European Journal of Operational Research 301 (2022) 797–814

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Invited Review

Assembly line balancing: What happened in the last fifteen years?

Nils Boysen[a,*], Philipp Schulze[b], Armin Scholl[b]

[a] Lehrstuhl für Operations Management, Friedrich-Schiller-Universität Jena, Carl-Zeiß-Straße 3, Jena 07743, Germany
[b] Lehrstuhl für Management Science, Friedrich-Schiller-Universität Jena, Carl-Zeiß-Straße 3, Jena 07743, Germany

ARTICLE INFO

Article history:
Received 20 August 2021
Accepted 22 November 2021
Available online 29 November 2021

Keywords:
Production
Assembly line balancing
Optimization
Survey

ABSTRACT

Ever since the times of Henry Ford up to today's industry 4.0 era, flow-oriented assembly processes, where an assembly line conveys the workpieces from workstation to workstation, are very important for mass-producers in manifold branches of industry. Among the most elementary optimization problems in this context is the assembly line balancing problem, which decides on the division of labor among the stations of an assembly line. This paper surveys the scientific literature on assembly line balancing that has been published since the last major review papers have appeared in 2006 and 2007, respectively. We cover all essential stages of the decision making process: we address novel methods to efficiently gather the relevant (precedence graph) data, review especially new problem variants and models treated in the literature, and survey the most important algorithmic developments. Furthermore, we outline a possible research agenda for the next fifteen years.

1. Introduction

To explain the importance of the simple assembly line balancing problem (SALBP) to an operations researcher who has not yet heard of this optimization problem, one could say that SALBP is the pendant for the production domain what the traveling salesman problem is for the transportation area. SALBP is an elementary optimization problem, which subdivides the total workload of an assembly process, consisting of a set of tasks subject to precedence constraints, among multiple stations that are arranged in a serial production process. Applying the notation summarized in Table 1, a basic mixed-integer programming (MIP) model formulation for SALBP-1 is given in Fig. 1(a).

The basic problem of any workload division is to maximize the output of a process for a given amount of available resources. Alternatively, a predefined output level can be fixed, while minimizing the costs for the necessary resources. SALBP-1 follows the latter approach and assumes a given output level, which can be realized if a given cycle time c (i.e., the maximum or average time available per station and workpiece) is not exceeded in any station. Given this output level, SALBP-1 aims to minimize the number of stations (and, thus, the wage costs, if each station is equipped with a single worker and each worker receives the same payment). Note that alternative model SALBP-2 takes the alternative view on the workload division trade-off and minimizes the cycle time for a given number of stations. The general aim of SALBP-1 is realized within objective function (1) by minimizing the station index to which final task n is assigned to. Constraints (2) ensure that each task is assigned to a station, and constraints (3) enforce that the cycle time is not violated in any station. Observation of precedence constraints are ensured by constraints (4), and, finally, constraints (5) define the domain of the variables.

Ever since the first mathematical formulation of an assembly line balancing (ALB) problem by Salveson (1955), this field has attracted plenty research. Dozens of exact and heuristic solution procedures for SALBP have been introduced. These efforts are, for instance, documented in the survey paper of Scholl & Becker (2006). In addition to SALBP, which reduces workload division to the very basics, a parallel research stream denoted general assembly line balancing problem (GALBP) has established. GALBP focuses on general ALB problems, where additional aspects such as U-shaped lines, parallel stations, or processing alternatives extend the basic problem. The differentiation of SALBP and GALBP has been introduced by Baybars (1986), and the research efforts in the field of GALBP are, for instance, summarized in the survey paper of Becker & Scholl (2006). Furthermore, Boysen, Fliedner, & Scholl (2007) provide a detailed classification scheme for the most important GALBP extensions.

