

International Journal of Production Research

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/tprs20

Model and heuristics for the multi-manned assembly line worker integration and balancing problem

Adalberto Sato Michels & Alysson M. Costa

To cite this article: Adalberto Sato Michels & Alysson M. Costa (2024) Model and heuristics for the multi-manned assembly line worker integration and balancing problem, International Journal of Production Research, 62:24, 8719-8744, DOI: 10.1080/00207543.2024.2347572

To link to this article: https://doi.org/10.1080/00207543.2024.2347572

© 2024 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.

Published online: 10 May 2024.

Submit your article to this journal

Article views: 826

View related articles

View Crossmark data

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=tprs20

INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH
2024, VOL. 62, NO. 24, 8719–8744
https://doi.org/10.1080/00207543.2024.2347572

Taylor & Francis
Taylor & Francis Group

OPEN ACCESS Check for updates

Model and heuristics for the multi-manned assembly line worker integration and balancing problem

Adalberto Sato Michels [a,b] and Alysson M. Costa[a,b]

[a]School of Mathematics and Statistics, University of Melbourne, Melbourne, Australia; [b]ARC Training Centre in Optimisation Technologies, Integrated Methodologies, and Applications (OPTIMA), University of Melbourne, Melbourne, Australia

ABSTRACT

This paper examines the balancing of assembly lines with multi-manned stations and a heterogeneous workforce. Both topics received considerable attention in the literature, but not in an integrated fashion. Combining these two characteristics gives rise to a highly combinatorial Multi-manned Assembly Line Worker Integration and Balancing Problem. When considering multi-manned stations, the already coupled decisions on assigning tasks to heterogeneous workers and workers to stations must be further linked with task scheduling assessments. We propose a Mixed-Integer Linear Programming model and develop two heuristic solution procedures, which tackle the problem with a hierarchical decomposition approach. Computational tests on a large dataset indicate that the proposed method can obtain good primal bounds in short computational times. We demonstrate that these results can be applied to the monolithic model either as a warm start or in a proximity search procedure to obtain synergistic gains with statistically significant differences. From a managerial perspective, we show that multi-manned stations can reduce the assembly line's length even in the presence of a heterogeneous workforce, which is crucial for many industries manufacturing large-size products.

ARTICLE HISTORY
Received 3 July 2023
Accepted 10 April 2024

KEYWORDS
Assembly line balancing; multi-manned station; heterogeneous workforce integration; mixed-integer linear programming; decomposition heuristic; proximity search

1. Introduction

In this paper, we propose and study the problem of balancing assembly lines with multi-manned stations while integrating a heterogeneous workforce. This problem can be seen as a generalisation of two research axes in assembly lines, which have received a considerable amount of attention in the literature. Indeed, we combine the Multi-Manned Assembly Line Balancing Problem (MALBP) with the Assembly Line Worker Assignment and Balancing Problem (ALWABP) or, more specifically, with its more generic version, known as Assembly Line Worker Assignment and Integration Problem (ALWIBP).

The MALBP extends the hypothesis of each station in an assembly line employing a single worker (Michels et al. 2019). By allowing more than one worker per station, multi-manned solutions can potentially decrease the number of stations and the workforce size needed to perform a given task set. For industries that manufacture large-size products, such as cars, buses, and airplanes, the potential benefits of line length reduction and production efficiency increase are enough to motivate the use of this strategy.

The ALWABP generalises the assumption that workers are identical and, therefore, that any task can be executed at any station with the same efficiency (Miralles et al. 2008). In the ALWABP, task processing times depend on which worker performs the task, to the extent that some workers may not be able to perform some of them. The ALWIBP is a generalisation of the ALWABP, which considers workers belonging to two sets associated with a heterogeneous and a homogeneous workforce (Moreira, Miralles, and Costa 2015). While Miralles et al. (2008) proposed the ALWABP motivated by assembly lines found in sheltered work centres employing workers with disabilities, Moreira, Miralles, and Costa (2015) went a step further and attempted to integrate these workers with disabilities into regular assembly lines. Hence, the goal in the ALWIBP is to design and balance the line within the medium to long-term planning scope, assigning all workers from the former set whilst minimising the number of identical workers from the latter. Further motivation for studying the problem comes from cyclical or seasonal industries (Corominas, Pastor, and Plans 2008) willing

CONTACT Adalberto Sato Michels a.satomichels@unimelb.edu.au, daru.michels@gmail.com School of Mathematics and Statistics, University of Melbourne, Peter Hall Building, 813 Swanston Street, Parkville VIC 3052, Melbourne, Australia

© 2024 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.
This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited, and is not altered, transformed, or built upon in any way. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

to manage a short-term rebalancing and ramp up their production rate by hiring temporary, unskilled staff to work among permanent, skilled workers in a transitional training period. Additionally, modern lines purchasing collaborative robots (Weckenborg et al. 2020) may conduct short-term interventions to assimilate human-robot cooperation in the assembly line with a long-term goal in mind.

The herein proposed combination of the MALBP and the ALWIBP considers all motivations listed above and gives rise to the Multi-Manned Assembly Line Worker Integration and Balancing Problem (MALWIBP), which is the focus of this paper. In Figure 1, we schematically represent the main decisions involved in the MALWIBP. A directed acyclic graph represents the set of tasks (nodes from 1 to 10) and their precedence relations (edges). The task set is partitioned and each subset of tasks is assigned to a worker as represented by the dashed lines with arrow heads. The workers are then assigned to stations (dashed lines with dot heads). Both of these assignment decisions must respect the partial ordering indicated by the precedence graph. Since stations are multi-manned, an additional scheduling problem must be solved at each station, defining the order in which tasks can be executed and possibly generating idle times. Finally, the load of each station must not exceed the imposed cycle time and depends on the characteristics of the heterogeneous workforce (represented by the different symbols $*$, $\wedge$, $+$, and # associated with some workers), ultimately combining all decisions together.

In this paper, we propose a new Mixed-Integer Linear Programming (MILP) model drawing on the literature from the MALBP and the ALWIBP. We also propose a Hierarchical Decomposition Heuristic (HDH) that builds upon the multiple layers of decisions in the

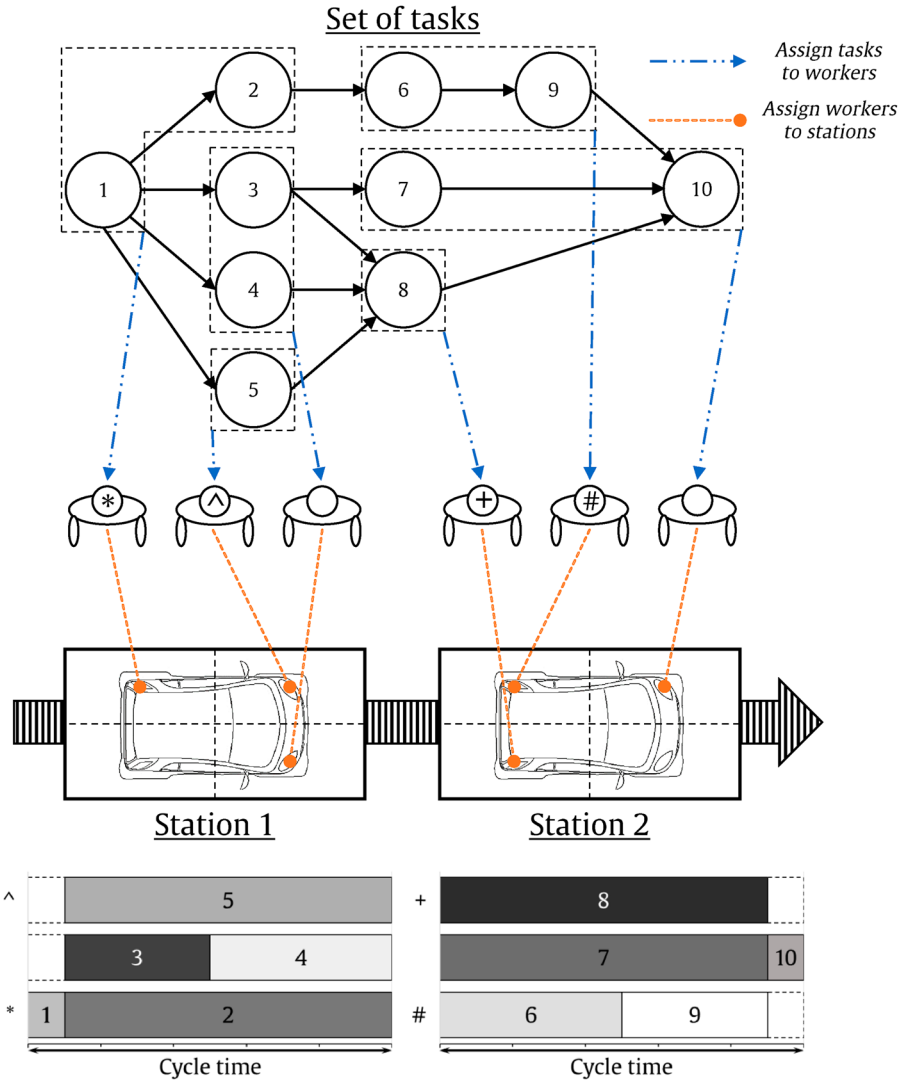


Figure 1. Layout example of a multi-manned assembly line with heterogeneous workers.

problem. Firstly, the HDH finds an assembly line balancing solution that only considers homogeneous workers and does not allow multi-manned stations. A constructive insertion heuristic is then utilised to integrate all available heterogeneous workers among the homogeneous ones. Finally, we apply a MILP-based approach to group workers into multi-manned stations while maintaining the feasibility of the precedence constraints. This process generates feasible MALWIBP solutions with the heterogeneous workers set entirely employed and lean line configurations. Lastly, the quality of the solutions is further improved by using warm start procedures and a large-scale MILP search based on the idea of proximity search (Fischetti and Monaci 2014).

The remainder of this paper is organised as follows. In Section 2, the relevant literature for both the MALBP and the ALWIBP is discussed, focussing on the objective functions, extensions, and methodologies that have been investigated so far. In order to formally define the problem, Section 3 precisely delineates the characteristics of a MALWIBP and presents a new MILP model for it. A brief discussion on the modelling decisions of the problem's combinatorial aspects is conducted, and valid inequalities are introduced to strengthen the model's linear relaxation. Furthermore, a motivating example numerically demonstrates the advantages associated with using a multi-manned configuration. Section 4 describes all steps of the developed heuristic, its use as a warm start for the proposed MILP model, and an iterative local search heuristic based on proximity search principles (Fischetti and Monaci 2014). Extensive computational results are conducted on a literature benchmark dataset and discussed in Section 5. Finally, in Section 6, a series of concluding remarks and further research directions end the paper.

2. Literature review

Assembly lines are production systems widely applied to manufacturing industries with a high-volume output of standardised products. Their product-oriented layouts are generally built to fit flow-shops, which conveniently enables the mass production of homogeneous goods. The literature on Assembly Line Balancing Problems (ALBP) was comprehensibly surveyed by Battaïa and Dolgui (2013) and, more recently, by Battaïa and Dolgui (2022) to address new trends and formulations concerning the hybridisation of ALBP extensions. The ALBP computational complexity was analysed by Álvarez Miranda and Pereira (2019) and Álvarez Miranda, Pereira, and Vilà (2023), categorising it as a NP-hard problem, since it subsumes the bin-packing problem as a special case. Furthermore, a systematic review of research themes (Eghtesadifard, Khalifeh, and Khorram 2020) and a chronological survey from the last fifteen years (Boysen, Schulze, and Scholl 2022) were also recently published in the literature.

The core decisions in assembly line balancing have been detailed by Baybars (1986), who considers several simplification hypotheses to define the most basic version of the problem, the Simple Assembly Line Balancing Problem (SALBP). The SALBP consists of deciding the assignment of a task set to stations. Tasks cannot be arbitrarily assigned to any station; they must respect precedence constraints imposed by a precedence graph as shown in the top part of Figure 1. Moreover, stations are organised in a straight, serial line. Such layout enforces the line's pace to respect the most loaded station's total processing time, which is known as the cycle time. Finally, the assembly line produces a unique model of a single product. The most common goals in a SALBP are the minimisation of the number of stations (SALBP-1) and of the cycle time (SALBP-2).

Crucial contributions to the SALBP literature were previously surveyed by Scholl and Becker (2006). They list various exact algorithmic solution approaches and practical applications. Among those, SALOME is an effective bidirectional branch-and-bound (B&B) procedure developed by Scholl and Klein (1997) to solve the SALBP. More recently, Sewell and Jacobson (2012) proposed a branch, bound, and remember (BBR) algorithm and Morrison, Sewell, and Jacobson (2014) subsequently added a new backtracking rule to save memory in their own BBR variant. They proved optimal solutions for 86% of unsolved problems in the benchmark dataset proposed by Otto, Otto, and Scholl (2013). These and other techniques were gathered in an overview and further improved for the SALBP-1 by Pape (2015).

The SALBP has been widely extended to incorporate features of assembly lines found in industry, as reviewed in Becker and Scholl (2006) and Boysen, Fliedner, and Scholl (2007, 2008). These generalisations are proposed to cope with features such as task incompatibilities, space constraints, different line layouts, equipment selection, unproductive times, among others. These extensions of the SALBP are usually classified under the umbrella of General Assembly Line Balancing Problems (GALBP). Becker and Scholl (2006) presents a review particularly focussed on GALBPs.

In our context, two generalisations of interest extend the following SALBP hypotheses: (i) a single worker must be assigned per station and (ii) any task can be processed at any station with the same task processing time. These generalisations lead to multi-manned assembly lines and workforce heterogeneity, respectively reviewed in Sections 2.1 and 2.2.

2.1. The multi-manned assembly line balancing problem

In many industrial contexts, considering that only a single worker can be assigned to a station is artificially restrictive. Contemplate, for example, the automotive industry: large-size products, such as cars and buses, force the work-piece sizes and station physical spaces to be much larger than the working area needed by an operator. As a consequence, the number of available positions in a station is directly tied to the product size. This makes it possible (and often desirable) to assign more than one worker to each station and have them perform tasks simultaneously on the same product.

A pioneering work in multi-manned assembly lines is that of Akagi, Osaki, and Kikuchi (1983). They allowed more than one worker in each station with a two-phase approach denominated Parallel Assignment Method (PAM). In the first phase of PAM, tasks are assigned to stations under upper time limits. As multiple workers can be assigned to a station, PAM assigns tasks to workers in each station in the second phase. Later on, Dimitriadis (2006) proposed a heuristic method building on the work of Hoffmann (1963) in order to minimise the number of stations as equivalent to the total space requirement. Dimitriadis (2006) showed that the heuristic is helpful in enhancing the physical utilisation of stations. It requires a fixed cycle time and considers the goal of worker and station minimisation, which has since then been the most usual goal function employed in a variety of works.

In practice, we must ensure that workers do not interfere with each other when simultaneously performing tasks on the same work-piece. However, it might not be easy to designate workplace positions arbitrarily, as done in the MALBP. To address this issue, the two-sided assembly line balancing problem (TALBP) assumes right and left workplaces in each station, allowing at most two operators (Bartholdi 1993). Some tasks must exclusively be performed on the right or left side of the product, while others can be performed from either side. Similarly, the assembly line balancing problem with variable workplaces (VWALBP) defines a grid of mounting positions associated with the work-piece and the station space around it Becker and Scholl (2009). In order to avoid interference, once a task is assigned to a worker in a given station, only the remaining tasks which share the same or neighbouring mounting positions with the assigned tasks can be assigned to such worker. Cycle times are given and each position of the multi-manned station can only receive a single worker, while the aim is to minimise the needed working areas (i.e. number of workers in the line). Becker and Scholl (2009) formulated the VWALBP as a MILP model and proposed lower bounding techniques. The authors also implemented the so-called VWSolver (a branch-and-bound algorithm based on SALOME) to solve large-size benchmark instances. Kellegöz and Toklu (2012), in turn, explore a more general situation in which tasks occupying the same mounting position can be assigned to different workers of the same station by evading their simultaneous operations. They developed a novel efficient branch-and-bound algorithm called Jumper, which outperformed a modified (less restrictive) version of VWSolver both in solution quality and computational processing times. Their method can be readily adapted to solve a VWALBP by imposing a time-incompatibility to all task pairs requiring at least one joint mounting position.

