in which the ergonomic constraints are included. In the "Number of violations" column, the total number of mounting and assignment constraints is listed. It should be noted that it is not possible to violate all constraints at the same time, as the total number of constraints represent the number of ways a violation may occur.

8.3 Discussion

In this section, we discuss the sensitivity analysis results obtained for each band.

8.3.1 Band 1 (50 tasks)

The histogram for the ergonomic scores of tasks in band 1 is shown in Fig. 7. Results for band 1 - experiment 1 is shown in Table 15. The average ergonomic score for band 1 is 24.29. The first "maximum ergonomic score" level used for this band is 155, in which an optimal solution is found with 9 workers. Increasing the maximum ergonomic score to 160 reduced the number of workers in the optimal solution to 8. The IP struggles when the maximum ergonomic score is increased to 175 as the lower bound is reduced to 7 workers but the best solution found within the time limit is 8. The CP managed to find the optimal solution in 7 out of 10 replications. The combinatorial problem of finding a combination of tasks with a total score of 175 assigned to 7 workers and conforming to the rest of constraints seems to be harder than other levels of maximum ergonomic score. The ergonomic constraint is not binding if a maximum score limit of 180 or more are used as the solution to the problem is 7 workers which is the lower bound to the problem.

The mounting position used for tasks in band 1 is shown in Fig. 8. Mounting position 0 is for tasks that are sub assembled on the side of the station. The tasks are distributed on the four corners of the car; this is reflected on the lower count of mounting position constraints and violations shown in Table 15. Also, this explains why this band behaved differently when the mounting position constraints are the only constraints enforced as shown in Table 12. The solution time did not change significantly as it did in the other two bands. The biggest impact on solution time occurred when both the mounting and ergonomics constraints are enforced. The IP solution time increased from a minute to over 7 hours. The CP solution time did not change and is consistent across the different instances of experiment 2.

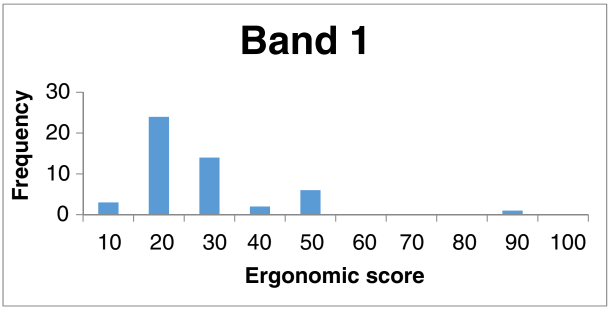

Fig. 7 Band 1 ergonomic score histogram

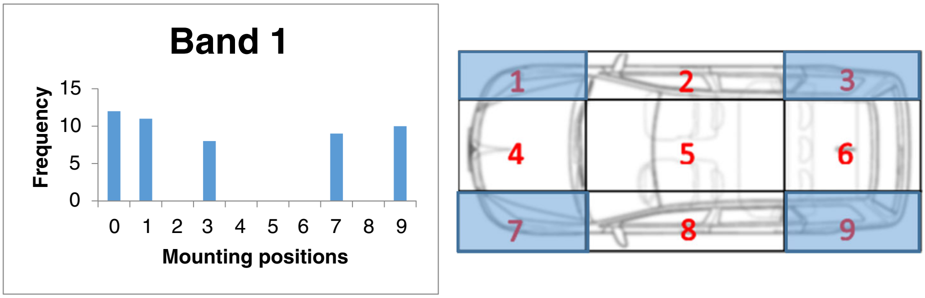

Fig. 8 Band 1 mounting positions histogram

8.3.2 *Band 26 (75 tasks)*

Ergonomic scores of band 26 tasks are shown in the histogram of Fig. 9. The average ergonomic score for band 26 is 22.53. Unlike band 1, band 26 histogram shows a trend in the relation between the number of tasks and ergonomic score. Results for band 26 - experiment 1 is shown in Table 16. Starting maximum ergonomic score for experiment 1 is 150; solving the problem gives an optimal line balance with 12 workers. Increasing the maximum ergonomic score to 170 makes the problem hard to solve for the IP. The number of workers are reduced to 11 but the lower bound is 10. The IP is unable to reach the optimal solution within the time limit. The CP managed to find an optimal line balance with 10 workers in 3 out the 10 replications. In addition, the IP was unable to reach the optimal when the maximum score limit is increased to 250 while the CP managed to get to the optimal in all 10 replications. Increasing the maximum ergonomic score to 350 or more makes this constraint a non-binding constraint.