However, the gap between the SALBP-1 model depicted within Fig. 1(a) and configuration planning of a real-world assembly line

* Corresponding author.
E-mail addresses: nils.boysen@uni-jena.de (N. Boysen), philipp.schulze@uni-jena.de (P. Schulze), armin.scholl@uni-jena.de (A. Scholl).
URL: https://www.om.uni-jena.de (N. Boysen), https://www.mansci.uni-jena.de (A. Scholl)

https://doi.org/10.1016/j.ejor.2021.11.043
0377-2217/

Table 1
Notation for SALBP.

n	number of tasks (index $j = 1, \ldots, n$)
V	set of tasks $V = \{1, \ldots, n\}$; tasks are numbered according to a topological ordering
E	set of precedence constraints $(i, j) \in E$ defining that task i has to precede j
$\overline{m}$	upper bound on the number of workstations (index $k = 1, \ldots, \overline{m}$)
K	set of potential stations $K = \{1, \ldots, \overline{m}\}$
p_j	processing time of task j
c	cycle time
x_{jk}	binary variable: 1, if task j is assigned to station k (0, otherwise)

$$\text{Minimize } \sum_{k \in K} k \cdot x_{nk} \quad (1)$$

subject to

$$\sum_{k \in K} x_{jk} = 1 \quad \forall j \in V \quad (2)$$

$$\sum_{j \in V} p_j \cdot x_{jk} \leq c \quad \forall k \in K \quad (3)$$

$$\sum_{k \in K} k \cdot x_{ik} \leq \sum_{k \in K} k \cdot x_{jk} \quad \forall (i, j) \in E \quad (4)$$

$$x_{jk} \in \{0, 1\} \quad \forall j \in V;\ k \in K \quad (5)$$

(a) Optimization model for SALBP-1

(b) Seat assembly at the Ford assembly plant in Claycomo, USA.

Fig. 1. Optimization model versus real-world assembly line (the picture on the right is public domain published under the CC BY 2.0 license).

(see Fig. 1(b)) with all its complexities is vast. Assembly lines are applied by mass-producers in many branches of industry, e.g., for the assembly of white-goods, consumer electronics, trucks, and airplanes (Boysen, Fliedner, & Scholl, 2008). The most famous application is certainly the assembly of cars. In a typical automotive assembly line, hundreds of workers have to assemble thousands of parts and need to be coordinated with machinery in hundreds of stations at a typical cycle time between 60 and 90 seconds (Battini, Boysen, & Emde, 2013). Due to these vast dimensions and the manifold interdependencies with related processes such as part logistics (see Boysen, Emde, Hoeck, & Kauderer, 2015), the layout design of a novel assembly line and the operational adaptations of existing lines become very challenging planning tasks, which involve large planning departments. In spite of all research efforts to support these complex planning tasks with suitable optimization procedures, it is our subjective impression from plenty factory visits in the past decades that the empirical findings already made in the 1970s (Chase, 1974) and 1980s (Schöniger & Spingler, 1989) are still valid today: Only a small percentage of companies apply sophisticated optimization procedures for setting up and operating their assembly lines. This is an ongoing challenge for the ALB research community, and this paper is intended to review the progress that has been made in the past fifteen years.

The remainder of the paper is structured as follows. In Section 2, we define the scope of this survey and describe our paper retrieval approach. As we see a demand for research along the complete (quantitative) decision making process, we address the following three stages each in a separate section: research efforts on data retrieval methods to obtain the input data for optimization, i.e., the precedence graph (Section 3), the new problems and models of ALB (Section 4), and the most important algorithmic developments of the past fifteen years (Section 5). Afterwards, it is our aim to recalibrate the research agenda for the next fifteen years. Specifically, we see three fruitful fields for future research (Section 6): more general models addressing the interdependencies of ALB with neighboring planning tasks, a customization of ALB for varying planning occasions during the life cycle of an assembly system, and breaking up the linear flow of assembly lines to enable the flexibilities targeted by smart factories of the industry 4.0 era.