Fattahi and Roshani (2011) were the first to propose a mathematical model using the goals defined by Dimitriadis (2006) in a hierarchical way, with the minimisation of workers given higher priority than the minimisation of stations, formally defining the problem's variant known as MALBP-1. They also developed an Ant Colony Optimisation (ACO) algorithm to tackle instances that could be not solved with their model. Kellegöz (2017) improved the MALBP-1 MILP model of Fattahi and Roshani (2011) and proposed a Gantt-based heuristic within a Simulated Annealing (SA) algorithm, which outperformed the previously presented ACO algorithm, finding better feasible solutions for most instances in the evaluated benchmark. Naderi, Azab, and Borooshan (2019) and Michels et al. (2019) developed and applied Benders' decomposition algorithms (Benders 1962) with combinatorial Benders' cuts (Codato and Fischetti 2006) to the MALBP-1 variant. Naderi, Azab, and Borooshan (2019) could solve a realistic five-sided MALBP with moving workers and limited workspace while Michels et al. (2019) solved the majority of medium and large-size MALBP-1 benchmark instances to optimality. More recently, Possan Junior, Michels, and Magatão (2024) proposed a Constraint Programming (CP) model and stronger lower bounds in an integrated MILP-CP procedure to provide nine new optimality proofs to the same benchmark dataset.

For a given fixed number of stations, Roshani and Giglio (2017) defined a type-II variant of the MALBP, in which the minimisation of the cycle time and number of workers are the primary and secondary goals, respectively. They proposed a mathematical formulation and two metaheuristics based on SA. These heuristics were called the indirect and direct SA, hereafter referred to as ISA and DSA. The DSA performed better both in terms of solution quality and computational processing time. Michels, Lopes, and Magatão (2020) tackled the problem exactly and proposed a new MILP model, which

was used within a Combinatorial Benders' decomposition approach to solve large MALBP-2 instances to optimality. Yang et al. (2022) extended the MALBP-2 by considering positional constraints associated with the worker assignments. A Filtered Beam Search (FBS) algorithm with dominance rules and lower bounds was developed for large-sized instances.

The MALBP was studied with a multi-objective approach by Roshani et al. (2013). They adapted a SA algorithm to maximise line efficiency while minimising line length and a smoothness index. Roshani and Nezami (2017) also used a SA algorithm, this time to solve a MALBP-1 mixed-model variant. Huo and Lee (2023) also solved the MALBP-1 mixed-model variant, this time with a Benders' decomposition algorithm. Kellegöz and Toklu (2015), in turn, presented a constructive heuristic based on priority rules with an improvement procedure based on GA. They performed computational experiments on MALBP benchmark instances, intending to minimise the total number of workers employed along the line. Yilmaz and Yilmaz (2015) aimed at concurrently minimising the number of workers, stations, and workload difference between workers while Yilmaz and Yilmaz (2016) analysed the impact of considering skilled workers and equipment requirements on MALBP solutions with classified teams. For this purpose, a mathematical model is developed for the problem and a heuristic procedure is used to solve it.

Giglio et al. (2017) considered the MALBP with a skilled workforce (MALBP-SW). In this extension, each worker earns a different salary depending on their capabilities to perform tasks (Moon, Logendran, and Lee 2009). They developed a mathematical formulation to represent the MALBP-SW in a cost-oriented model with the objective of minimising the assembly line's total operating cost. In addition to the topic of cost-oriented models, Kazemi and Sedighi (2013), Michels et al. (2018), and Zangaro, Minner, and Battini (2023) examined assembly lines with multi-manned stations regarding cost-oriented problem instances on real-size case studies. Kazemi and Sedighi (2013) presented a Genetic Algorithm to minimise costs per production unit. Michels et al. (2018) formulated a MILP to design a robotic assembly line that considers the employment of multiple robotic workers per station, minimising the combined cost of robots, facilities, and tool prices. The compilation of Michels et al. (2018, 2019) and Michels, Lopes, and Magatão (2020) was also published in a thesis as a collection of articles (Michels 2020). Finally, Zangaro, Minner, and Battini (2023) proposed the Joint Multi-manned Assembly Line Balancing and Feeding Problem (JMALBFP). They minimise supermarket, transportation, operations, and investment costs with a MILP model and an Adaptive Large Neighbourhood Search (ALNS) heuristic.

Contemporary MALBP variants that have attracted quite a lot of attention in the past couple of years are the consideration of task processing time variability and demand uncertainty with alternative modelling approaches. Yang and Cheng (2020) proposed the MALBP with forward and backward sequence-dependent setup times (MALBPS). Besides establishing a MILP model to characterise the problem, they developed an SA algorithm to solve it. Zhang et al. (2022) explored the MALBPS with a nature-inspired metaheuristic. Their experiments showed that their method is an effective method to solve the problem when compared to a set of other metaheuristics. Güner et al. (2023) tackled the sequence-dependent setup times on MALBPs with a CP approach applied to a real-world workforce scheduling problem. Non-deterministic task processing times were taken into account by Zakaraia, Zaher, and Ragaa (2022) in a U-shaped MALBP. They formulated it as a MILP with stochastic task processing times (random variables with known means and variances) and cycle time chance-constraints. A Differential Evolution Algorithm (DEA) was proposed to solve the problem. At last, Andreu-Casas, García-Villoria, and Pastor (2022) took into account a realistic aspect of multi-manned stations – the possibility of interference between workers – and dealt with it by considering task processing time increments depending on the number of parallel workers in a station. In order to solve this MALBP variant, they developed a mathematical model, two relax-and-fix procedures, and a heuristic based on solving a partition problem with constraints. Zhang, Tang, and Chica (2021) and Zhang et al. (2024) considered mixed-model lines with uncertain demand to propose a robust MILP, a gene expression programming based on heuristic rules, and a reinforcement learning-based multi-objective evolutionary algorithm.

Other recent papers on MALBP variants are chronologically listed in the following. Sahin and Kellegöz (2019a) considered resource investment and developed a hybrid heuristic based on Particle Swarm Optimisation (PSO) to tackle the problem. Sahin and Kellegöz (2019b), Şahin and Kellegöz (2022) and Hashemi-Petroodi et al. (2022, 2023) took into account the possibility of walking workers in the line. The former solved large instances with a reduced integer gap by applying an Electromagnetic Field Optimisation (EFO) algorithm and a Benders' decomposition based exact solution method to the problem, while the latter further extended the problem by considering a mixed-model environment and employed a Constructive Matheuristic (CM), a Fix-and-Optimise Heuristic (FOH), and a Markov Decision

Process (MDP) to deal with large-scale instances. Cantos Lopes, Michels, and Magatão (2019) proposed a model-based hierarchical decomposition procedure to minimise the number of stations after the worker-task assignment solution has been decided in a multi-manned assembly line. Cantos Lopes et al. (2019) brought more flexibility to multi-manned lines by explicitly considering a continuous paced line control instead of an unpaced one, relaxing the limitation of fixed, discrete, and restrictive frontiers between stations. Their model, heuristic procedure, and algorithmic lower bounds were able to produce significantly shorter line lengths when compared to previous results. Yilmaz and Yilmaz (2020) proposed a mathematical model to consider assignment restrictions, i.e. positive and negative zoning, distance, station, and synchronous task restrictions. They also developed a tabu-search algorithm to solve the problem effectively. Lastly, Yadav, Kumar, and Agrawal (2021) improved worker and station efficiencies of a real-life automotive plant by reconfiguring the assembly line under study.

By adopting the modelling decisions that reduce the number of variables and constraints (Kellegöz 2017; Michels et al. 2019), this paper aims to extend the classical MALBP to deal with heterogeneous workforces, which broadens the idea of skilled workers dealt by Yilmaz and Yilmaz (2016) and Giglio et al. (2017). In brief, we generalise the ability to classify workers into categories such as fast and slow or into sets of workers earning different salaries to consider each worker individually. This has important practical implications, as reviewed in Section 2.2.

2.2. The assembly line worker integration and balancing problem

The ALWABP considers assembly lines with a set of heterogeneous workers. The same task might have very different processing times depending on the worker who performs it, to the extent that some workers may not be able to execute a given task. Miralles et al. (2008) originally proposed the problem to model assembly lines in sheltered work centres employing workers with disabilities. The flexibility and applicability of their model made this research quickly gain momentum in the literature. This worker assignment concept deserved several mentions in a recent review (Katiraee et al. 2021) investigating workers' differences in production systems modelling and design regarding skills, age, gender, and anthropometric measures.

Various studies have tackled the ALWABP, with both heuristic and exact approaches. An example of a fast heuristic is presented in Moreira et al. (2012), in which priority rules are defined for both task and worker assignments. These metrics are incorporated in a stand-alone method and also as a decoder in a biased random-key genetic algorithm. Other examples of successful metaheuristics are the genetic algorithm from Mutlu, Polat, and Supciller (2013), the probabilistic beam search of Borba and Ritt (2014) and the two-phase Variable Neighbourhood Search from Polat et al. (2016). These methods are able to quickly generate good quality feasible solutions for instances of realistic size.

Exact methods based on branch-and-bound were also studied in the literature. Borba and Ritt (2014) presented a task-oriented branch-and-bound algorithm while Vilà and Pereira (2014) proposed a station-oriented branch, bound, and remember algorithm. The methods are somewhat complementary, with the task-oriented branch-and-bound being more efficient on instances with a higher order strength, for which the branch, bound, and remember algorithm faced memory limitation issues.

Moreover, some authors extended the ALWABP to incorporate practical characteristics found in real-world assembly lines. For instance, the treatment of disabled-friendly job rotation strategies (Costa and Miralles 2009), different line layouts, such as parallel stations/lines (Araújo, Costa, and Miralles 2012, 2015) and U-shaped lines (Oksuz, Buyukozkan, and Satoglu 2017), the production of mixed-models (Cortez and Costa 2015; Ramezanian and Ezzatpanah 2015; Zhang et al. 2024), robotic workers (Borba, Ritt, and Miralles 2018), multi-manned stations with cycle time minimisation (Yang et al. 2022), rebalancing scenarios (Michels and Costa 2022), lines with hierarchical workers having different costs (Campana, Iori, and Moreira 2022; Sungur and Yavuz 2015), workload allocation with chain precedence constraints (Pereira and Ritt 2023), and uncertainty on worker availability (Liu et al. 2021; Ritt, Costa, and Miralles 2016) or task processing times (Moreira et al. 2015; Pereira 2018) were considered.

As employment is one of the main paths for communal inclusion, the basis for personal stability, and a key way to participate as a citizen in modern societies, the welfare of people with disabilities strongly depends on the degree of labour integration they may achieve. Therefore, our work in this paper falls into the framework of incorporating workforce heterogeneity in more generic assembly lines (Moreira, Miralles, and Costa 2015; Moreira et al. 2017). In particular, we study the extension of the ALWABP in which two sets of workers are considered. Some workers are homogeneous, executing tasks with virtually identical processing times, while the second set of workers is heterogeneous, as in the ALWABP. This version of the problem is known as the Assembly Line Worker Integration and Balancing Problem (ALWIBP). It represents a further step in meeting the requirements

to integrate workers with disabilities into ordinary workplaces, granting production managers with practical tools to facilitate this process in the most effective manner. In this context, these workers are no longer segregated to sheltered work centres but instead incorporated into conventional assembly lines. In this paper, we extend the ALWIBP to consider multi-manned stations, as formally defined in Section 3.

3. Formal definitions and a mathematical formulation

This section formally describes the MALWIBP, presents the proposed mathematical formulation to represent it (Section 3.1) along with valid inequalities (Section 3.2) used to strengthen the linear relaxation of the model, and demonstrates the advantages of using the multi-manned flexibility with a motivating example (Section 3.3).

We consider a set S of stations sequentially ordered. Each station can only process one work-piece at a given time. An unpaced control is used to move these pieces forward between stations within a fixed and known cycle time limit ($\overline{C}$), whilst their moving times between stations are neglected. In a steady-state scenario, the production rate is determined by measuring how much time one unit of the finished product takes to emerge from the last station along the line. As the line produces a unique product model, the most loaded station exclusively determines the line's pace and, consequently, its production output.

The line must perform a set N of tasks to assemble a given product. Tasks are indivisible and the order of their execution must respect a set R of precedence relations. The set W represents the available workers. The first element of this set corresponds to identical homogeneous workers, while each of the other elements corresponds to a specific heterogeneous worker. Each task i has a different deterministic processing time (t_{wi}) dependent on the worker w who executes it. Furthermore, Patterson and Albracht (1975) proposed the implementation of the earliest and latest stations at which task i can occur (ES_i and LS_i, respectively), which reduces the search space by limiting the task assignment to occur in a lower range of stations. Adaptations of this technique were proposed for the presence of heterogeneous workers (Michels and Costa 2022) and multi-manned stations (Possan Junior, Michels, and Magatão 2024) by considering the lowest task processing times among workers, i.e. $t_{1i},\ \forall\, i \in N$.

In the scenario associated with the ALWIBP, a reasonable aim for a production manager is to fully integrate the given set of workers with disabilities whilst minimising the number of additional homogeneous workers and stations needed for achieving the desired throughput rate. Hence, we assume that all heterogeneous workers must be employed along the line. Moreover, we can take into account the possibility of parallel work within a station, with P representing the set of possible positions in a station to execute such parallel work. Each position constitutes an arbitrary physical space within a station and the maximum number of available positions in each station is tightly related to the products' size.

Table 1 summarises the notation used in this paper. In addition to the parameters described above, the table also lists tuples indicating the positions available for each station (SP), the stations, workers, and positions to which a task can be assigned (SN, WN, and PN, respectively), and the positions available for each worker depending on the station (SWP). These tuples are used to simplify the presentation of the model in Section 3.1. Furthermore, they readily allow incorporating more practical extensions and functional limitations when these modifications are desirable. In the same way that we define infeasible assignments (due to worker-task incompatibility for heterogeneous workers in WN), machine positioning or task zoning features could be analogously restricted by sparsing SN and PN tuples. For instance, if a particular tool is necessary to perform a given task and it is inevitably physically located at a given station, the definition of an appropriate tuple SN restricts the

Table 1. Notation of indexes, parameters, and sets.

Index	Meaning
s	Station index
w	Worker index
p	Position index
i,j	Task indexes
Parameter	**Meaning**
m	Maximum number of stations
d	Number of heterogeneous workers
l	Number of positions per station
n	Number of tasks
$\overline{C}$	Cycle time limit
t_{wi}	Task processing time for worker w to perform task i
ES_i	Earliest station at which task i can occur: $ES_i = \lceil \frac{(t_{1i}+\sum_{j\in P_i} t_{1j})}{\overline{C}} \rceil,\ \forall\, i \in N$
LS_i	Latest station at which task i can occur: $LS_i = m + 1 - \lceil \frac{(t_{1i}+\sum_{j\in S_i} t_{1j})}{\overline{C}} \rceil,\ \forall\, i \in N$
M_i	A sufficiently large natural number for task i: $M_i = \overline{C} \cdot LS_i,\ \forall\, i \in N$
Set	**Meaning**
S	Stations set: $S = \{1, \ldots, m\}$
W	Workers set: $W = \{1, \ldots, d\}$, 1 for homogeneous workers and 2–d for heterogeneous workers
P	Positions set: $P = \{1, \ldots, l\}$
N	Tasks set: $N = \{1, \ldots, n\}$
R	Precedence relations set: $R = \{(i,j) \in N \mid i \prec j\}$
S_i	All direct and indirect successor of task i
P_i	All direct and indirect predecessor of task i
SP	Tuple with Station-Position feasible elements
SN	Tuple with Station-Task feasible elements
WN	Tuple with Worker-Task feasible elements
PN	Tuple with Position-Task feasible elements
SWP	Triplet with Station-Worker-Position feasible elements

station-task assignment variables and properly includes this viable practical requirement. Additionally, we can limit task positioning on product sides by applying the same procedure. An example of such limitation is when a task must necessarily be performed at a specific region of the product (e.g. head and tail lights for vehicles), which may not be conceivable if workers are assigned to the opposite side. Finally, due to space inaccessibility, we might make the *SWP* triplet sparse by pondering infeasible stations or positions for workers. Real-life interpretations of such inaccessible spaces are stairs, underground stations, or areas blocked by heavy machinery.