The tasks in band 26 mostly belong to two mounting positions as shown in Fig. 10. This is the reason behind having so many mounting position constraints. It took the IP 5 minutes to reach the optimal solution without enforcing any of the soft constraints. Adding the ergonomics constraints increased the time to reach the optimal for the IP. The huge number of mounting constraints affected the IP as expected; no optimal solution is found by the IP within the time limit. By enforcing the assignment constraint only the IP solution time is reduced to less than a minute. By enforcing all soft constraints, the IP is able to find an optimal solution in 10 minutes. Disabling the mounting constraints only led to an

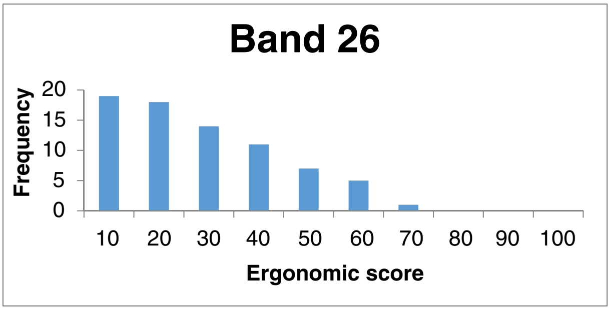

Fig. 9 Band 26 ergonomic score histogram

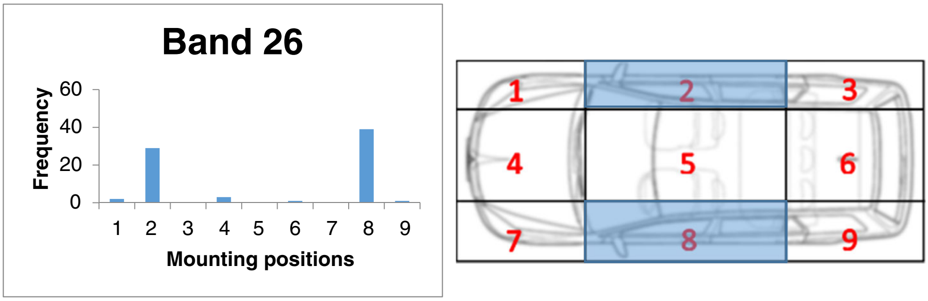

Fig. 10 Band 26 mounting positions histogram

increase in the IP's computation time. In addition, by disabling the assignment constraints, the IP is unable to reach the (relaxed) optimal solution within the time limit.

The number of band 26 violations recorded for experiment 2 is shown in Table 16. The highest number of violations occurred when no soft constraints are enforced. Among the 75 tasks, there are 67 worker interference cases in the line balancing schedule. This number might look small when compared to the number of mounting constraints, but it is not possible to violate all of the constraints at the same time and this number is large relative to the number of tasks. The number of assignments violation is also large relative to the total number of tasks. That is, almost half of the tasks violated the assignment constraint. Enforcing either the ergonomics, the assignment, or both constraints led to a decrease in mounting worker interference violations.

8.3.3 Band 47 (100 tasks)

The histogram for the ergonomic scores of band 47 is shown in Fig. 11. The average ergonomic score for this band is 28.29. The scores for this band are almost uniform with more tasks having a low ergonomic score. This is reflected on the wide range of maximum ergonomic scores for which the IP is able reach an optimal solution. The results of band 47 - experiment 1 is shown in Table 11; the maximum ergonomic score for this band is varied between 90 and 500. The lower maximum ergonomic score used (90) yielded a line balance with 32 workers while the highest level used (300) resulted in a line balance with only 11 workers. This range gives the decision maker a good room to change the maximum ergonomic score based on the balance of cost/workers' health risks. Again for one level of