2. Scope of survey

To precisely define the scope of this survey, we have to provide a definition of ALB. This, however, is no simple task, because the literature generally subsumed to the ALB field is vast and hundreds of extensions of SALBP have been introduced to support the planning tasks of real-world assembly lines. To give an impression of the huge variety of ALB problems, we summarize the basic assumptions contained in SALBP-1 (see Boysen et al., 2007) and give some examples from the ALB literature how each assumption can be relaxed (indicated by the → symbol):

- Mass-production of one homogeneous product. → While the single-model line was the standard innovated by Henry Ford, today's assembly lines are either multi-model lines, where significant setup operations are required to change from one production lot to another (see Burns & Daganzo, 1987; Dobson & Yano, 1994), or mixed-model lines, where setups are fast enough to produce in lot-size one (see Boysen, Fliedner, & Scholl, 2009b; Thomopoulos, 1967).
- All tasks are processed in a predetermined mode. → When setting up an assembly line, the inherent cost-performance tradeoff requires to either decide for a more costly but faster (automated) processing mode or a less costly but slower (manual) alternative. Integrating such a mode choice into ALB has a long-lasting tradition (e.g., see Bukchin & Tzur, 2000; Graves & Lamar, 1983; Pinto, Dannenbring, & Khumawala, 1983).
- Paced line with a fixed common cycle time according to a desired output quantity. → The standard case is a continuously

moving conveyor belt, which enforces observance of the cycle time per station and workpiece. However, unpaced assembly lines, where each station works at individual speed and input and output buffers in between subsequent stations regulate the handover of workpieces, require that the average cycle time is balanced, at least over a mid-term planning horizon (e.g., see Buzacott, 1967; Hillier, So, & Boling, 1993; Powell, 1994).

- A single serial assembly line is considered. → The assembly flow in real-world systems may be more complicated than a straight line, so that U-shaped lines (Miltenburg & Wijngaard, 1994; Urban, 1998), parallel stations (Bard, 1989; Buxey, 1974), or parallel line segments (Gökçen, Ağpak, & Benzer, 2006; Scholl & Boysen, 2009) need to be considered.
- The processing sequence of tasks is subject to precedence restrictions. → Various precedence constraints are the norm in assembly operations (e.g., first assemble the rubber seal, then the sunroof), so that they are a constituting element of ALB. However, alternative processing modes of a task may have alternative precedence constraints, so that also varying subgraphs within a precedence graph have been considered in the ALB literature (Scholl, Boysen, & Fliedner, 2009).
- Deterministic task times p_j. → Especially manual work (and its larger variation compared to automated tasks) is often better modeled by stochastic task times, so that it is not surprising that stochastic ALB also has a long-lasting tradition (e.g., see Kottas & Lau, 1976; Moodie & Young, 1965).
- No assignment restrictions of tasks besides precedence constraints. → For instance, a task requiring a lifted workpiece to be safely executed on a car's under-floor is incompatible with a simultaneous assembly of the sunroof. Such task incompatibilities have, e.g., been considered by Johnson (1983) and Bartholdi (1993).
- Tasks cannot be split among two or more stations. → Especially tasks whose durations exceed the cycle time can be assigned to multiple stations, so that task execution on subsequent workpieces can alternate. Then, each station has to execute the laborious task only on every other workpiece, which gives them time to recover. Split tasks have, e.g., been considered by Pinto, Dannenbring, & Khumawala (1975) and Boysen & Fliedner (2008).
- All stations are equally equipped with respect to machines and workers. → In real-world assembly systems, workstations vary considerably, e.g., with regard to the applied machinery and the number of workers whose skill levels may vary. In this case, SALBP-1's approach to minimize the number of stations may be a too strong simplification. Instead, cost-oriented (Amen, 2000; 2001) or profit-oriented (Rosenblatt & Carlson, 1985; Zäpfel, 1975) objective functions have to be applied.

To cover this wide range of possible extensions, we follow the definition given in Boysen et al. (2007) and define ALB (and, thus, the scope of this survey) as "all optimization models which aim at supporting that part of assembly line configuration that deals with grouping tasks and their required resources to stations", such that a coordination according to a cycle time is enabled.