3.1. The MALWIBP model

This section describes the developed MILP model to represent the MALWIBP using the notation presented in Table 1. A summary of the decision variables is presented in Table 2, which are created depending on their sets and tuples. Our proposed model builds upon the ALWIBP model by Moreira, Miralles, and Costa (2015) and the MALBP model by Michels et al. (2019).

Expression (1) states the objective function considered for this problem. In line with the common strategy used in the literature, it hierarchically minimises the total number of workers employed along the assembly line (main goal) and the total number of opened stations (secondary goal). Note that the light weight assigned to the secondary goal ensures that the hierarchy is respected.

$$\text{Minimise}: \underbrace{\sum_{(s,w,p)\in SWP} y_{swp}}_{workers} + \underbrace{\frac{1}{m+1}\cdot\sum_{s\in S} o_s}_{stations} \quad (1)$$

In the model's task assignment core, Constraints (2) enforce each task to be assigned to a station within the earliest to latest station range (Patterson and Albracht 1975) to restrict the search space. Constraints (3) and (4) ensure each task is exclusively performed by one worker and at a specific position, respectively. Worker assignment is partly conducted by Constraints (5), which guarantee that any position in each station receives at most one homogeneous or heterogeneous worker. Constraints (6) state that all heterogeneous workers ($w > 1$) must be assigned exactly once. By not applying this constraint to the homogeneous worker index ($w = 1$), we allow the model to use as many homogeneous workers as necessary to satisfy the productivity constraints. In conjunction with the objective function, these constraints keep the number of homogeneous workers used in the line at a minimum to fit our model in the worker integration framework. Constraints (7) are logical constraints that ensure the correct meaning of y_{swp} variables: if task i is assigned to station s ($x_{si} = 1$), to worker w ($z_{wi} = 1$), at position p ($q_{pi} = 1$), then these constraints force worker w to be assigned to position p of station s ($y_{swp} = 1$)

Balancing core

$$\sum_{s\in S|ES_i\le s\le LS_i} x_{si} = 1 \quad \forall\, i \in N \quad (2)$$

$$\sum_{w\in W} z_{wi} = 1 \quad \forall\, i \in N \quad (3)$$

$$\sum_{p\in P} q_{pi} = 1 \quad \forall\, i \in N \quad (4)$$

$$\sum_{w\in W} y_{swp} \le 1 \quad \forall\, (s,p) \in SP \quad (5)$$

$$\sum_{(s,p)\in SP} y_{swp} = 1 \quad \forall\, w \in W \mid w > 1 \quad (6)$$

$$y_{swp} \ge x_{si} + z_{wi} + q_{pi} - 2 \quad \forall\, (s,i) \in SN,\ (w,i) \in WN,\ (p,i) \in PN \quad (7)$$

To model the problem's scheduling core, we define binary *task ordering* variables f_{ij} indicating that task i is executed after task j in the proposed solution. Additional continuous *task starting time* variables e_i indicate the execution start time of a task $i \in N$. The constraints associating task ordering and starting variables make sure precedence relations are respected: Constraints (8) force task j to follow task i if the precedence set R contains such a pair of tasks (i,j). Constraints (9) and (10) properly decide ordering variables depending on task assignment and worker employment with logical ties. Constraints (9) work at station level and ensure that task i follows task j

Table 2. Notation of the model's variables.

Variable	Set	Domain	Meaning
x_{si}	$(s,i) \in SN$	$\{0,1\}$	Station-task assignments: 1 if station s performs task i
z_{wi}	$(w,i) \in WN$	$\{0,1\}$	Worker-task assignments: 1 if worker w performs task i
q_{pi}	$(p,i) \in PN$	$\{0,1\}$	Position-task assignments: 1 if task i is performed on position p
y_{swp}	$(s,w,p) \in SWP$	$\{0,1\}$	Station-worker-position tie: 1 if station s employs worker w on position p
o_s	$s \in S$	$\{0,1\}$	Station opening decisions: 1 if station s is used
e_i	$t \in T$	$\mathbb{R}_+$	Starting time of the execution of task i
t_i	$t \in T$	$\mathbb{R}_+$	Actual task processing time to perform task i
f_{ij}	$i,j \in N \mid i \ne j$	$\{0,1\}$	Ordering variables: 1 if task i is followed by task j

if it is assigned to a later station in the line while Constraints (10) define precedence relations within the same station: if tasks i and j are performed at the same position p of the same station s, then one of them must follow the other. Unlike the ALWIBP model (Moreira, Miralles, and Costa 2015), worker-task assignments (z_{wi}) are explicitly defined here. Therefore, instead of relying on big-M expressions, Constraints (11) can compute the actual task processing time (t_i) of task i by capturing the appropriate worker execution time (t_{wi}) according to the active worker assignment variable (z_{wi}). Constraints (12) bind the execution starting time of any task i to a minimum value (lower bound) in accordance to which station s it has been assigned. Constraints (13) limit the maximum execution starting time (upper bound) of a task i assigned to station s. Note that we enhanced the previous MALBP model scheduling core (Michels et al. 2019), which used a big-M formulation and did not limit the station range for Constraints (12) and (13) to help with model convergence. Constraints (14) state that all tasks i must be finished up within the line makespan (number of stations multiplied by the cycle time). Finally, all tasks i that precede j must be finished before the execution of task j starts, as required by Constraints (15). These last constraints require a sufficiently large natural number (M_i) to be effective. A simplistic valid value for M_i could be given by $\overline{C} \cdot LS_i$, where LS_i is the latest station at which task i can be performed. In the case of Constraints (14), a tighter value can be obtained without compromising the linearity of the model by using the actual number of opened stations in the current solution.

Scheduling core

$$f_{ij} = 1 \quad \forall\, (i,j) \in R \tag{8}$$

$$f_{ij} \geq x_{sj} - \sum_{k \in S | k \geq s} x_{ki} \quad \forall\, s \in S,\ i,\ j \in N \tag{9}$$

$$f_{ij} + f_{ji} \geq x_{si} + x_{sj} + q_{pi} + q_{pj} - 3 \quad \forall\, s \in S,\ p \in P,\ i,\ j \in N \mid i \neq j \tag{10}$$

$$t_i \geq \sum_{w \in W} t_{wi} \cdot z_{wi} \quad \forall\, i \in N \tag{11}$$

$$e_i \geq \overline{C} \cdot \sum_{s \in S | s \geq ES_i} (s-1) \cdot x_{si} \quad \forall\, i \in N \tag{12}$$

$$e_i + t_i \leq \overline{C} \cdot \sum_{s \in S | s \leq LS_i} s \cdot x_{si} \quad \forall\, i \in N \tag{13}$$

$$e_i + t_i \leq \overline{C} \cdot \sum_{s \in S} o_s \quad \forall\, i \in N \tag{14}$$

$$e_j \geq e_i + t_i - M_i \cdot (1 - f_{ij}) \quad \forall\, i,\ j \in N \mid i \neq j \tag{15}$$

3.2. Valid inequalities

Model (1)–(15) represents the MALWIBP. To strengthen the problem's linear relaxation, we propose valid inequalities based on the ordering of stations, workers, and positions to reduce the natural symmetries of the model (Walsh 2006). Constraints (16) and (17) force stations to be open once there is a task or a worker assigned to it. Moreover, symmetry-breaking Constraints (18) enforce that stations are opened sequentially. Similarly, symmetry-breaking Constraints (19) impose that a worker can only use a position in a given station if the previous position is already in use; they have an equivalent effect of sequentially hiring workers in the MALBP because, in both cases, we assume that there is no interference between workers. These constraints were developed by Kellegöz (2017) and Michels et al. (2019) to accelerate their MILP model convergence and preliminary tests demonstrated that they are useful for our specific case as well.

$$o_s \geq x_{si} \quad \forall\, (s,i) \in SN \tag{16}$$

$$o_s \geq y_{swp} \quad \forall\, (s,w,p) \in SWP \tag{17}$$

$$o_s \leq o_{s-1} \quad \forall\, s \in S \mid s > 1 \tag{18}$$

$$\sum_{w \in W} y_{swp} \leq \sum_{w \in W} y_{sw(p-1)} \quad \forall\, (s,p) \in SP \mid p > 1 \tag{19}$$

Note that Constraints (18) and (19) are only applied to the model from the second station/position onwards, as it was analogously done by Kellegöz (2017) and Michels et al. (2019) for homogeneous workers. Also, note that these constraints could not be applied to the model if we had specific machinery fixed at certain line segments or were to extend the problem to deal with variable workplaces and product sides.

Henceforth, the MILP model stated by the minimisation of (1) and subject to (2)–(19) defines the proposed MALWIBP. We hereafter refer to this formulation as M1. Following the same logic as in Kellegöz (2017) to assess the problem's complexity, we know that it is impossible for the number of workers or stations in a line to be greater than the total number of tasks (n) in a MALWIBP. Otherwise, we would have idle workers or empty stations with no task processing time assigned to them. Additionally, the number of positions allowed to perform tasks at any given station cannot be greater than n either. In practice, however, this number of available positions would be further constrained by the product size. Under these values, with big-O notation, the proposed M1 requires the order of $O(n^3)$ binary variables and $O(n^4)$ constraints to be formulated.

3.3. A motivating example

We use the toy problem represented in Figure 2 to numerically illustrate the difference between an ALWIBP solution and a MALWIBP solution. The instance has 5 tasks and a single heterogeneous worker (indicated by *). The top part of Figure 2 presents the associated precedence graph for these 5 tasks and also shows the task processing times t_{wi} represented by the values on the nodes' top right corner. The first value indicates the execution time when the task is assigned to one of the homogeneous workers, while the second value indicates the execution time if the task is assigned to the heterogeneous worker $*$. Notice that tasks 1 and 5 are incompatible with worker $*$ capabilities and, therefore, are not feasible assignments (–). The bottom part of Figure 2 portrays optimal solutions for both the ALWIBP and the MALWIBP with a defined cycle time ($\overline{C} = 10$).

Since each station s has more than a single position, stations are permitted to be occupied by more than one worker in the multi-manned line. Workers therein assigned may simultaneously work on the same product. The admitted maximum number of positions (l, with $l = 2$ for the illustrative instance) at each station depends on the work-piece characteristics (mostly, on its size). The ALWIBP's optimal layout requires four workers assigned to four stations. In this case, the idle time totals 11-time units, i.e. 27.5% of the line's available productive time. The blank spaces in each station represent such idle time.

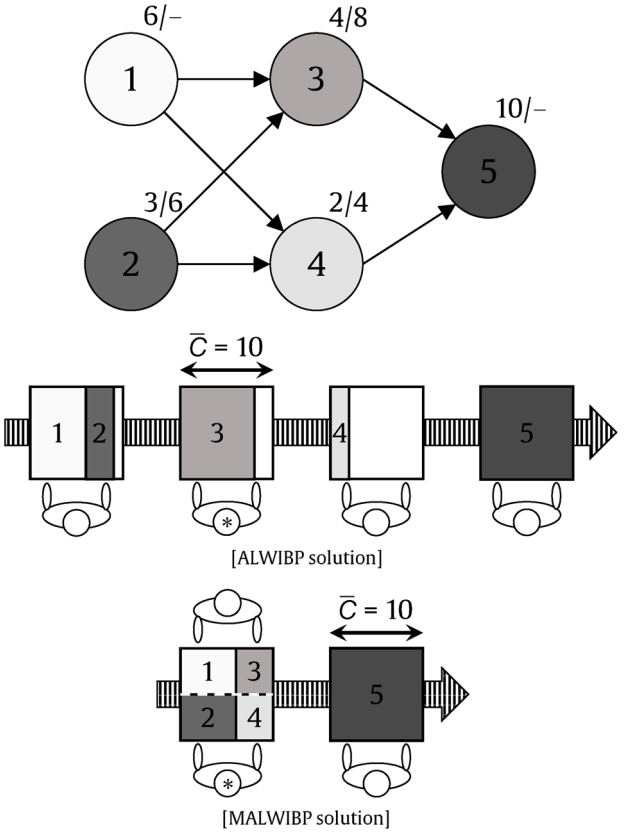


Figure 2. Motivating example solutions for the ALWIBP and the MALWIBP.

With this simple example, it is possible to verify some advantages of multi-manned lines with worker integration over its simpler counterpart: by allowing more than one worker per station, the MALWIBP solution could (i) reduce the line length from four to two stations, (ii) assign three workers instead of four to perform the task set, and (iii) reduce the idle time.

Unfortunately, the computational burden in solving the problem comes along as a drawback with this viable advantage. Notice that the task indivisibility aspect is still constraining the problem and must be respected. Whilst workers placed in different positions can synchronously perform non-coincident tasks, they still must satisfy all precedence relations imposed by the precedence graph. Hence, whenever a station employs more than one worker, a task scheduling problem arises on top of the double assignment problem. In such situations, we can expect waiting times for workers to occur before or between the task's execution. As remarked by Fattahi and Roshani (2011), this added combinatorial complexity makes MALBPs more challenging to solve than SALBPs, which are known to be members of the NP-hard class (Álvarez Miranda and Pereira 2019; Álvarez Miranda, Pereira, and Vilà 2023). This complexity is further increased by the addition of heterogeneous workers, limiting the applicability of off-the-shelf solvers directly to the proposed model for larger instances and making it hard to scale. To cope with this problem, we propose a heuristic solution method, described in Section 4.

4. Hierarchical decomposition heuristic

The proposed Hierarchical Decomposition Heuristic (HDH) for the MALWIBP is herein presented. Section 4.1 presents the necessary steps to quickly and effectively solve the single-manned counterpart of the problem (ALWIBP) presented in Figure 2, whilst Section 4.2 explains the developed MILP-based HDH to construct feasible solutions that use multi-manned stations. Section 4.3 describes the post-optimisation process to input a warm start solution into the proposed mathematical formulation (M1) and summarises our method in a flowchart (Figure 4).

In the current implementation, the first step is associated with decisions concerning task and worker assignments to stations (Algorithms 1 and 2). The intermediate step fixes worker-task assignments from the ALWIBP solution and attempts to minimise the number

of required stations (Algorithm 3). Finally, the last step is related to the entire problem, where M1 receives the feasible solution obtained by the HDH thus far as a warm start (Algorithm 4). It then seeks to enhance such a solution and prove its optimality.

Another option for a post-optimisation process is the use of a local search heuristic (Algorithm 5). More specifically, as exhibited in Section 4.4, we iteratively conduct minimal improvements to the HDH's solution with a proximity search procedure (Fischetti and Monaci 2014).

4.1. Solving the ALWIBP

A fast heuristic to solve the ALWIBP has previously been developed by Moreira, Miralles, and Costa (2015). The method is named Constructive Insertion Heuristic (CIH), and it relies on existing similarities between the ALWIBP and the classical SALBP. In general terms, a SALBP-1 solution is initially found (Section 4.1.1) and then iteratively adapted to integrate all heterogeneous workers from the available workforce (Section 4.1.2). Finally, the procedure goes through a post-optimisation routine, as described in Section 4.1.3.

4.1.1. Solving the SALBP

The initial step to obtaining a SALBP solution for the ALWIBP is to ignore heterogeneous workers. In other words, we exclusively take care of station-task assignments (x_{si}) and station opening decisions (o_s) in this step. For the size of the proposed dataset instances (Section 5), the SALBP can be solved by any heuristic or exact approach that yields optimal or high-quality feasible solutions in a few seconds, such as SALOME (Scholl and Klein 1997) or BBR algorithms (Morrison, Sewell, and Jacobson 2014; Sewell and Jacobson 2012). In this study, we opted for the best solution found by SALOME by running Algorithm 1.

Algorithm 1: SALOME execution pseudo-code.

```
Input  : Standard ALB format data
Output: SALBP solution
while x̄_si = ∅ ∧ ō_s = ∅ do
    Run SALOME with N, S, C̄, t_1i
    //following these instructions
    if opt_sol ∨ time_limit then
        x̄_si ← x*_si
        ō_s ← o*s
        //store optimal or best
          found SALBP solution
    end
end
```

4.1.2. Constructive insertion heuristic

Once in possession of a feasible SALBP solution, the CIH divides the current line into segments and attempts to assign a heterogeneous worker to a station contained in the segment under analysis. Starting from the station in which a heterogeneous worker was integrated, the remaining part of the solution is rebuilt using multiple priority rules based on a SALBP constructive heuristic (Scholl and Voss 1997). Lastly, CIH chooses the best solution among those obtained for each station-worker assignment, with the entire heterogeneous workforce performing tasks (z_{wi}), assigned to some station (y_{swp}), and the updated total number of used stations along the line for the ALWIBP solution.