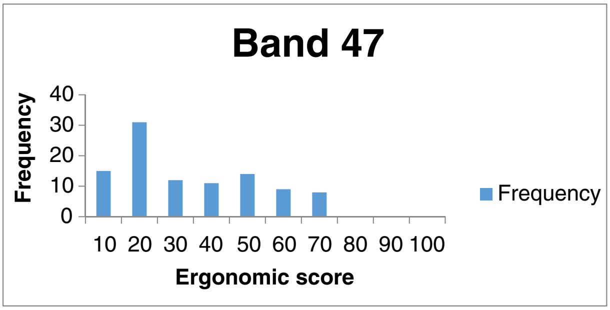

Fig. 11 Band 47 ergonomic score histogram

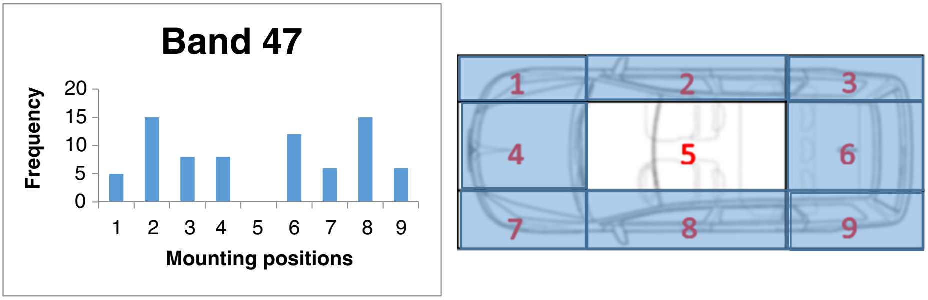

Fig. 12 Band 47 mounting positions histogram

maximum ergonomic score (110), the IP is unable to reach the optimal solution within the time limit. However, the CP reached the optimal solution in 5 out of the 10 replications. The time of CP computations is more consistent when compared to the time of IP computations.

The mounting positions used in band 47 are shown in Fig. 12. In this band, all mounting position are used except for mounting position 5. The results for band 47 - experiment 2 are shown in Table 14. Solving the problem with no soft constraints enforced yielded a line balance with 11 workers. It took the IP over 12 hours to reach the optimal solution; perhaps there are many alternate solutions, and symmetry breaking constraints would help. When the mounting position constraints are the only soft constraints enforced, the IP computation time is doubled when comparing to the solution time when no constraints are enforced. For the case when the mounting position constraints are disabled and the assignment and ergonomic constraints are enforced, the IP computation time is reduced from 12 hours to 1 hour. Also, enforcing only the assignment constraints reduced the IP time to reach the optimal solution. On the other hand, the IP is unable to reach an optimal solution within the time limit when the assignment constraints are removed and the other two soft constraints are enforced. Enabling all constraints reduced the computation time to less than half the case with no constraints enforced.

The number of constraint violations recorded for each instance of band 47 - experiment 2 is shown in Table 17. When disabling all soft constraints, the number of mounting and assignment violations amount to less than half the number of tasks on this band which is still high. In this line balance, one of the workers is assigned tasks with a total weighted ergonomic score of 456.59. By applying only the ergonomic constraints, both the mounting and assignment violations are reduced.

9 Conclusion

In this paper, the model by [3] is extended to the mixed model environment with additional constraints from [12] to improve the model's applicability to real mixed model assembly lines. First, we explain each constraint incorporated into the model and illustrate the problem at hand with an example. Then, we propose an IP formulation to solve the mixed model assembly line balance with parallel stations, zoning constraints, and ergonomics. Since the problem is NP-hard, we develop a scheduling CP model to solve industrial sized assembly line problems.

To test the model's applicability to real assembly lines, the data of three bands from an automotive assembly line were used. The results show the limitation of the IP, in which only instances with 50 tasks were solved to optimality within the allowed 1 hour of processing time. On the other hand, the CP model's results were optimal for 10 out of the 12 instances under the same time limit. For example, the CP managed to balance 178 tasks over 12 stations for band 47. The computational results show the potential of CP scheduler as a tool to solve real life-sized assembly line balancing problems.