This survey provides an update on our previous survey papers on SALBP (Scholl & Becker, 2006), GALBP (Becker & Scholl, 2006), and the ALB classification scheme (Boysen et al., 2007), which have appeared in 2006 and 2007, respectively. Our literature survey will show that a vast amount of new literature has appeared since then. Furthermore, all other surveys which have been published since then only cover specific aspects, e.g., two-sided assembly lines (Make, Rashid, & Razali, 2017), the requirements of different real-world applications (Boysen et al., 2008), ALB solved with soft computing (Fathi & Ghobakhloo, 2014; Rashid, Hutabarat, & Tiwari, 2012) or genetic algorithms (Tasan & Tunali, 2008), balancing of parallel lines (Lusa, 2008), cost and profit oriented ALB (Hazır, Delorme, & Dolgui, 2015), unbalanced lines (Hudson, McNamara, & Shaaban, 2014), disassembly line balancing (Özceylan, Kalayci, Güngör, & Gupta, 2019), and ALB in an industry 4.0 context (Dolgui, Sgarbossa, & Simonetto, 2021). A bibliographic analysis is proposed by Eghtesadifard, Khalifeh, & Khorram (2020). The only exception is the paper of Battaïa & Dolgui (2013), which is also a general survey paper covering the complete field of ALB. However, plenty more papers have been published since then, and they do not apply our concise classification scheme for ALB (Boysen et al., 2007) to precisely and systematically explore the existing literature. Therefore, we think that yet another general survey paper on ALB is well justified.

Finally, we briefly specify our paper retrieval procedure (for a general description of how to set up a systematic literature search see, e.g., Hochrein, Glock et al., 2013). We started with a search in two scholarly databases, namely Science Direct and Business Source Premier. As search string we applied "assembly line" AND ("line balancing" OR "balancing problem") OR "production line balancing". All English-language papers published in peer-reviewed journals that have been retrieved and those cited in their reference lists (snowball approach) were checked for relevance, novelty, and significance of their contributions to the field, at least by analyzing their abstracts. Since we build up on our aforementioned survey papers, we only consider papers published after 2006 (except for noteworthy exceptions). To precisely record the problem settings of the obtained ALB papers, we apply our classification scheme (Boysen et al., 2007) and classify every paper that fits into this scheme. Furthermore, we discuss novel extensions that are not already within our classification scheme. In order to keep the paper short and readable, the classification is given in the Online-Appendix, while the most noteworthy contributions are explicitly described in the text. In total, we examined more than 1000 papers in the retrieval phase and selected almost 500 relevant and novel contributions for this survey.

3. Data retrieval: how to determine the precedence graph

To model and solve real-world ALB problems, plenty of (valid and reliable) data is required. Since more than 65 years of ALB research, the vast majority of scientific papers simply state that the problem data has to be collected and provided as an input to models and solution procedures. Only a few application-oriented researchers have ever dealt with the question how to actually obtain the necessary data in a suitable and effective manner. According to our experience, this is one major reason for the lack of applying ALB solution approaches in practice. In the following, we highlight some recent approaches that focus on data retrieval of the precedence graph.

As defined for SALBP-1 in Table 1 and Fig. 1(a), the core data of each line balancing problem are the tasks, their task times, and the precedence relations among them. These elements are visualized by a (directed) precedence graph with each task being represented by a node $j \in V$; task times p_j are given as node weights. A precedence relation (i, j) means that task $i \in V$ must be finished before task $j \in V$ can be executed (e.g., rubber seal before sunroof). The direct precedence relations define the arc set E of the precedence graph, while indirect ones are represented by chains of two ore more arcs from E. Fig. 2 shows an example of a precedence graph with $n = 6$ tasks.