4.1.3. Post-optimisation process

A MILP-based neighbourhood search procedure can improve the solution obtained by the CIH. Under this post-optimisation process, we take two simplifications into consideration. Firstly, worker assignment variables for both homogeneous and heterogeneous workers are fixed to what had been found by the CIH. This simplification allows changes to occur exclusively in task assignment variables. Additionally, the required computational effort to solve the model might further decrease by limiting task assignment variables' freedom during the post-optimisation process. Moreira, Miralles, and Costa (2015) implemented this procedure by only permitting tasks to be assigned to either the same station it had been assigned in the CIH solution or adjacent stations. Given the fact that considering more options would mostly lead to infeasible assignments (due to precedence relations constraints), such simplification is reasonable. In combination with the heuristic's past steps, this post-optimisation process is named CIH-LS1 and presented in Algorithm 2.

4.2. MILP-based heuristic

The second phase of the proposed heuristic consists of using the ALWIBP solution obtained by the CIH-LS1 (explained in Section 4.1) to generate feasible multi-manned solutions. Section 4.2.1 describes the parameters to be input in the model presented in Section 4.2.2. A numerical example based on the illustrative instance (Figure 2) is also depicted.

4.2.1. Input parameters

In broad terms, the entire CIH-LS1 solution – with task and worker assignment decisions – contains all the input parameters for the proposed HDH. Using the notation introduced in Tables 1 and 2, the information we

Algorithm 2: CIH-LS1 execution pseudo-code.

Input : SALBP reference solution
Output: ALWIBP solution
while $\bar{y}_{sw} = \varnothing$ **do**
  Run CIH-LS1 with $N, S, W, \overline{C}, t_{wi}$, and $\bar{x}_{si}, \bar{o}_s$ from the SALBP solution
  //according to Moreira, Miralles, and Costa (2015) formal description
  if *opt_sol* $\vee$ *time_limit* **then**
    $\bar{x}_{si} \leftarrow x^*_{si}$
    $\bar{y}_{sw} \leftarrow y^*_{sw}$
    $\bar{o}_s \leftarrow o^*_s$
    //store optimal or best found ALWIBP solution
    $\bar{z}_{wi} \leftarrow \lceil x^*_{si} + y^*_{sw} - 1 \rceil \quad \forall s \in S,\ w \in W,\ i \in N$
    //assign appropriate values to $\bar{z}_{wi}$, since we know that task i is assigned to station s and worker w is also employed in the same station s, then it is possible to induce which worker w is performing task i
  end
end

can acquire from the ALWIBP solution includes station-task assignments (x_{si}), station-worker-position assignments (y_{swp}), and the actual execution time of tasks (t_i). As there is only one worker per station ($l = 1$) in this more straightforward case, we can simplify the y_{swp} variable and introduce a station-worker assignment variable: y_{sw}. Moreover, if we know which tasks are performed and which worker is employed in a given station, it is possible to infer worker-task assignments (z_{wi}).

Given this ALWIBP solution, a simple heuristic manner of converting it into a MALWIBP solution is to keep the same number of workers along the line, whilst evaluating if a solution with fewer stations can be obtained. In other words, we can find the minimum number of stations needed to employ all workers with their task assignments from the previous solution, while still respecting precedence relation constraints and cycle time limits in every station. This alternative method to discover such a minimum number of stations defines a residual multi-manned problem with a heterogeneous workforce. From the simpler version of the problem, worker-task assignments (z_{wi}) and actual execution time of tasks (t_i) are extracted as parameters, whereas station-worker assignment decisions (and consequently station-task assignments) remain as variables of the problem. As discussed earlier, we must conduct the scheduling of tasks within stations to guarantee precedence relations and cycle time constraints are respected.

This idea of focussing on station minimisation alone in MALBPs is discussed by Cantos Lopes, Michels, and Magatão (2019) for a homogeneous workforce. The reasoning behind this approach is that, empirically, it is extremely unlikely that MALBPs will reduce the number of workers when compared to SALBPs. In Michels et al. (2019), all instances were solved with a reduced number of stations exclusively, which indicates a tendency that it is more profitable to accept the SALBP optimal solution as the number of workers and try to minimise the line length as much as possible, i.e. the number of multi-manned stations. A piece of evidence to support this methodology is that the possibility of also reducing the number of workers in a multi-manned assembly line presented was not verified in any instance of the benchmark. Moreover, as demonstrated in Michels, Lopes, and Magatão (2020), although this MILP-based heuristic does not ensure optimality for any type of MALBP, it has been successfully applied to the MALBP-1 instances with homogeneous workers by Cantos Lopes, Michels, and Magatão (2019) and Possan Junior, Michels, and Magatão (2024), as it yielded optimal or near-optimal solutions in quite reduced computational times. As evidenced by this literature, using such worker-task assignment as an initial solution for an approach that explores the entire search space can drastically improve faster convergences for exact algorithms. Lastly, Possan Junior, Michels, and Magatão (2024) also demonstrated that the SALBP-1 solution is generally a valid lower bound for the classic MALBP-1.

In Section 4.2.2, we adapt Cantos Lopes, Michels, and Magatão (2019)'s model to fit the workforce heterogeneity concept into it. The residual problem is defined with the following parameters. For each task i from set N, let $\bar{t}_i$ be the actual task processing time parameter, which is taken from a reference solution obtained by the ALWIBP method. As we know that each task i is solely assigned to a previously determined worker w, worker-task assignments (z_{wi}) can be converted to a ω_i vector, which contains the homogeneous or heterogeneous worker that performs each task i. This procedure allows us to keep the information on which tasks are performed by each heterogeneous worker. For the illustrative example in Figure 2, parameters for task processing times and worker-task assignments are respectively set to $\bar{t}_i = \{6, 3, 8, 2, 10\}$ and $\omega_i = \{1, 1, 2, 3, 4\}$.

4.2.2. Station minimisation model

Given the notation from Section 4.2.1 and Tables 1 and 2, the following model (20)–(29) formally defines the residual station minimisation problem (S_{min}).

$$\text{Minimise}: \sum_{s \in S} o_s \tag{20}$$

subject to:

$$\sum_{w \in W} y_{sw} \leq l \quad \forall s \in S \tag{21}$$

$$\sum_{s \in S} y_{sw} = 1 \quad \forall w \in W \tag{22}$$

$$e_i \geq \overline{C} \cdot \sum_{s \in S} (s-1) \cdot y_{s\omega_i} \quad \forall i \in N \tag{23}$$

$$e_i + \bar{t}_i \leq \overline{C} \cdot \sum_{s \in S} s \cdot y_{s\omega_i} \quad \forall i \in N \tag{24}$$

$$e_j \geq e_i + \bar{t}_i \quad \forall (i,j) \in R \tag{25}$$

$$e_j \geq e_i + \bar{t}_i - M_i \cdot f_{ij} \quad \forall i, j \in N \mid \omega_i = \omega_j,\ i \neq j \tag{26}$$

$$e_i \geq e_j + \bar{t}_j - M_i \cdot (1 - f_{ij}) \quad \forall i, j \in N \mid \omega_i = \omega_j,\ i \neq j \tag{27}$$

$$o_s \geq y_{sw} \quad \forall (s,w) \in SW \tag{28}$$

$$o_s \leq o_{s-1} \quad \forall s \in S \mid s > 1 \tag{29}$$

The objective function is denoted by Expression (20): it attempts to minimise the total number of required stations to fit the entire workforce. Constraints (21) and (22) respectively state that at most l workers may be assigned to each station and that each worker must be assigned to exactly one station. Constraints (23) and (24) require tasks to be performed within the station where their worker is assigned. Furthermore, they impose cycle time restrictions. Precedence relation requirements are imposed by Constraints (25). Constraints (26) and (27) enforce each worker to perform at most one task at any given time and that the order defined by f_{ij} is observed. Finally, valid inequalities are adapted for this residual problem: Constraints (28) and (29) respectively state that workers can only be assigned to opened stations and stations must be opened in order of their indexes.

The procedure described in Sections 4.1 and 4.2 defines the proposed HDH. Therefore, in combination with Algorithms 1 and 2 being sequentially executed, Algorithm 3 covers the last step in the HDH to find a feasible MALWIBP solution.

Finally, given the ALWIBP solution shown in Figure 2 as the input, the HDH's output is illustrated in Figure 3. Notice that, although worker-task assignments have not been changed from the initial ALWIBP solution, the HDH managed to shorten the line length by one station.

Algorithm 3: MILP-based heuristic pseudo-code.

```
Input  : ALWIBP reference solution
Output: HDH's feasible MALWIBP solution
while ō_s = ∅ do
    Run S_min with N, S, W, C̄, t̄_i, ω_i
    //according to Cantos Lopes,
     Michels, and Magatão (2019)
     procedure
    if opt_sol ∨ time_limit then
        ē_i ← e*_i
        ō_s ← o*_s
        Infer x̄_si from e*_i and C̄
        Infer q̄_pi from y*_sw and z̄_wi
        //store the best found
         MALWIBP feasible solution
    end
end
```

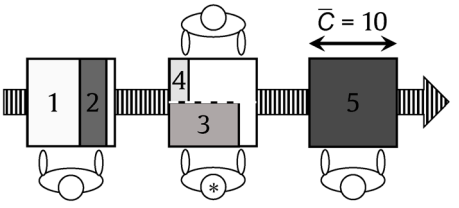


Figure 3. Illustrative solution obtained by the HDH.

This improvement was possible because both workers from the second and third stations could be grouped together into a single one without compromising precedence relations or cycle time restrictions. This feasible multi-manned solution includes task scheduling decisions related to position-task assignments (q_{pi}) and execution starting times for each task (e_i). Thus, it can be used as a warm start to the M1 in the sequence, as explained in Section 4.3.

4.3. Warm start procedure

Once in possession of a feasible MALWIBP solution, a warm start procedure can be prompted into M1: the decision variables obtained by the HDH are used as a tentative initial solution in M1. When the model loads the HDH's solution (e.g. Figure 3), it establishes a valid primal bound (i.e. the best found upper bound) for the problem as a consequence. The optimal MALWIBP solution for the illustrative instance (Figure 2) can then be quickly found by M1 within a compacted search space.

The combination of M1 using the HDH solution as an input (i.e. the warm start procedure) is henceforth

referred to as WS-MIP. Algorithm 4 describes this last step to further explore the entire search space and attempt to provide an optimal solution for the MALWIBP.

Algorithm 4: Warm start procedure pseudo-code.

Input : HDH reference solution
Output: WS-MIP MALWIBP solution
while *opt_sol* ∨ *time_limit* **do**
  Solve M1 with $N, S, W, P, \overline{C}, t_{wi}$, and $\bar{x}_{si}, \bar{z}_{wi}, \bar{q}_{pi}, \bar{e}_i, \bar{o}_s$ as a warm start initial solution
  `//this yields the final balancing and scheduling solution`
    $x_{si} \leftarrow x^*_{si}$
    $z_{wi} \leftarrow z^*_{wi}$
    $q_{pi} \leftarrow q^*_{pi}$
    $y_{swp} \leftarrow y^*_{swp}$
    $e_i \leftarrow \mathrm{e}^*_i$
    $o_s \leftarrow o^*_s$
  `//store optimal or best found MALWIBP solution`
end

Alternatively, Figure 4 summarises the proposed solution method, intuitively showing which sets and parameters are required for each step of the process. Furthermore, Figure 4 also evidences a hierarchy in the multiple layers of decisions found in the MALWIBP. We first decide on an initial station-task assignment only using the task processing times of homogeneous workers and then employ heterogeneous workers, who are integrated by the CIH-LS1 algorithm. Having made this worker-task assignment decision, we allow multi-manned stations, and the HDH solves the task scheduling part of the problem. Finally, using the WS-MIP, we attempt to solve the entire combination of balancing and scheduling problems with a feasible warm start solution. Advantageous synergies for this last step process are demonstrated in Section 5.

4.4. Proximity search procedure

We also propose an additional post-optimisation strategy based on a local search heuristic introduced by Fischetti and Monaci (2014), namely proximity search (PS). This procedure is a general approach focussed on improving a given feasible reference solution. Algorithm 5 lists the steps taken by our adapted strategy to solve the MALWIBP in the PS framework. Variants of this PS framework were also developed by Boland et al. (2016) and Rodrigues et al. (2021). The former presented the Proximity Benders (PB), a PS-inspired heuristic approach for two-stage mixed-integer linear stochastic programming models, while the latter introduced the concept of weighted Hamming distance, allowing the design of a new method called weighted proximity search (WPS). When we can give up on optimality proofs and the emphasis is particularly bestowed on finding high-quality solution, the PB combines two structures that naturally complement each other. Complementarily, there are static and dynamic strategies to determine weights in the WPS. They may help us to distinguish between alternative solutions in the neighbourhood, which provides guidance to the solver when locating an improved solution.

Among practical applications of this PS paradigm, Fischetti and Monaci (2016) proposed a heuristic for the turbine layout optimisation in a wind farm. Their computational results on large scale instances proved the practical viability of the overall approach. Recently, Ahmed et al. (2022) addressed the robust integrated airline fleet assignment, aircraft routing, and crew pairing problem with a matheuristic consisting of a decomposition approach and a proximity search algorithm. They carried out computational experiments on real-world instances from an airline company, providing evidence of the proposed approach efficacy.

The proximity search approach requires the definition of a proximity function to be minimised. A natural option is to use a proximity function that penalises a solution according to its distance from the incumbent solution, i.e. the Hamming distance. All binary variables or a subset of them can be used to define the distance function. After preliminary tests, we decided to include the binary variables regarding assignment and station opening decisions in the proximity function (30) for the proposed MALWIBP.

$$\begin{aligned}\text{Minimise}: \quad & \sum_{(s,i)\in SN|\bar{x}_{si}=0} x_{si} + \sum_{(s,i)\in SN|\bar{x}_{si}=1} (1-x_{si}) \\ & + \sum_{(w,i)\in WN|\bar{z}_{wi}=0} z_{wi} + \sum_{(w,i)\in WN|\bar{z}_{wi}=1} (1-z_{wi}) \\ & + \sum_{(p,i)\in PN|\bar{q}_{pi}=0} q_{pi} + \sum_{(p,i)\in PN|\bar{q}_{pi}=1} (1-q_{pi}) \\ & + \sum_{(s,w,p)\in SWP|\bar{y}_{swp}=0} y_{swp} + \sum_{(s,w,p)\in SWP|\bar{y}_{swp}=1} (1-y_{swp}) \\ & + \sum_{s\in S|\bar{o}_s=0} o_s + \sum_{s\in S|\bar{o}_s=1} (1-o_s) \end{aligned} \tag{30}$$

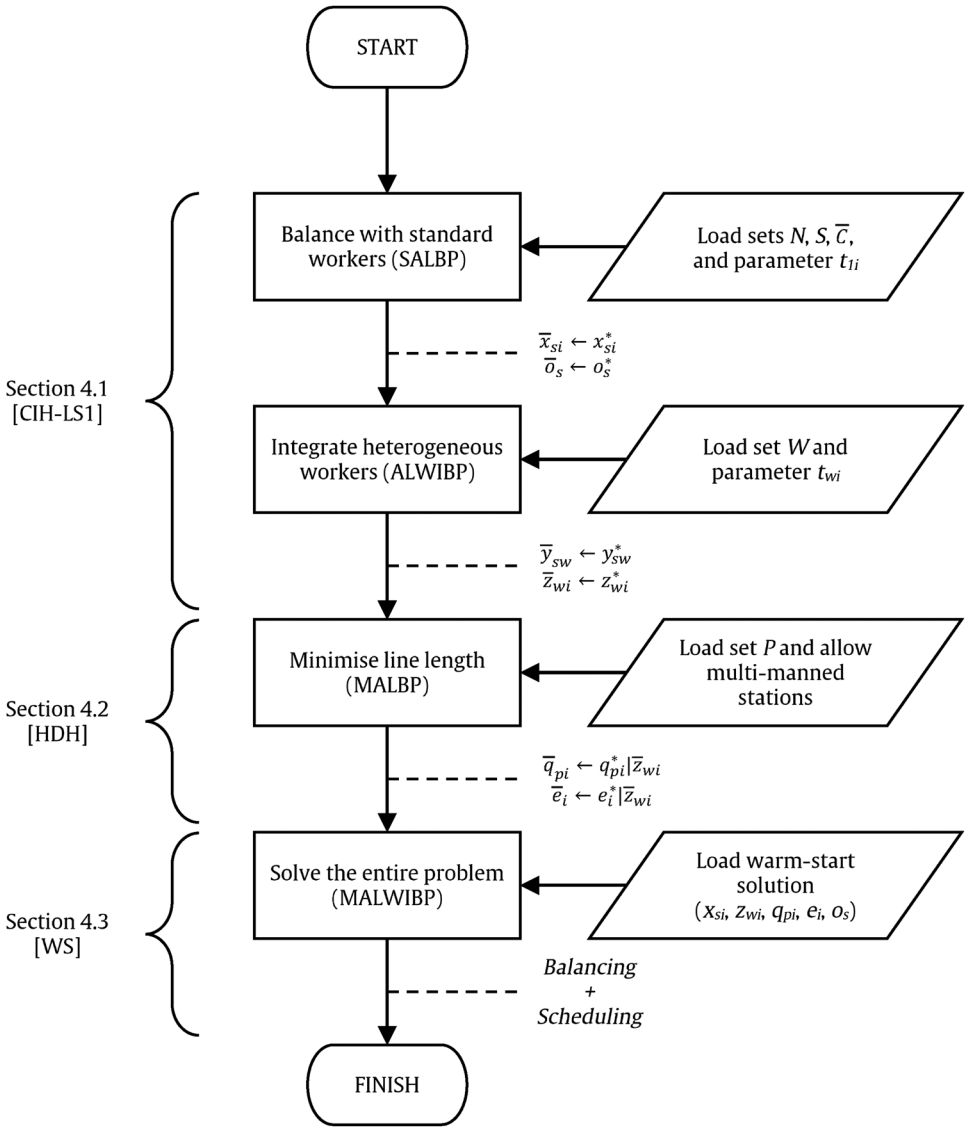


Figure 4. Flowchart of the proposed solution method.