Two sensitivity analysis experiments are conducted using data from an automaker. The first experiment studies the effect of changing the maximum ergonomic score limit on the solution quality and computational time. Results show that the problem becomes harder for some levels of maximum ergonomic level limit. For these instances, the IP failed to reach the optimal solution within the allowed time. However, the optimal solution is found in some of the CP runs for each hard instance. The decision maker should be aware of this when deciding on the maximum ergonomic limit. The second experiment studies the effect of disabling a subset of constraints on the solution quality and computational time. Results show that the assignment constraint helps the IP reach the optimal solution within the time limit. By disabling the assignment constrain the IP is unable to reach the optimal in two bands and the computation effort required to reach the optimal increased in one band. Furthermore, using only the mounting constraint led to a huge computational effort to reach the optimal in one band while the optimal is not even reached in one band.

These three bands, while from an operating OEM and hence observed and not designed for this study, provide varied characteristics that may be useful in future research. The bands have different distributions of ergonomic scores and different types of mounting positions that may provide some hypotheses for future research. For example, a further study of the impact of ergonomic score distribution can be conducted assuming all parts are in the same mounting position and only one worker can be assigned to each station; this would extend the concept presented earlier of a two-dimensional bin packing problem enhanced with precedence constraints. As a result of such a study, individual tasks could be targeted for redesign or increased attention, based on their impact on the overall line balance. As another example of a future study, we observe that two bands studied here had disjoint mounting positions while the third had contiguous mounting positions. Further, in band 26, the mounting positions were such that no single person could work in more than one, while the potential for a person to work in more and more mounting positions was moderate in band 1 and extreme in band 47. The types of creative assignments of tasks to workers in the bands with more potential for work being done in adjacent mounting positions may merit further investigation.

This work can be extended in future research by incorporating the idea of horizontal balancing and model sequencing to the mixed model line balancing problem. If a worker is expected to have a notably different amount of time to work on different models, his station might not be able to manage the work overload caused by having several models with time consuming tasks since only the average task duration satisfies the cycle time constraint. There are two ways to address this problem: horizontal balancing and model sequencing. Horizontal balancing makes the workload of each station (with respect to models/options) as uniform as possible over time regardless of the product sequence. This can be seen as a robust mixed model line balance that minimizes work overload regardless of the sequence used. On the other hand, this problem can also be avoided by solving a sequencing problem that prohibits having multiple demanding models in a consecutive sequence and thus avoids any work overload. This can be achieved by developing an integrated model that solves real life-sized line balancing / sequencing problems simultaneously. A challenge in this line of research will be the size of an entire assembly line for a large manufacturer, which may have over 300 stations which are essentially linked by the sequence in which workpieces are released.