In practice, there are established procedures to subdivide the total work content into tasks and to estimate task times by so-called predetermined motion time systems like MTM – "Methods-Time Measurement" (Maynard, Stegemerten, & Schwab, 1948) or MOST – "Maynard Operation Sequence Technique" (Zandin, 2002). Such methods subdivide assembly tasks into elementary motions

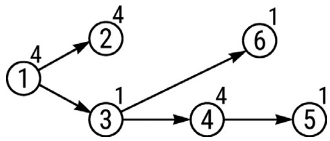

Fig. 2. Precedence graph.

such as lifting and gripping, whose durations are pre-defined based on an in-depth analysis of the typical motions of an average worker. The task times are obtained by aggregating the interconnected motion times in a suitable manner. Beyond task times, it is also necessary to evaluate the ergonomic conditions under which assembly tasks are performed, in order to design healthy and efficient working places. The method EAWS – "European Assembly Worksheet" (Schaub, Caragnano, Britzke, & Bruder, 2013), for instance, is used by many major European automobile manufacturers. More recent systems like MTM-HWD – "Human Workplace Design" integrate time and ergonomic analyses (e.g., Faber et al., 2019). These methods are widely used in practice and continuously improved based on modern digitization techniques such as digital human models (e.g., Caputo, Greco, Fera, & Macchiaroli, 2019; Schaub et al., 2012) and virtual reality (e.g., Kunz, Zank, Nescher, & Wegener, 2016).

While task-related data thus can be collected in an established manner by the mentioned tools, there is still a lack of practicable methods to obtain the required precedence relations. Traditionally, research focused on explicitly detecting all or some promising feasible sequences of task execution rather than on the complete precedence graph, which implicitly represents all feasible sequences. The traditional proposals include expert interviews, the usage of CAD data, and case-based reasoning approaches (see the summary in Klindworth, Otto, & Scholl, 2012). None of these methods, however, is widely applied in industry, because they come for the price of extensive manual effort (e.g., interview-based techniques) and often produce incomplete or incoherent results that can hardly be kept up-to-date during the steady evolution of products and processes. For example, CAD models often lack the specification of soft components such as clips and seals (Altemeier, 2009) and are often not available for all components of the product. As a consequence, precise precedence information is often not available and/or up-to-date, so that most manufacturers do not use automated or semi-automated decision support but employ numerous human planners to solve the line balancing task manually.

One promising approach to overcome these problems is denoted the *learning precedence graph concept* and was proposed by Klindworth et al. (2012) based on findings of Minzu, Bratcu, & Henrioud (1999), Altemeier (2009), and Altemeier, Brodkorb, & Dangelmaier (2010). Antani et al. (2013, 2014) show that this approach is actually helpful in practical applications.

The procedure of Klindworth et al. (2012) assumes that there is an (at first) unknown precedence graph (called *target graph TG*), which is approximated bottom-up and top-down as follows:

- A *maximum graph MxG* is a supergraph of *TG* with the same set of nodes and a superset of arcs (i.e., *MxG* is at least as restricting regarding the line balance as *TG*). It is obtained from feasible (historic) task sequences (FTS), which have been used in former production periods. The procedure starts with a single FTS, so that *MxG* is a chain of all tasks as defined by the first feasible production sequence. With each additional FTS, obtained from altered production processes of later periods, arcs can be removed from *MxG* as follows: If there is an arc (i, j) in *MxG* and task j is at an earlier position of the new FTS than i, then arc (i, j) is contradicted and can be removed. There is obviously no precedence relation between tasks i and j, because there exist FTS where i is executed before j and vice versa. This process is repeated as long as new FTS are available and further precedence relations can be removed from *MxG*. In each iteration of this learning approach, the current maximum graph *MxG* can be applied to (heuristically) balance the line, because it guarantees feasible line balances. Note that removing arcs requires adjusting the transitive closure of *MxG* in each iteration.
- A *minimum graph MnG* is a subgraph of target graph *TG* with the same set of nodes and a subset of the arcs (i.e., *MnG* is at most as restricting regarding the line balance as *TG*). Starting with an empty arc set, precedence relations that definitely exist are added, once they are verified by a traditional approach such as an expert interview or the interpretation of CAD data.
- In this manner, precedence relations are subsequently removed from *MxG* in a bottom-up approach and added to *MnG* in a top-down manner. Whenever $MxG = MnG$ is achieved, the target graph *TG* is finally detected, without having to establish each precedence constraint manually. Naturally, real-world precedence graphs are large, and, thus, *TG* is not easily obtainable by this approach. However, the experiments of Klindworth et al. (2012) for SALBP-1 show that near-optimal line balances can be obtained from *MxG* even if there remains a considerable gap between *MxG* and *MnG*. Even optimality of a line balance can be confirmed, whenever *MxG* and *MnG* result in the same number of stations. This is a very useful feature for practice: We can find feasible and even optimal solutions to a problem, whose data is only partly known.