We then apply a universal solver to the modified problem, hoping to find a nearby solution with fewer stations than the one previously found by the HDH. To this end, we also must add an explicit cutoff constraint (31) to the original MILP, where $\varepsilon > 0$ is a given tolerance that specifies the required marginal improvement. With the problem's knowledge, we can implement ad-hoc modifications by fixing the total number of workers to the HDH's value and setting $\varepsilon = 1$ to this minimum improvement, which is the subsequent marginal decrement in the number of stations.

$$\sum_{s \in S} o_s \leq \sum_{s \in S} \overline{o}_s - \varepsilon \quad \forall\, s \in S \tag{31}$$

Finally, as summarised in Algorithm 5, our proposed PS operates iteratively. We try to improve the HDH solution step by step with a PS approach until any termination condition is met, i.e. either if the time limit is reached or if we detect an infeasibility. Whenever an improved feasible assignment is located, we use this new incumbent solution to recentre the PS. Although this procedure is not an exact method, optimality can be guaranteed whenever the PS finds an incumbent solution whose improvement is deemed infeasible by the model.

5. Computational study

This section carries out computational studies based on the benchmark dataset proposed by Moreira, Miralles, and Costa (2015), aiming to assess combinatorial aspects of the newly proposed MALWIBP and evaluate outcome changes. This dataset for the ALWIBP was generated from a subset of precedence graphs created by Otto, Otto, and Scholl (2013). Firstly, a subset of graph structures

Algorithm 5: Proximity Search pseudo-code.

```
Input  : HDH reference solution
Output: PS MALWIBP solution
Initialisation: Stopping_Criteria ← FALSE
while Stopping_Criteria = FALSE do
    Solve M1 with the Hamming distance
      objective function (30) and cutoff
      constraint (31)
    if infeasible ∨ time_limit then
        Stopping_Criteria ← TRUE
    else
        Recentre with improved feasible solution
    end
end
```

containing a mix of chains and bottlenecks is chosen. From them, instances with low (0.2) and high (0.9) order strength (OS) are selected. Lastly, regarding distributions of task processing times, the subsets with peak at the bottom (PB) and bimodal time (BI) distributions are picked. Afterwards, cases with low or high variability of task processing time ($U[t, 2t]$ or $U[t, 5t]$) and percentage of incompatibilities (10% or 20% of tasks) for heterogeneous workers are generated for each original instance. For the ALWIBP benchmark scheme (Moreira, Miralles, and Costa 2015), the cycle time limit ($\overline{C}$) is fixed to 1000 time units, and instances with one, two, three, or four heterogeneous workers are included, totalling 1600 instances for each problem size: medium (50 tasks) and large (100 tasks).

The current dataset, however, has to extend the previous ones by considering the possibility of employing multiple workers in the same station. In order to test the MALWIBP's combinatorial aspect, we expanded the current dataset to investigate stations with up to 2 or 4 positions, totalling 3200 cases for each instance size. Instead, a more compact dataset with medium-size instances for the studied problem is herein proposed. For diversity purposes, we take into account 10 precedence graphs for each category. Moreover, instances with a high variability of task processing time ($U[t, 5t]$) are discarded. The reasoning behind this decision is that high processing times for a given worker-task pair in the MALWIBP case have the actual effect of making those tasks infeasible for some heterogeneous workers. For a better analysis of the results, we prefer to set the number of infeasible tasks directly, using the parameter indicated by 10% or 20% of tasks being incompatible for heterogeneous workers. Finally, instances considering 4 heterogeneous workers were left out due to their marginal contribution regarding the problem's combinatorial aspect. Therefore, the tested and reported dataset contains 120 instances for each precedence graph category, totalling 480 instances. For the HDH, the entire 3200-instance benchmark dataset has been solved. These detailed results for all instances are made available at the research data.

The resulting list of instances is summarised in Table 3 and the reader can refer to this paragraph as a glossary. The computational study is split into two parts. Firstly, 480 medium-size instances for the Multi-manned Assembly Line Worker Integration and Balancing Problem (MALWIBP) are solved using the proposed monolithic Model (M1), the Hierarchical Decomposition Heuristic (HDH), the Warm Start (WS-MIP) procedure, and the Proximity Search (PS) procedure in Section 5.1. These instances are divided into several categories with different graph structures (binomial – BI and peak at the bottom – PB), order strengths (low – 0.2 and high – 0.9), incompatibilities (10% and 20% of tasks for heterogeneous workers), number of heterogeneous workers (1 to 3), and number of allowed positions in each station (2 and 4). An in-depth analysis is conducted for these cases according to the instance's size and structure. Besides, results are compared to those obtained by Moreira, Miralles, and Costa (2015)'s Constructive Insertion Heuristic with post-optimisation phase (CIH-LS1) for the Assembly Line Worker Integration and Balancing Problem (ALWIBP). The second part regards the 3200 large-size instances (Section 5.2), which are exclusively solved by the HDH. Improvements concerning the CIH-LS1 are reported in terms of line length reduction.

In both Sections 5.1 and 5.2, we perform the Kruskal–Wallis test by ranks. These tests are applied to HDH, WS-MIP and PS in Section 5.1 and between CIH-LS1 and HDH in Section 5.2 to account for the statistical significance of the results. A significant Kruskal–Wallis H test indicates that at least one sample stochastically dominates one other sample.

For every instance, Gurobi 9.1.2 (Gurobi Optimization 2019) was employed as the universal solver. In order to allow a fair comparison, one thread of a 64-bit Intel Core^{TM} i7-8550U CPU (1.8 GHz) with 16GB of RAM was used to run the implementations. The M1, HDH, WS-MIP, and PS, as well as their interactions with data files, were coded in Microsoft Visual Basic 2019 programming language.

5.1. Medium-size instances

In this section, both M1, WS-MIP, and PS were applied to the medium-size instances (50 tasks) presented in Table 3 with a time limit set to 1800 s. In contrast, the HDH was solved with a time limit of 60 s for all 3200 instances.

Table 3. Summary of dataset instances.

Problem size (Total of instances)	Time distribution	Order strength	Incompatibility	Variability	# of Heterogeneous workers	# of Positions
Medium (480)	BI, PB	0.2, 0.9	10%, 20%	U[t,2t]	1, 2, 3	2, 4

Table 4. Computational results for M1, HDH, WS-MIP, and PS under the parameter size analysis.

			Δ				CPU(s)				Gap(%)			
Het	Inc(%)	Pos	M1	HDH	WS-MIP	PS	M1	HDH	WS-MIP	PS	M1	HDH	WS-MIP	PS
1	10	2	12	7	17	16	1385.6	12.0	1156.3	663.7	5.8	4.5	4.0	4.3
		4	14	1	16	13	1454.5	12.0	1344.2	778.3	5.4	4.5	3.7	4.3
	20	2	12	9	19	17	1375.3	12.3	1015.1	550.2	6.2	4.2	4.0	4.0
		4	9	4	17	15	1553.7	12.3	1316.3	618.3	6.4	4.2	4.1	4.1
2	10	2	3	9	12	15	1729.5	12.1	1395.6	956.2	9.2	5.9	5.8	5.8
		4	2	6	11	13	1767.6	12.1	1692.4	1162.6	9.3	6.0	5.8	5.8
	20	2	5	9	13	14	1740.7	9.8	1400.9	937.5	8.1	5.9	5.8	5.8
		4	2	7	12	13	1788.1	9.8	1607.4	1070.3	8.3	6.2	6.1	6.1
3	10	2	1	8	9	9	1798.2	13.7	1483.3	1006.2	11.3	8.0	7.7	7.9
		4	0	8	9	8	1800.0	13.7	1775.8	1298.7	11.2	8.1	7.7	7.9
	20	2	2	8	11	9	1796.0	12.1	1510.1	923.8	11.1	7.2	7.0	7.0
		4	0	8	9	9	1800.0	12.1	1755.5	1382.1	11.4	7.8	7.7	7.7
Total/Average			62	84	155	151	1665.8	12.0	1454.4	945.7	8.6	6.1	5.8	5.9

Table 5. Computational results for M1, HDH, WS-MIP, and PS under the precedence graph structure analysis.

			Δ				CPU(s)				Gap(%)			
TD	OS	Pos	M1	HDH	WS-MIP	PS	M1	HDH	WS-MIP	PS	M1	HDH	WS-MIP	PS
BI	0.2	2	13	26	34	33	1602.6	33.7	998.1	783.5	6.5	3.0	3.0	3.0
		4	10	14	30	29	1715.8	33.7	1454.6	813.1	6.5	3.1	3.0	3.0
	0.9	2	0	0	0	1	1800.0	0.5	1800.0	881.8	8.8	6.8	6.7	6.7
		4	0	0	1	2	1800.0	0.5	1786.3	1163.4	8.6	6.9	6.7	6.7
PB	0.2	2	13	24	36	35	1517.6	13.7	906.0	558.5	9.7	4.8	4.8	4.8
		4	10	20	35	31	1593.6	13.7	1419.7	1055.7	9.6	4.9	4.8	4.8
	0.9	2	9	0	11	11	1631.9	0.1	1603.4	1134.6	9.5	9.1	8.5	8.9
		4	7	0	8	9	1664.6	0.1	1667.1	1174.7	9.9	9.8	8.9	9.6
Total/Average			62	84	155	151	1665.8	12.0	1454.4	945.7	8.6	6.0	5.8	5.9

Results are filtered and presented in terms of instances' parameter size and graph structure to perform computational analyses[1] (Section 5.1.1) and extract managerial insights (Section 5.1.2). For each sub-group of instances, results are reported in Tables 4–9 and discussed. In brief, the purpose of these analyses and comparisons is to verify and quantify whether the flexibility of multi-manned stations can decrease resource utilisation and under which conditions it does so.

5.1.1. Size and structure analysis

Table 4 is a summary of the solutions obtained by the monolithic model (M1), the proposed heuristic (HDH), and the model-driven post-optimisation phases (WS-MIP and PS). The results are summarised for all graph structures and order strengths. Thus, these clusters with 40 instances are addressed by their number of heterogeneous workers (Het), percentage of incompatibility (Inc(%)), and the number of workers/positions allowed in the same station (Pos). The number of optimal solutions (Δ) found by either M1, HDH, WS-MIP, or PS is reported. Note that HDH does not provide optimality proofs and PS only guarantee optimality when occasionally (see Section 4.4) therefore, we employ WS-MIP lower bounds when counting Δ for these methods. Computational processing times in seconds (CPU(s)) including all the steps and the average gap calculated as $\frac{UB-LB}{UB}$ (Gap(%)) for each cluster of instances are also presented.

Table 6. Improvements obtained by each MALWIBP approach (M1, HDH, WS-MIP, and PS) compared to the ALWIBP CIH-LS1 approach in terms of number of workers and stations under the parameter size analysis.

		θ_w			θ_s		
Pos	App	B	T	W	B	T	W
2	M1	1	165	74	239	1	0
	HDH	0	240	0	193	47	0
	WS-MIP	2	238	0	233	7	0
	PS	0	240	0	209	31	0
4	M1	2	167	71	239	1	0
	HDH	0	240	0	193	47	0
	WS-MIP	3	237	0	236	4	0
	PS	0	240	0	209	31	0
Total		5	475	0	473	7	0

[1] Solutions for individual instances are made available in the research data.

Table 7. Improvements obtained by each proposed solution method (HDH, WS-MIP, and PS) compared to the monolithic model (M1) in terms of number of workers and stations under the parameter size analysis.

		θ_w			θ_s		
Pos	App	B	T	W	B	T	W
2	HDH	74	165	1	74	45	121
	WS-MIP	75	165	0	144	75	21
	PS	74	165	1	142	54	44
4	HDH	71	167	2	64	37	139
	WS-MIP	72	168	0	146	70	24
	PS	71	167	2	146	49	45
	Total	147	333	0	333	119	28

Table 8. Improvements obtained by each MALWIBP approach (M1, HDH, WS-MIP, and PS) compared to the ALWIBP CIH-LS1 approach in terms of number of workers and stations under the precedence graph structure analysis.

			θ_w			θ_s		
TD	OS	App	B	T	W	B	T	W
BI	0.2	M1	0	65	55	120	0	0
		HDH	0	120	0	120	0	0
		WS-MIP	0	120	0	120	0	0
		PS	0	120	0	120	0	0
	0.9	M1	0	86	34	118	2	0
		HDH	0	120	0	98	22	0
		WS-MIP	0	120	0	111	9	0
		PS	0	120	0	104	16	0
PB	0.2	M1	0	72	48	120	0	0
		HDH	0	120	0	120	0	0
		WS-MIP	0	120	0	120	0	0
		PS	0	120	0	120	0	0
	0.9	M1	3	109	8	120	0	0
		HDH	0	120	0	48	72	0
		WS-MIP	5	115	0	116	4	0
		PS	0	120	0	74	46	0
Total			5	475	0	473	7	0

Table 9. Improvements obtained by each proposed solution method (HDH, WS-MIP, and PS) compared to the monolithic model (M1) in terms of number of workers and stations under the precedence graph structure analysis.

			θ_w			θ_s		
TD	OS	App	B	T	W	B	T	W
BI	0.2	HDH	55	65	0	69	31	20
		WS-MIP	55	65	0	94	18	8
		PS	55	65	0	97	15	8
	0.9	HDH	34	86	0	0	4	116
		WS-MIP	34	86	0	36	55	29
		PS	34	86	0	53	40	27
PB	0.2	HDH	48	72	0	69	43	8
		WS-MIP	48	72	0	98	20	2
		PS	48	72	0	96	22	2
	0.9	HDH	8	109	3	0	4	116
		WS-MIP	10	110	0	62	52	6
		PS	8	109	3	42	26	52
Total			147	333	0	333	119	28

In this subset containing 240 instances for each number of positions (Pos), the monolithic model M1 solved with Gurobi could obtain 35 and 27 optimal solutions for 2 and 4 available positions. Average computational processing time was around 1600 s and the average integrality gap was 8.6%. Furthermore, the proposed heuristic decomposition method proved its effectiveness, as seen in Table 4. The HDH is extremely fast, obtaining feasible solutions in a few seconds (12 s, on average). Our method also improved the number of proven optimal solutions (using the lower bound obtained by M1) from 62 to 84.