References

1. Baybars, I. (1986). A survey of exact algorithms for the simple assembly line balancing problem. *Management Science*, *32*(8), 909–932.
2. Becker, C., & Scholl, A. (2006). A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research*, *168*(3), 694–715.
3. Becker, C., & Scholl, A. (2009). Balancing assembly lines with variable parallel workplaces: problem definition and effective solution procedure. *European Journal of Operational Research*, *199*(2), 359–374.
4. Bockmayr, A., & Pisaruk, N. (2001). Solving assembly line balancing problems by combining IP and CP. In *Proceedings of the 6th annual workshop of the ercim working group on constraints.* Prague.
5. Boysen, N., & Fliedner, M. (2008). A versatile algorithm for assembly line balancing. *European Journal of Operational Research*, *184*(1), 39–56.
6. Boysen, N., Fliedner, M., Scholl, A. (2007). A classification of assembly line balancing problems. *European Journal of Operational Research*, *183*(2), 674–693.
7. Boysen, N., Fliedner, M., Scholl, A. (2008). Assembly line balancing: which model to use when? *International Journal of Production Economics*, *111*(2), 509–528.
8. Boysen, N., Fliedner, M., Scholl, A. (2009). Assembly line balancing: joint precedence graphs under high product variety. *IIE Transactions*, *41*(3), 183–193.
9. Bukchin, Y., & Rabinowitch, I. (2006). A branch-and-bound based solution approach for the mixed-model assembly line-balancing problem for minimizing stations and task duplication costs. *European Journal of Operational Research*, *174*(1), 492–508.
10. Ege, Y., Azizoglu, M., Ozdemirel, N.E. (2009). Assembly line balancing with station paralleling. *Computers & Industrial Engineering*, *57*(4), 1218–1225.
11. Erel, E., & Gokcen, H. (1999). Shortest-route formulation of mixed-model assembly line balancing problem. *European Journal of Operational Research*, *116*(1), 194–204.
12. Falkenauer, E. (2005). Line balancing in the real world. In *Proceedings of the international conference on product lifecycle management PLM* (Vol. 5, pp. 360–370).
13. Gökcen, H., & Erel, E. (1997). A goal programming approach to mixed-model assembly line balancing problem. *International Journal of Production Economics*, *48*(2), 177–185.
14. Gökcen, H., & Erel, E. (1998). Binary integer formulation for mixed-model assembly line balancing problem. *Computers & Industrial Engineering*, *34*(2), 451–461.
15. Kara, Y., Özgüven, C., Seçme, N.Y., Chang, C.T. (2011). Multi-objective approaches to balance mixed-model assembly lines for model mixes having precedence conflicts and duplicable common tasks. *The International Journal of Advanced Manufacturing Technology*, *52*(5-8), 725–737.
16. Mahdavi, I., Javadi, B., Sahebjamnia, N., Mahdavi-Amiri, N. (2009). A two-phase linear programming methodology for fuzzy multi-objective mixed-model assembly line problem. *The International Journal of Advanced Manufacturing Technology*, *44*(9-10), 1010–1023.
17. Öztürk, C., Tunal, S., Hnich, B., Örnek, A.M. (2010). Simultaneous balancing and scheduling of flexible mixed model assembly lines with sequence-dependent setup times. *Electronic Notes in Discrete Mathematics*, *36*, 65–72.
18. Öztürk, C., Tunal, S., Hnich, B., Örnek, A. (2013). Balancing and scheduling of flexible mixed model assembly lines with parallel stations. *The International Journal of Advanced Manufacturing Technology*, *67*(9–12), 2577–2591.
19. Öztürk, C., Tunal, S., Hnich, B., Örnek, M.A. (2013). Balancing and scheduling of flexible mixed model assembly lines. *Constraints*, *18*(3), 434–469.
20. Öztürk, C., Tunal, S., Hnich, B., Örnek, A. (2015). Cyclic scheduling of flexible mixed model assembly lines with parallel stations. *Journal of Manufacturing Systems*, *36*(1), 147–158.
21. Pastor, R., Ferrer, L., García, A. (2007). Evaluating optimization models to solve SALBP. In *International conference on computational science and its applications* (pp. 791–803). Berlin: Springer.
22. Pearce, B.W. (2015). *A study on general assembly line balancing modeling methods and techniques*. Doctoral dissertation: Clemson University.
23. Pil, F.K., & Holweg, M. (2004). Linking product variety to order-fulfilment strategies. *Interfaces*, *34*, 394–403.
24. Sawik, T. (2002). Monolithic vs. hierarchical balancing and scheduling of a flexible assembly line. *European Journal of Operational Research*, *143*(1), 115–124.
25. Scholl, A. (1999). *Balancing and sequencing assembly lines*, 2nd Edn. Heidelberg: Physica.
26. Scholl, A., & Becker, C. (2006). State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research*, *168*(3), 666–693.

27. Sewell, E.C., & Jacobson, S.H. (2012). A branch, bound, and remember algorithm for the simple assembly line balancing problem. *INFORMS Journal on Computing*, *24*(3), 433–442.
28. Vilà, M., & Pereira, J. (2014). A branch-and-bound algorithm for assembly line worker assignment and balancing problems. *Computers & Operations Research*, *44*, 105–114.
29. Wilhelm, W.E., & Gadidov, R. (2004). A branch-and-cut approach for a generic multiple-product, assembly-system design problem. *INFORMS Journal on Computing*, *16*(1), 39–55.