Example: Let the precedence graph of Fig. 2 be the unknown target graph *TG*. Given a cycle time of $c = 5$ for SALBP-1, $FLB_1 = (\{1\}, \{2, 3\}, \{4, 5\}, \{6\})$ represents a first feasible line balance with four stations (the sets represent the station loads from station 1–4), found by manual planning. The resulting first *MxG* is just a chain according to the given sequence $FTS_1 = (1, 2, 3, 4, 5, 6)$. After some time, the line gets re-balanced, so that we receive a second assembly sequence $FTS_2 = (1, 3, 6, 4, 5, 2)$ with line balance $FLB_2 = (\{1\}, \{3, 6\}, \{4, 5\}, \{2\})$. Merging the information of FTS_1 and FTS_2, leads to *TG*: task 2 has no successors (i.e., it is the last task in FTS_2), and task 6 is executed before its former predecessors 4 and 5 in FTS_2. Thus, all these contradicted precedence relations can be removed.

To reduce the gap between *MxG* and *MnG* in real-life applications, a rather long learning phase and a lot of effort with determining confirmed precedence relations are required. Furthermore, planners often tend to not considerably change an existing line balance and rather make small adaptations. Naturally, this reduces the potential to remove many precedence constraints from *MxG*. Confirming precedence relations is often a laborious task, which takes experts plenty of time. To accelerate the learning process, Otto & Otto (2014b) extend the approach by deducing many precedence relations between tasks from a high-level module graph, which is often available in the real world. Furthermore, they propose to identify critical precedence relations, so that the effort for experts when confirming precedence relations can be focused on the elementary ones.

Another problem related to obtaining reliable precedence data is due to the ever increasing product variety that is produced on today's assembly lines. Traditional approaches to obtain a joint precedence graph for multiple products that are jointly produced on the same line (e.g. Macaskill, 1972; Thomopoulos, 1970) presuppose that each product comes along with its own individual precedence graph. Nowadays, however, customers can customize their products by (de-)selecting different options, and modern cars provide hundreds of options to choose from. Hence, individual precedence graphs for each possible product variant are not at hand. Instead, Boysen, Fliedner, & Scholl (2009a) propose an option-based

Table 2
Structure of the classification scheme.

Precedence graph characteristics (α)		**Resource characteristics** (β)	
α_1	Product specific precedence graphs	β_1	Movement of workpieces
α_2	Structure of the precedence graph	β_2	Line layout
α_3	Processing times	β_3	Parallelization
α_4	Sequence-dependent task time increments	β_4	Resource assignments
α_5	Assignment restrictions	β_5	Station-dependent time increments
α_6	Processing alternatives	β_6	Additional configuration aspects
Objectives (γ)			

scheme for generating a joint precedence graph for mixed-model production. In this single graph, options are represented by subgraphs that occur with a specific probability equal to the occurrence of the specific option.

When comparing the few papers reviewed in this section with the large number of publications of the next section on new ALB extensions, we can conclude that data retrieval is still not in the focus of ALB research. This, however, does not mirror the outstanding importance of generating reliable problem data in real-world applications. Thus, we want to encourage the next generation of ALB researchers to not repeat the mistake of their predecessors. The application of machine learning techniques to support precedence graph generation, for instance, should be a fruitful field for future research. For instance, it could be possible to learn from past CAD drawings of parts and their relative positions in the motor compartment whether or not this leads to precedence constraints. In this way, a machine-generated proposal list for potential precedence relations could be derived in order to further speed up the data retrieval process.