By inputting HDH's solution into the M1, our proposed WS-MIP and PS showed synergistic gains compared to M1 and HDH. In combination, they achieved better solutions for 273 of the 480 instances and equivalent solutions for the remainder one, while keeping CPU times less than M1's for all instances in which both methods reached optimality (209.4 s against 764.5 s, on average). Overall, the WS-MIP and the PS respectively achieved optimality in 155 and 151 instances, whereas integrality gaps were reduced to 5.8% and 5.9%, on average. Finally, it is important to point out that the total number of optimal solutions found by either WS-MIP or PS sums up to 166 in the dataset, indicating some complementariness between the methods.

Table 5 follows the same structure of Table 4, with results now summarised in terms of task time distributions (TD), precedence graphs order strengths (OS), and number of positions in each station (Pos), generating clusters with 60 instances each.

Further insights can be extracted by analysing results from a graph structure perspective. For instance, while M1 was able to optimally solve 46 instances with a less restrictive precedence graph (OS = 0.2), it only solved to optimality 16 of them in the more restrictive half (OS = 0.9). In other words, 74.2% of the M1's optimal solutions were obtained in the less restrictive portion of the dataset (OS = 0.2), which was a counter-intuitive result, since instances with more restrictive precedence graphs are generally easier to solve, as hinted by the reported average CPU times in Moreira, Miralles, and Costa (2015). A closer look at these results can give a reason for such finding: instances with a more restrictive precedence graph (OS = 0.9) might be more challenging to solve due to the scheduling portion of the problem, which explains the inverted trend when compared to literature results for the ALWIBP.

The HDH solves high OS instances much faster than their low OS counterparts, but only found optimal solutions for the latter. The explanation here is twofold: (i) since there is not much room for improvement, it is easier for the station minimisation model (presented in Section 4.2.2) to find local optima in instances with stricter precedence graphs and (ii) the lower bound provided by M1 and used to check optimality has a lower

quality for the low OS instances, as discussed in the previous paragraph.

Finally, the usefulness of MILP-based post-optimisation processes is demonstrated for instances with both low and high OS. In summary, the HDH optimally solved fewer instances than its counterparts: a total of 84 optimal solutions against 155 and 151 for WS-MIP and PS, respectively. By performing the Kruskal–Wallis test among these three sample groups, we observed that there is a significant difference among them, with $H = 68.014$ $(2, N = 1440, P < 0.001)$. However, this difference in performance does not appear by only comparing WS-MIP to PS: $H = 2.882$ $(1, N = 960, P > 0.05)$. Naturally, these statistically significant improvements obtained by WS-MIP and PS over the HDH came with higher average CPU times. These Kruskal–Wallis test results were made available in the research data.

5.1.2. Managerial insights

The computational study herein conducted aims to answer specific requirements that normally arise in assembly lines of ordinary companies manufacturing large-size products with a heterogeneous workforce. With these results in hand, we can equip the line managers with practical tools that ease the integration of heterogeneous workers in production processes. As all medium-size instances have also been solved by Moreira, Miralles, and Costa (2015) with the single worker hypothesis, results obtained by M1, HDH, WS-MIP, and PS approaches (App) are compared to those documented in the referred paper. Such analysis is helpful in obtaining managerial insights for assembly lines with heterogeneous workers and multi-manned stations, because it effectively demonstrates that, by the means of the proposed approaches, the additional flexibility indeed brings compelling gains in resource usage without compromising productivity levels. We report the comparison in terms of total number of workers and stations in the line, considering the optimal or the best found primal bound by CIH-LS1. The number of instances in which M1, HDH, WS-MIP, and PS outperformed (B – better), underperformed (W – worse), or tied with (T – tie) CIH-LS1 in terms of total number of workers (θ_w) and stations (θ_s) are described in Table 6. Note that it is possible for a solution of a given instance to simultaneously perform better in terms of stations and worse in terms of workers. In such cases, the result is considered worse overall due to the hierarchy considered in the objective function (1). The Total row sums up the results for the best incumbent found by all methods combined. Similarly, instances in which HDH, WS-MIP and PS outperformed, underperformed, or tied with M1 are also counted in Table 7 in terms of θ_w and θ_s.

According to Table 6, M1 has reduced the line length while maintaining or reducing the number of workers in the line over several cases. This improvement has been detected in 165 and 168 instances compared to CIH-LS1, when allowing 2 or 4 workers per station. Nonetheless, many instances (145 out of 480) ended up resulting in worse solutions for the given time limit, especially when the number of heterogeneous workers is increased. Therefore, in terms of number of workers, the M1 has reached solutions that are as good as the ones found by CIH-LS1 for the majority of instances whilst reducing the line length.

The HDH, WS-MIP and PS are compared to best-found solutions by Moreira, Miralles, and Costa (2015) in Table 6 and directly to M1 in Table 7. By allowing multiple workers per station, it can be observed that the HDH obtained improved solutions in 80.4% of instances (386 out of 480) when compared to CIH-LS1, out of which 212 were also better than M1. Meanwhile, WS-MIP and PS further improved the results by obtaining better primal bounds in 357 and 351 instances compared to M1, or 390 in total. Note that for all cases in which WS-MIP or PS yielded a solution with more stations than M1, it necessarily used fewer workers. Moreover, they also obtained improved primal bounds in 98.5% of instances (473 out of 480) compared to CIH-LS1.

Table 7 also demonstrates synergistic gains in combining M1 and HDH, since the aggregation of WS-MIP and PS results outperformed the best ones between the model and the heuristic. In fact, our proposed methods managed to find 272 exclusively best solutions compared to the proposed model and heuristic applied separately. Furthermore, a direct comparison between the methods showed that the WS-MIP outperformed PS in 77 instances, whereas PS outperformed WS-MIP in 85 of them, and they tied in the remaining 318 results.

In Table 8, results reported by Moreira, Miralles, and Costa (2015) are compared to those obtained by M1, HDH, WS-MIP, and PS approaches (App) according to precedence graph structures. This comparison is also conducted within the multi-manned paradigm, i.e. in terms of θ_w and θ_s in the line. We consider the optimal or the best found primal bound by CIH-LS1 and the number of instances in which the proposed methods outperform, underperform, or tie with Moreira, Miralles, and Costa (2015) is reported. Note that Moreira, Miralles, and Costa (2015) proposed the ALWIBP, so the parallel workplace flexibility was not examined by them, meaning any given solution provided by CIH-LS1 has the same number of workers as stations. Thus, the goal here is not to show that our approaches are better than Moreira, Miralles, and Costa (2015)'s, but to evidence that they

are capable of finding improved solutions once the proposed flexibility is embedded into the methods. Likewise, instances in which HDH, WS-MIP, and PS were better, worse, or equal to M1 are enumerated in terms of θ_w and θ_s in Table 9.

Table 8 brings another interesting insight: although M1 had proven fewer solutions for high OS instances, it reduced the line length while maintaining the number of workers in more cases than for low OS instances, when compared to CIH-LS1. Whilst M1 only obtained 137 improved solutions for low OS instances, this number was much higher for high OS ones, i.e. 196 solutions.

The HDH, WS-MIP, and PS comparisons are made against the best solutions found by Moreira, Miralles, and Costa (2015) in Table 8 and to M1's results in Table 9. For low and high OS instances, it was observed that the HDH improved CIH-LS1 solutions in 240 and 146 instances, respectively. This effect is precisely the opposite of what had been observed for M1 and can be explained by the fact that worker-task assignments are fixed in the HDH, which shrinks the possible room for improvement. Complementary, WS-MIP and PS further improved the results by combining the strengths of M1 and HDH: out of 480 instances, they obtained improved solutions in 240 (low OS) and 233 (high OS) instances when compared to ALWIBP results (Moreira, Miralles, and Costa 2015). Note that those improvements are much more noticeable in low OS instances for the exact same reasons, i.e. more opportunities to assign parallel tasks to workers due to fewer precedence relations.

Table 9 also depicts the previously mentioned synergistic gains in combining M1 and HDH from a different angle. Out of the 390 instances in which WS-MIP or PS performed better than M1 in terms of solution quality, 220 are low OS instances by the virtue of HDH's incumbent solution being prompted as a starting point to the proposed procedures, meaning that HDH was capable of finding the optimal solution with the initial CIH-LS1 worker-task assignment in low OS instances.

For a production manager with the primary aim of integrating heterogeneous workers into the assembly line while maintaining a given productivity rate, this computational study brought relevant insights. From a trade-off perspective, managers may face a scenario where they either (i) keep single-manned stations to avoid task scheduling problems or (ii) consume fewer resources in terms of homogeneous workers and line length. According to the herein obtained results, there are settings in which deciding for (ii) might be more profitable, including, but not limited to, manufacturing layouts designed for large-size products to support more workers in the same station, precedence diagrams with low ordering strengths to allow for parallel task execution, and a high degree of heterogeneity, both in terms of number of heterogeneous workers and task processing variability. Hence, the proposed methods work as a decision-making tool to aid assembly line managers, so they can follow and benefit from this alternative path (ii) to aggregate workers into multi-manned stations equipped with a numerical analysis to support their decision on switching configurations. Such an approach also helps avoid technical downsides related to task scheduling coordination risks, alleviating potential idle times caused by parallel work.

Lastly, we were able to reduce the workforce in five instances compared to its single-manned line counterpart. We also confirmed the solutions' feasibility and made them available in the research data. This outcome proves that both theoretical advantages of using multi-manned stations presented in Section 3 are also achievable for benchmark instances, which exceeds the expected results obtained in previous MALBP-related studies (Kellegöz 2017; Michels et al. 2019). Hence, these results underline an important contribution of the proposed HDH integrated with its warm start and proximity search procedures (Section 4).

5.2. Large-size instances

The proposed monolithic model (M1) struggled to solve most medium-size instances in the evaluated dataset and was outperformed by the proposed heuristic (HDH). Thus, only the HDH was applied to the 3200 large-size instances (100 tasks) with a time limit set to 60 s in this section. Since the HDH gets the total number of workers as a fixed parameter from the CIH-LS1, only line length improvements are possible. Therefore, results are compared to the CIH-LS1 presented by Moreira, Miralles, and Costa (2015) in terms of line length reduction in Table 10. Absolute values represent the number of instances in which improvements were detected (θ_s) and relative values stand for the average line length reduction in such cases. Each cluster of instances contains average values of 25 similar cases, given by the information stated in the rows and columns. These are addressed by their number of heterogeneous workers (Het), task processing times variability (Var) distributions, percentage of incompatibility (Inc(%)), number of workers/positions allowed in the same station (Pos), and precedence graph structure, i.e. time distribution (BI, PB) and order strength (OS = 0.2, OS = 0.9).

The HDH is compared to the CIH-LS1 in this subset containing 1600 instances for each number of positions (2 and 4). With the additional multi-manned flexibility, the former obtained 3084 solution improvements out of 3200 instances over the latter in terms of total number of used stations. For the improved cases, lines were on

Table 10. Results comparison for large-size instances between heuristics (HDH and CIH-LS1) in terms of line length.

				θ_s			
				BI		PB	
Het	Var	Inc(%)	Pos	0.2	0.9	0.2	0.9
1	U[t,2t]	10	2	25 (45.4%)	24 (14.3%)	25 (44.9%)	19 (6.8%)
			4	25 (61.6%)	24 (15.0%)	25 (61.7%)	19 (7.3%)
		20	2	25 (45.0%)	25 (15.9%)	25 (44.7%)	20 (8.2%)
			4	25 (62.0%)	25 (16.5%)	25 (60.1%)	20 (8.5%)
	U[t,5t]	10	2	25 (46.7%)	24 (14.9%)	25 (44.9%)	20 (8.5%)
			4	25 (63.2%)	24 (15.4%)	25 (62.2%)	20 (8.8%)
		20	2	25 (46.4%)	25 (13.8%)	25 (45.2%)	21 (8.4%)
			4	25 (63.6%)	25 (14.7%)	25 (60.7%)	21 (9.0%)
2	U[t,2t]	10	2	25 (46.0%)	24 (13.9%)	25 (44.8%)	24 (7.7%)
			4	25 (62.6%)	24 (14.9%)	25 (61.8%)	24 (7.7%)
		20	2	25 (45.8%)	24 (14.4%)	25 (44.3%)	22 (8.7%)
			4	25 (62.0%)	24 (14.9%)	25 (61.9%)	22 (9.0%)
	U[t,5t]	10	2	25 (46.3%)	25 (16.6%)	25 (44.7%)	20 (7.6%)
			4	25 (63.6%)	25 (18.1%)	25 (61.3%)	20 (7.6%)
		20	2	25 (46.4%)	25 (15.7%)	25 (46.1%)	22 (7.7%)
			4	25 (63.1%)	25 (16.5%)	25 (61.9%)	22 (7.7%)
3	U[t,2t]	10	2	25 (46.0%)	24 (16.0%)	25 (45.3%)	23 (8.4%)
			4	25 (63.0%)	24 (16.6%)	25 (62.6%)	23 (8.7%)
		20	2	25 (47.0%)	25 (15.0%)	25 (45.6%)	21 (7.4%)
			4	25 (62.2%)	25 (16.1%)	25 (61.0%)	21 (7.4%)
	U[t,5t]	10	2	25 (46.4%)	25 (14.8%)	25 (45.3%)	23 (8.7%)
			4	25 (63.1%)	25 (15.9%)	25 (59.5%)	23 (9.5%)
		20	2	25 (45.9%)	25 (14.5%)	25 (44.9%)	22 (8.5%)
			4	25 (62.1%)	25 (15.7%)	25 (63.5%)	22 (9.0%)
4	U[t,2t]	10	2	25 (46.4%)	25 (16.3%)	25 (44.3%)	20 (7.9%)
			4	25 (61.8%)	25 (16.8%)	25 (61.6%)	20 (8.4%)
		20	2	25 (46.0%)	25 (17.0%)	25 (44.9%)	24 (9.2%)
			4	25 (62.1%)	25 (18.1%)	25 (61.2%)	24 (9.4%)
	U[t,5t]	10	2	25 (45.7%)	24 (16.0%)	25 (45.3%)	22 (8.4%)
			4	25 (61.4%)	24 (17.1%)	25 (60.7%)	22 (8.4%)
		20	2	25 (46.3%)	25 (17.0%)	25 (43.9%)	25 (10.4%)
			4	25 (62.7%)	25 (18.4%)	25 (60.4%)	25 (10.6%)
	Total (Average)			800 (54.3%)	788 (15.8%)	800 (53.2%)	696 (8.4%)

average 28.7% and 37.2% shorter for 2 and 4 station positions, respectively. In other words, HDH has reduced the line length while maintaining the number of workers in the line in 96.3% of instances when allowing multiple workers per station.

We also applied the Kruskal–Wallis test to HDH and CIH-LS1 for large instances. As a result, it can be observed that there is a significant difference between these two sample groups, with $H = 2099,634$ $(1, N = 6400, P < 0.001)$. Thus, these reported improvements are statistically significant. The Kruskal–Wallis test result for large instances comparing HDH to CIH-LS1 was made available in the research data.

No clear correlation could be observed between the number of improvements and the number of heterogeneous workers (Het), task processing times variability (Var), percentage of incompatibility (Inc(%)), or number of available positions (Pos). Conversely, improvements are less likely when graphs have a high OS. It can be observed that, independently from time distributions, it is much easier to find solutions with shorter lines when graph structures are not too restrictive: we found 1600 and 1484 improvements out of 1600 instances for order strengths of 0.2 and 0.9, respectively. The reason behind this outcome might be twofold: one is CIH-LS1's remarkable capability of finding optimal or near-optimal solutions for high OS instances (Moreira, Miralles, and Costa 2015), and the other is that worker-task assignments are fixed for the HDH, which does not leave much room for improvement since the precedence graph is quite restrictive.

As expected, more significant line length reductions were detected in BI instances over PB ones. Once again, this can be explained by the fact that CIH-LS1 is more likely to perform better on PB graphs, which yields less margin for straightforward improvements. Similarly, as the number of heterogeneous workers (Het) increases, slightly higher line length reductions were possible.

Finally, the average computational processing time to solve the large-size dataset was 47.3 s for the CIH-LS1, whereas HDH took 48.0 and 48.5 s with 2 and 4 available positions, respectively, which is only a marginal increment to add the parallel workplace flexibility into the picture and obtain statistically significant better solutions. This further highlights the importance of the HDH presented in Section 4: we can quickly obtain good primal bounds for large-size instances considering multi-manned stations with reduced computational burden.

6. Conclusions

We have considered a general assembly line balancing problem with two particularities: stations can be multi-manned and the workforce is heterogeneous. Both of these characteristics have been studied in the literature, but this was the first time they were addressed together, giving rise to a highly combinatorial problem that involves task and worker assignment decisions coupled with task scheduling problems within each multi-manned station.

We have formalised the problem and proposed a mixed-integer linear programming model that is enriched with symmetry-breaking valid inequalities to solve instances with up to 50 tasks. Problems of larger size have been solved by a heuristic decomposition approach, which has also been proposed in this paper. The solutions obtained by this heuristic approach are improved by two post-optimisation strategies based on a warm-start of the original model and a large-neighbourhood search based on the idea of proximity search.

Computational experiments on a large dataset of instances show that the proposed methods are able to successfully solve the problem. In particular, the heuristic decomposition can find solutions for large instances in a few seconds. The obtained solutions allow for a discussion of the proposed modelling framework implications

on a managerial scale. Indeed, our results demonstrate that considering multi-manned stations can often reduce the number of stations needed in the assembly line, even in the presence of a highly heterogeneous workforce. Additionally, the proposed warm start and proximity search procedures might also be applied to challenging MALBP instances that remain unsolved.

As the proposed heuristic hierarchically decomposes the problem, optimality proofs cannot be guaranteed. Further research may investigate holistic strategies that take all decision levels into account simultaneously or iteratively. Lastly, incorporating more practical extensions into the problem is a desirable modification, such as line layout (U-line, parallel stations), product variety (multi and mixed model lines), assignment restrictions, continuous pace control with flexible stations, or salaries for a skilled cohort. The complementary variant of the problem (type-II) can also be explored by modifying the model to maximise the production rate (minimising cycle time) for a given set of resources (fixed workforce).

Acknowledgments

The authors would like to thank Professor Mayron César de Oliveira Moreira for providing us with the codes used in Moreira, Miralles, and Costa (2015) and four anonymous referees for constructive criticism, which has helped improve this manuscript.

Disclosure statement

No potential conflict of interest was reported by the author(s).

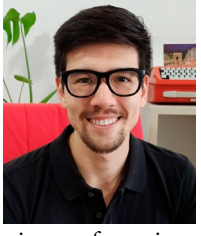
Data Availability Statement

The data supporting the findings of this study are available within the article and in its supplementary materials, which are openly available in Mendeley Data.

Funding

This research was partially funded by the Australian Government through the Australian Research Council Industrial Transformation Training Centre in Optimisation Technologies, Integrated Methodologies, and Applications (OPTIMA), Project ID IC200100009.

Notes on Contributors



Adalberto Sato Michels received his PhD in 2024 from the School of Mathematics and Statistics – University of Melbourne - Australia. He also holds a master's degree from the Graduate Program in Electrical and Computer Engineering (CPGEI) at UTFPR – Brazil. His research involves applications of Operational Research techniques, focussing on Mixed-Integer Linear Programming, heuristics, and decomposition methods. His current projects include extensions of the Assembly Line Balancing Problem with an economic and social perspective, namely multi-manned stations and workforce heterogeneity.



Alysson M. Costa is an Associate Professor in Operations Research at the School of Mathematics and Statistics – University of Melbourne. Throughout his career, he has worked extensively with mixed integer programming (modelling and solution methods) applied to problems in different areas such as environmental water management, disaster relief operations, educational timetabling, crop rotation, assembly line balancing and city logistics, among others. He received his PhD in 2006 from HEC Montreal / University of Montreal – Canada. His thesis, titled "Models and algorithms for two network design problems", received the Cecil Graham doctoral dissertation award from the Canadian Applied and Industrial Mathematics Society. Before that, he received MSc. and B.Eng. Degrees in Electrical Engineering from the State University of Campinas – Brazil.

ORCID

Adalberto Sato Michels http://orcid.org/0000-0003-2635-9700

References

Şahin, Murat, and Talip Kellegöz. 2022. "Benders' Decomposition Based Exact Solution Method for Multi-Manned Assembly Line Balancing Problem with Walking Workers." *Annals of Operations Research* 321 (1): 507–540. https://doi.org/10.1007/s10479-022-05118-z.

Ahmed, Mohamed Ben, Maryia Hryhoryeva, Lars Magnus Hvattum, and Mohamed Haouari. 2022. "A Matheuristic for the Robust Integrated Airline Fleet Assignment, Aircraft Routing, and Crew Pairing Problem." *Computers & Operations Research* 137: 105551. https://doi.org/10.1016/j.cor.2021.105551.

Akagi, Fumio, Hirokazu Osaki, and Susumu Kikuchi. 1983. "A Method for Assembly Line Balancing with More Than One Worker in Each Station." *International Journal of Production Research* 21 (5): 755–770. https://doi.org/10.1080/00207548308942409.

Álvarez Miranda, Eduardo, and Jordi Pereira. 2019. "On the Complexity of Assembly Line Balancing Problems." *Computers & Operations Research* 108: 182–186.

Álvarez Miranda, Eduardo, Jordi Pereira, and Mariona Vilà. 2023. "Analysis of the Simple Assembly Line Balancing Problem Complexity." *Computers & Operations Research* 159: 106323. https://doi.org/10.1016/j.cor.2023.106323.

Andreu-Casas, Enric, Alberto García-Villoria, and Rafael Pastor. October, 2022. "Multi-Manned Assembly Line Balancing Problem with Dependent Task Times: A Heuristic Based on Solving a Partition Problem with Constraints." *European Journal of Operational Research* 302 (1): 96–116. https://doi.org/10.1016/j.ejor.2021.12.002.

Araújo, Felipe F. B., Alysson M. Costa, and Cristóbal Miralles. 2012. "Two Extensions for the Alwabp: Parallel Stations and Collaborative Approach." *International Journal of Production Economics* 140 (1): 483–495. https://doi.org/10.1016/j.ijpe.2012.06.032.

Araújo, Felipe F. B., Alysson M. Costa, and Cristóbal Miralles. 2015. "Balancing Parallel Assembly Lines with Disabled Workers." *European Journal of Industrial Engineering* 9 (3): 344–365. https://doi.org/10.1504/EJIE.2015.069343.

Bartholdi, J. J. 1993. "Balancing Two-Sided Assembly Lines: A Case Study." *International Journal of Production Research* 31 (10): 2447–2461. https://doi.org/10.1080/00207549308956868.

Battaïa, Olga, and Alexandre Dolgui. 2013. "A Taxonomy of Line Balancing Problems and Their Solution Approaches." *International Journal of Production Economics* 142 (2): 259–277. https://doi.org/10.1016/j.ijpe.2012.10.020.

Battaïa, Olga, and Alexandre Dolgui. November, 2022. "Hybridizations in Line Balancing Problems: A Comprehensive Review on New Trends and Formulations." *International Journal of Production Economics* 250: 108673. https://doi.org/10.1016/j.ijpe.2022.108673.

Baybars, Ilker August, 1986. "A Survey of Exact Algorithms for the Simple Assembly Line Balancing Problem." *Management Science* 32 (8): 909–932. https://doi.org/10.1287/mnsc.32.8.909.

Becker, Christian, and Armin Scholl. 2006. "A Survey on Problems and Methods in Generalized Assembly Line Balancing." *European Journal of Operational Research* 168 (3): 694–715. https://doi.org/10.1016/j.ejor.2004.07.023.

Becker, Christian, and Armin Scholl. 2009. "Balancing Assembly Lines with Variable Parallel Workplaces: Problem Definition and Effective Solution Procedure." *European Journal of Operational Research* 199 (2): 359–374. https://doi.org/10.1016/j.ejor.2008.11.051.

Benders, J. F. 1962. "Partitioning Procedures for Solving Mixed-Variables Programming Problems." *Numerische Mathematik* 4 (1): 238–252. https://doi.org/10.1007/BF01386316.

Boland, Natashia, Matteo Fischetti, Michele Monaci, and Martin Savelsbergh. 2016. "Proximity Benders: A Decomposition Heuristic for Stochastic Programs." *Journal of Heuristics* 22 (2): 181–198. https://doi.org/10.1007/s10732-015-9306-1.

Borba, Leonardo, and Marcus Ritt. 2014. "A Heuristic and a Branch-And-bound Algorithm for the Assembly Line Worker Assignment and Balancing Problem." *Computers & Operations Research* 45: 87–96. https://doi.org/10.1016/j.cor.2013.12.002.

Borba, Leonardo, Marcus Ritt, and Cristóbal Miralles. 2018. "Exact and Heuristic Methods for Solving the Robotic Assembly Line Balancing Problem." *European Journal of Operational Research* 270 (1): 146–156. https://doi.org/10.1016/j.ejor.2018.03.011.

Boysen, Nils, Malte Fliedner, and Armin Scholl. 2007. "A Classification of Assembly Line Balancing Problems." *European Journal of Operational Research* 183 (2): 674–693. https://doi.org/10.1016/j.ejor.2006.10.010.

Boysen, Nils, Malte Fliedner, and Armin Scholl. 2008. "Assembly Line Balancing: Which Model to Use when?." *International Journal of Production Economics* 111 (2): 509–528. https://doi.org/10.1016/j.ijpe.2007.02.026.

Boysen, Nils, Philipp Schulze, and Armin Scholl. September, 2022. "Assembly Line Balancing: What Happened in the Last Fifteen Years?." *European Journal of Operational Research* 301 (3): 797–814. https://doi.org/10.1016/j.ejor.2021.11.043.

Campana, Nícolas P., Manuel Iori, and Mayron César O. Moreira. 2022. "Mathematical Models and Heuristic Methods for the Assembly Line Balancing Problem with Hierarchical Worker Assignment." *International Journal of Production Research* 60 (7): 2193–2211. https://doi.org/10.1080/00207543.2021.1884767.

Cantos Lopes, Thiago, Adalberto Sato Michels, and Leandro Magatão. 2019. "A Note to: A Hybrid Algorithm for Allocating Tasks, Operators, and Workstations in Multi-Manned Assembly Lines." *Journal of Manufacturing Systems* 52: 205–208. https://doi.org/10.1016/j.jmsy.2019.06.003.

Cantos Lopes, Thiago, Giuliano Vidal Pastre, Adalberto Sato Michels, and Leandro Magatão. 2019. "Flexible Multi-Manned Assembly Line Balancing Problem: Model, Heuristic Procedure, and Lower Bounds for Line Length Minimization." *Omega (United Kingdom)* 95: 102063. https://doi.org/10.1016/j.omega.2019.04.006.

Codato, Gianni, and Matteo Fischetti. 2006. "Combinatorial Benders' Cuts for Mixed-Integer Linear Programming." *Operations Research* 54 (4): 756–766. https://doi.org/10.1287/opre.1060.0286.

Corominas, Albert, Rafael Pastor, and Joan Plans. 2008. "Balancing Assembly Line with Skilled and Unskilled Workers." *Omega* 36 (6): 1126–1132. https://doi.org/10.1016/j.omega.2006.03.003.

Cortez, Pâmela M. C., and Alysson M. Costa. 2015. "Sequencing Mixed-Model Assembly Lines Operating with a Heterogeneous Workforce." *International Journal of Production Research* 53 (11): 3419–3432. https://doi.org/10.1080/00207543.2014.987881.

Costa, Alysson M., and Cristóbal Miralles. 2009. "Job Rotation in Assembly Lines Employing Disabled Workers." *International Journal of Production Economics* 120 (2): 625–632. https://doi.org/10.1016/j.ijpe.2009.04.013.

Dimitriadis, Sotirios G. 2006. "Assembly Line Balancing and Group Working: A Heuristic Procedure for Workers' Groups Operating on the Same Product and Workstation." *Computers & Operations Research* 33 (9): 2757–2774. https://doi.org/10.1016/j.cor.2005.02.027.

Eghtesadifard, Mahmood, Mojtaba Khalifeh, and Morteza Khorram. 2020. "A Systematic Review of Research Themes and Hot Topics in Assembly Line Balancing Through the Web of Science Within 1990–2017." *Computers & Industrial Engineering* 139: 106182. https://doi.org/10.1016/j.cie.2019.106182.

Fattahi, Parviz, and Abdolreza Roshani. 2011. "A Mathematical Model and Ant Colony Algorithm for Multi-Manned Assembly Line Balancing Problem." *International Journal of Advanced Manufacturing Technology* 53 (1-4): 363–378. https://doi.org/10.1007/s00170-010-2832-y.

Fischetti, Matteo, and Michele Monaci. 2014. "Proximity Search for 0-1 Mixed-Integer Convex Programming." *Journal of Heuristics* 20 (6): 709–731. https://doi.org/10.1007/s10732-014-9266-x.

Fischetti, Martina, and Michele Monaci. 2016. "Proximity Search Heuristics for Wind Farm Optimal Layout." *Journal of Heuristics* 22 (4): 459–474. https://doi.org/10.1007/s10732-015-9283-4.

Giglio, Davide, Massimo Paolucci, Abdolreza Roshani, and Flavio Tonelli. July, 2017. "Multi-Manned Assembly Line Balancing Problem with Skilled Workers: A New Mathematical Formulation." *IFAC-PapersOnLine* 50 (1): 1211–1216. https://doi.org/10.1016/j.ifacol.2017.08.344.

Güner, Funda, Abdül K. Görür, Benhür Satır, Levent Kandiller, and John H. Drake. 2023. "A Constraint Programming Approach to a Real-World Workforce Scheduling Problem for Multi-Manned Assembly Lines with Sequence-Dependent Setup Times." *International Journal of Production Research* 60 (9): 1211–1216. https://doi.org/10.1080/00207543.2023.2226772.

Gurobi Optimization. 2019. Gurobi Optimizer reference manual.

Hashemi-Petroodi, S. Ehsan, Simon Thevenin, Sergey Kovalev, and Alexandre Dolgui. December, 2022. "Model-Dependent Task Assignment in Multi-Manned Mixed-Model Assembly Lines with Walking Workers." *Omega* 113: 102688. https://doi.org/10.1016/j.omega.2022.102688.

Hashemi-Petroodi, S. Ehsan, Simon Thevenin, Sergey Kovalev, and Alexandre Dolgui. January, 2023. "Markov Decision Process for Multi-Manned Mixed-Model Assembly Lines with Walking Workers." *International Journal of Production Economics* 255: 108661. https://doi.org/10.1016/j.ijpe.2022.108661.

Hoffmann, T. R. 1963. "Assembly Line Balancing with a Precedence Matrix." *Management Science* 9 (4): 551–562. https://doi.org/10.1287/mnsc.9.4.551.

Huo, Jiage, and Carman K. M. Lee. 2023. "A Benders' Decomposition Algorithm for Balancing and Sequencing of the Mixed-Model Multi-Manned Assembly Lines." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 54 (4): 2114–2124. https://doi.org/10.1109/TSMC.2023.3335998.

Katiraee, Niloofar, Martina Calzavara, Serena Finco, Daria Battini, and Olga Battaïa. 2021. "Consideration of Workers' Differences in Production Systems Modelling and Design: State of the Art and Directions for Future Research." *International Journal of Production Research* 59 (11): 3237–3268. https://doi.org/10.1080/00207543.2021.1884766.

Kazemi, Abolfazl, and Abdolhossein Sedighi. 2013. "A Cost-Oriented Model for Balancing Mixed-Model Assembly Lines with Multi-Manned Workstations." *International Journal of Services and Operations Management* 16 (3): 289. https://doi.org/10.1504/IJSOM.2013.056765.

Kellegöz, Talip 2017. "Assembly Line Balancing Problems with Multi-Manned Stations: A New Mathematical Formulation and Gantt Based Heuristic Method." *Annals of Operations Research* 253 (1): 377–404. https://doi.org/10.1007/s10479-016-2156-x.

Kellegöz, Talip, and Bilal Toklu. 2012. "An Efficient Branch and Bound Algorithm for Assembly Line Balancing Problems with Parallel Multi-Manned Workstations." *Computers & Operations Research* 39 (12): 3344–3360. https://doi.org/10.1016/j.cor.2012.04.019.

Kellegöz, Talip, and Bilal Toklu. 2015. "A Priority Rule-Based Constructive Heuristic and An Improvement Method for Balancing Assembly Lines with Parallel Multi-Manned Workstations." *International Journal of Production Research* 53 (3): 736–756. https://doi.org/10.1080/00207543.2014.920548.

Liu, Ming, Zhongzheng Liu, Feng Chu, Rongfan Liu, Feifeng Zheng, and Chengbin Chu. 2021. "Risk-Averse Assembly Line Worker Assignment and Balancing Problem with Limited Temporary Workers and Moving Workers." *International Journal of Production Research* 60 (23): 7074–7092. https://doi.org/10.1080/00207543.2021.2002960.

Michels, Adalberto Sato. 2020. "Industrial Assembly Lines with Multi-operated Workstations: Applications and Methods." https://repositorio.utfpr.edu.br/jspui/handle/1/24499

Michels, Adalberto Sato, and Alysson M. Costa. 2022. "Conserving Workforce While Temporarily Rebalancing Assembly Lines Under Demand Disruption." *International Journal of Production Research* 60 (21): 6616–6636. https://doi.org/10.1080/00207543.2021.1998694.

Michels, Adalberto Sato, Thiago Cantos Lopes, and Leandro Magatão. 2020. "An Exact Method with Decomposition Techniques and Combinatorial Benders' Cuts for the Type-2 Multi-Manned Assembly Line Balancing Problem." *Operations Research Perspectives* 7: 100163. https://doi.org/10.1016/j.orp.2020.100163.

Michels, Adalberto Sato, Thiago Cantos Lopes, Celso Gustavo Stall Sikora, and Leandro Magatão. 2018. "The Robotic Assembly Line Design (rald) Problem: Model and Case Studies with Practical Extensions." *Computers & Industrial Engineering* 120: 320–333. https://doi.org/10.1016/j.cie.2018.04.010.

Michels, Adalberto Sato, Thiago Cantos Lopes, Celso Gustavo Stall Sikora, and Leandro Magatão. 2019. "A Benders' Decomposition Algorithm with Combinatorial Cuts for the Multi-Manned Assembly Line Balancing Problem." *European Journal of Operational Research* 278 (3): 796–808. https://doi.org/10.1016/j.ejor.2019.05.001.

Miralles, Cristóbal, José P. García-Sabater, Carlos Andrés, and Manuel Cardós. February, 2008. "Branch and Bound Procedures for Solving the Assembly Line Worker Assignment and Balancing Problem: Application to Sheltered Work Centres for Disabled." *Discrete Applied Mathematics* 156 (3): 352–367. https://doi.org/10.1016/j.dam.2005.12.012.

Moon, Ilkyeong, Rasaratnam Logendran, and Jeonghun Lee. 2009. "Integrated Assembly Line Balancing with Resource Restrictions." *International Journal of Production Research* 47 (19): 5525–5541. https://doi.org/10.1080/00207540802089876.

Moreira, Mayron César O., Jean François Cordeau, Alysson M. Costa, and Gilbert Laporte. August, 2015. "Robust Assembly Line Balancing with Heterogeneous Workers." *Computers and Industrial Engineering* 88: 254–263. https://doi.org/10.1016/j.cie.2015.07.004.

Moreira, Mayron César O., Cristóbal Miralles, and Alysson M. Costa. February, 2015. "Model and Heuristics for the Assembly Line Worker Integration and Balancing Problem." *Computers & Operations Research* 54: 64–73. https://doi.org/10.1016/j.cor.2014.08.021.

Moreira, Mayron César O., Rafael Pastor, Alysson M. Costa, and Cristóbal Miralles. 2017. "The Multi-Objective Assembly Line Worker Integration and Balancing Problem of Type-2." *Computers & Operations Research* 82: 114–125. https://doi.org/10.1016/j.cor.2017.01.003.

Moreira, Mayron César O., Marcus Ritt, Alysson M. Costa, and Antonio A. Chaves. 2012. "Simple Heuristics for the Assembly Line Worker Assignment and Balancing Problem." *Journal of Heuristics* 18 (3): 505–524. https://doi.org/10.1007/s10732-012-9195-5.

Morrison, David R., Edward C. Sewell, and Sheldon H. Jacobson. July, 2014. "An Application of the Branch, Bound, and Remember Algorithm to a New Simple Assembly Line Balancing Dataset." *European Journal of Operational*

Research 236 (2): 403–409. https://doi.org/10.1016/j.ejor.2013.11.033.

Mutlu, Özcan, Olcay Polat, and Aliye Ayca Supciller. 2013. "An Iterative Genetic Algorithm for the Assembly Line Worker Assignment and Balancing Problem of Type-II." *Computers & Operations Research* 40 (1): 418–426. https://doi.org/10.1016/j.cor.2012.07.010.

Naderi, Bahman, Ahmed Azab, and Katayoun Borooshan. 2019. "A Realistic Multi-Manned Five-Sided Mixed-Model Assembly Line Balancing and Scheduling Problem with Moving Workers and Limited Workspace." *International Journal of Production Research* 57 (3): 643–661. https://doi.org/10.1080/00207543.2018.1476786.

Oksuz, Mehmet Kursat, Kadir Buyukozkan, and Sule Itir Satoglu. 2017. "U-shaped Assembly Line Worker Assignment and Balancing Problem: A Mathematical Model and Two Meta-Heuristics." *Computers & Industrial Engineering* 112: 246–263. https://doi.org/10.1016/j.cie.2017.08.030.

Otto, Alena, Christian Otto, and Armin Scholl. 2013. "Systematic Data Generation and Test Design for Solution Algorithms on the Example of Salbpgen for Assembly Line Balancing." *European Journal of Operational Research* 228 (1): 33–45. https://doi.org/10.1016/j.ejor.2012.12.029.

Pape, T. 2015. "Heuristics and Lower Bounds for the Simple Assembly Line Balancing Problem Type 1: Overview, Computational Tests and Improvements." *European Journal of Operational Research* 240 (1): 32–42. https://doi.org/10.1016/j.ejor.2014.06.023.

Patterson, James H., and Joseph J. Albracht. 1975. "Assembly-Line Balancing: Zero-One Programming with Fibonacci Search." *Operations Research* 23 (1): 166–172. https://doi.org/10.1287/opre.23.1.166.

Pereira, Jordi May, 2018. "The Robust (minmax Regret) Assembly Line Worker Assignment and Balancing Problem." *Computers and Operations Research* 93: 27–40. https://doi.org/10.1016/j.cor.2018.01.009.

Pereira, Jordi, and Marcus Ritt. 2023. "Exact and Heuristic Methods for a Workload Allocation Problem with Chain Precedence Constraints." *European Journal of Operational Research* 309 (1): 387–398. https://doi.org/10.1016/j.ejor.2022.12.035.

Polat, Olcay, Can B. Kalayci, Özcan Mutlu, and Surendra M. Gupta. 2016. "A Two-Phase Variable Neighbourhood Search Algorithm for Assembly Line Worker Assignment and Balancing Problem Type-II: An Industrial Case Study." *International Journal of Production Research* 54 (3): 722–741. https://doi.org/10.1080/00207543.2015.1055344.

Possan Junior, Moacyr Carlos, Adalberto Sato Michels, and Leandro Magatão. 2024. "An Exact Constraint Programming Based Procedure for the Multi-Manned Assembly Line Balancing Problem." *Computers & Operations Research* 162: 106451. https://doi.org/10.1016/j.cor.2023.106451.

Ramezanian, Reza, and Abdullah Ezzatpanah. 2015. "Modeling and Solving Multi-Objective Mixed-Model Assembly Line Balancing and Worker Assignment Problem." *Computers & Industrial Engineering* 87: 74–80. https://doi.org/10.1016/j.cie.2015.04.017.

Ritt, Marcus, Alysson M. Costa, and Cristóbal Miralles. February, 2016. "The Assembly Line Worker Assignment and Balancing Problem with Stochastic Worker Availability." *International Journal of Production Research* 54 (3): 907–922. https://doi.org/10.1080/00207543.2015.1108534.

Rodrigues, Filipe, Agostinho Agra, Lars Magnus Hvattum, and Cristina Requejo. 2021. "Weighted Proximity Search." *Journal of Heuristics* 27 (3): 459–496. https://doi.org/10.1007/s10732-021-09466-0.

Roshani, Abdolreza, and Davide Giglio. 2017. "Simulated Annealing Algorithms for the Multi-Manned Assembly Line Balancing Problem: Minimising Cycle Time." *International Journal of Production Research* 55 (10): 2731–2751. https://doi.org/10.1080/00207543.2016.1181286.

Roshani, Abdolreza, and Farnaz Ghazi Nezami. 2017. "Mixed-Model Multi-Manned Assembly Line Balancing Problem: A Mathematical Model and a Simulated Annealing Approach." *Assembly Automation* 37 (1): 34–50. https://doi.org/10.1108/AA-02-2016-016.

Roshani, Abdolreza, Arezoo Roshani, Abdolhassan Roshani, Mohsen Salehi, and Azadeh Esfandyari. 2013. "A Simulated Annealing Algorithm for Multi-Manned Assembly Line Balancing Problem." *Journal of Manufacturing Systems* 32 (1): 238–247. https://doi.org/10.1016/j.jmsy.2012.11.003.

Sahin, Murat, and Talip Kellegöz. 2019a. "A New Mixed-Integer Linear Programming Formulation and Particle Swarm Optimization Based Hybrid Heuristic for the Problem of Resource Investment and Balancing of the Assembly Line with Multi-Manned Workstations." *Computers & Industrial Engineering* 133: 107–120. https://doi.org/10.1016/j.cie.2019.04.056.

Sahin, Murat, and Talip Kellegöz. 2019b. "Balancing Multi-Manned Assembly Lines With Walking Workers: Problem Definition, Mathematical Formulation, and An Electromagnetic Field Optimisation Algorithm." *International Journal of Production Research* 57 (20): 6487–6505. https://doi.org/10.1080/00207543.2019.1566672.

Scholl, Armin, and Christian Becker. February, 2006. "State-Of-the-Art Exact and Heuristic Solution Procedures for Simple Assembly Line Balancing." *European Journal of Operational Research* 168 (3): 666–693. https://doi.org/10.1016/j.ejor.2004.07.022.

Scholl, Armin, and Robert Klein. 1997. "Salome: A Bidirectional Branch-And-Bound Procedure for Assembly Line Balancing." *INFORMS Journal on Computing* 9 (4): 319–334. https://doi.org/10.1287/ijoc.9.4.319.

Scholl, Armin, and Stefan Voss. 1997. "Simple Assembly Line Balancing – Heuristic Approaches." *Journal of Heuristics* 2 (3): 217–244. https://doi.org/10.1007/BF00127358.

Sewell, E. C., and S. H. Jacobson. 2012. "A Branch, Bound, and Remember Algorithm for the Simple Assembly Line Balancing Problem." *INFORMS Journal on Computing* 24 (3): 433–442. https://doi.org/10.1287/ijoc.1110.0462.

Sungur, Banu, and Yasemin Yavuz. October, 2015. "Assembly Line Balancing with Hierarchical Worker Assignment." *Journal of Manufacturing Systems* 37: 290–298. https://doi.org/10.1016/j.jmsy.2014.08.004.

Vilà, Mariona, and Jordi Pereira. 2014. "A Branch-And-bound Algorithm for Assembly Line Worker Assignment and Balancing Problems." *Computers & Operations Research* 44: 105–114. https://doi.org/10.1016/j.cor.2013.10.016.

Walsh, Toby 2006. "General Symmetry Breaking Constraints." *Principles and Practice of Constraint Programming – CP* 4204: 650–664. https://doi.org/10.1007/11889205_46.

Weckenborg, Christian, Karsten Kieckhäfer, Christoph Müller, Martin Grunewald, and Thomas S. Spengler. April, 2020. "Balancing of Assembly Lines with Collaborative Robots."

Business Research 13 (1): 93–132. https://doi.org/10.1007/s40685-019-0101-y.

Yadav, Ashish, Shashank Kumar, and Sunil Agrawal. 2021. "Reconfiguration of Assembly Line Balancing–an Automobile Case Study Solved by the Exact Solution Procedure." *Benchmarking* 28 (5): 1576–1604. https://doi.org/10.1108/BIJ-08-2019-0386.

Yang, Wucheng, and Wenming Cheng. 2020. "A Mathematical Model and a Simulated Annealing Algorithm for Balancing Multi-Manned Assembly Line Problem with Sequence-Dependent Setup Time." *Mathematical Problems in Engineering* 2020. https://doi.org/10.1155/2020/8510253.

Yang, Hyungjoon, Je Hun Lee, Sang Hyun Lee, Seung Gi Lee, Hyung Rok Kim, and Hyun Jung Kim. July, 2022. "A Multi-Manned Assembly Line Worker Assignment and Balancing Problem with Positional Constraints." *IEEE Robotics and Automation Letters* 7 (3): 7786–7793. https://doi.org/10.1109/LRA.2022.3185784.

Yilmaz, Hamid, and Mustafa Yilmaz. 2015. "Multi-Manned Assembly Line Balancing Problem with Balanced Load Density." *Assembly Automation* 35 (1): 137–142. https://doi.org/10.1108/AA-05-2014-041.

Yilmaz, Hamid, and Mustafa Yilmaz. 2016. "A Multi-Manned Assembly Line Balancing Problem with Classified Teams: A New Approach." *Assembly Automation* 36 (1): 51–59. https://doi.org/10.1108/AA-04-2015-035.

Yilmaz, Hamid, and Mustafa Yilmaz. 2020. "A Mathematical Model and Tabu Search Algorithm for Multi-Manned Assembly Line Balancing Problems with Assignment Restrictions." *Engineering Optimization* 52 (5): 856–874. https://doi.org/10.1080/0305215X.2019.1618288.

Zakaraia, Mohammad, Hegazy Zaher, and Naglaa Ragaa. 2022. "Solving Stochastic Multi-Manned U-Shaped Assembly Line Balancing Problem Using Differential Evolution Algorithm." *International Journal of Production Management and Engineering* 10 (1): 13–22. https://doi.org/10.4995/ijpme.2021.16084.

Zangaro, Francesco, Stefan Minner, and Daria Battini. 2023. "The Multi-Manned Joint Assembly Line Balancing and Feeding Problem." *International Journal of Production Research* 61 (16): 5543–5565. https://doi.org/10.1080/00207543.2022.2103749.

Zhang, Zikai, Manuel Chica, Qiuhua Tang, Zixiang Li, and Liping Zhang. 2024. "A Multi-Objective Co-Evolutionary Algorithm for Energy and Cost-Oriented Mixed-Model Assembly Line Balancing with Multi-Skilled Workers." *Expert Systems with Applications* 236: 121221. https://doi.org/10.1016/j.eswa.2023.121221.

Zhang, Zikai, Qiuhua Tang, and Manuel Chica. 2021. "A Robust MILP and Gene Expression Programming Based on Heuristic Rules for Mixed-model Multi-manned Assembly Line Balancing." *Applied Soft Computing* 109: 107513. https://doi.org/10.1016/j.asoc.2021.107513.

Zhang, Zikai, Qiuhua Tang, Manuel Chica, and Zixiang Li. 2024. "Reinforcement Learning-Based Multiobjective Evolutionary Algorithm for Mixed-Model Multimanned Assembly Line Balancing Under Uncertain Demand." *IEEE Transactions on Cybernetics* 54 (5): 2914–2927. https://doi.org/10.1109/TCYB.2022.3229666.

Zhang, Zikai, Qiuhua Tang, Dayong Han, and Zixiang Li. 2022. "Multi-Manned Assembly Line Balancing with Sequence-Dependent Set-Up Times Using An Enhanced Migrating Birds Optimization Algorithm." *Engineering Optimization* 55 (7): 1243–1262. https://doi.org/10.1080/0305215X.2022.2067992.