4. Problems and models: new ALB extensions

Once the necessary input data is available, the relevant part of an assembly line balancing problem (ALBP) has to be transferred into a suitable optimization model. As elaborated above, ALB research has to cover a wide range of practical assembly settings, so that there was always a wide arsenal of different yet closely related problem settings and models. In the following, we summarize central new developments in ALB research. We subdivide the text according to current research trends with respect to extended and new decision problems (each trend gets a shortcut given in the subsection title). In each case, we focus on important papers, which have created the considered trend or made a particularly important contribution to it. This selection is due to our own experience-based evaluation. Where available, the established problem names and abbreviations as well as the respective classification identifiers are given. These and further works can be found in the extensive classification provided in the Online-Appendix, where we specify the classification tuple according to Boysen et al. (2007), the methodical contributions, and the trends it refers to (if existent) for each paper. Table 2 summarizes the structure of the classification scheme, the entire scheme is explained in the Online-Appendix.

4.1. Flexible and parallel work (FPW)

Due to the ever increasing complexity and variety of products manufactured on assembly lines, flexibility and parallelism are increasingly important in order to allow for an efficient production of customized products. In car production, for instance, not only different models of the same car type but also considerably differing car types (e.g., big SUVs and small city cars) are manufactured on the same line. Two main approaches for making lines more flexible is to introduce some degree of parallelism or to arrange stations more flexible than the traditional serial line layout.

4.1.1. Parallel workplaces

Two-sided ALB (TALBP, $\beta_3 = pwork^2$): The two-sided ALBP, initially introduced by Bartholdi (1993), assumes two workplaces in each station, one at the left-hand side and one at the right-hand side of the line. Some of the tasks are fixed to one of the sides, others can be performed on either side. The latter type of tasks leads to a complicating restriction, which requires an explicit scheduling of tasks within the workplaces: a workplace executing a task j has to wait for the opposite workplace if there a preceding task of j is performed ("workplace interaction problem" for short).

Recently, some interesting extensions to TALBP have been proposed, which we elaborate in the following. For a more comprehensive overview on different versions of TALBP, see Make et al. (2017); different heuristic approaches for TALBP are surveyed and benchmarked by Li, Kucukkoc, & Nilakantan (2017):

- Typically, large workpieces are manufactured on a two-sided line. Some of the tasks require cooperation of the workers of both opposing stations in a *synchronized manner* (e.g., for jointly mounting the bumper of a bus). This problem extension (new symbol $\alpha_5 = sync$) has been introduced by Simaria & Vilarinho (2009). This problem setting is also considered in a remarkable number of follow-up papers (see Online-Appendix, Section A.2.2), which additionally consider further trends like assignment restrictions, multiple objectives and meta-heuristic search procedures.
- Since most real-world assembly lines produce multiple products, TALBP is extended to *mixed-model lines* ($\alpha_1 = mix$) based on average or even model-specific task times by, e.g., Simaria & Vilarinho (2009), Özcan & Toklu (2009a), Huang, Mao, Fang, & Yuan (2021).
- Kucukkoc, Li, Karaoglan, & Zhang (2018) extend TALBP to a three-sided problem by additionally considering *underground workplaces*, which perform work underneath large workpieces (e.g., buses) that cannot be titled by an overhead conveyor.

Multi-manned ALB (MMALBP, $\beta_3 = pwork$): Large workpieces allow for several workers mounting parts at the same station simultaneously. For example, up to four workers work in parallel without excessive interference at automobile assembly lines. That is, each station can contain one, two, three, or four parallel workplaces that are bound by the workplace interaction problem. Different MMALBP versions concerning the flexibility of workplace layouts and mutual interference of workers are proposed:

- A basic MMALBP considers *multiple flexible workplaces* per station (with a product-specific maximum degree of parallelism). In contrast to TALBP, the workplaces are not fixed to certain areas of the station but flexibly result from the assigned tasks (e.g. Dimitriadis, 2006; Kellegöz & Toklu, 2012; Michels, Lopes, Sikora, & Magatão, 2019; Roshani & Giglio, 2016).
- Lopes, Pastre, Michels, & Magatão (2020a) consider a variation of the problem, called *Flexible multi-manned ALB (FMMALBP)*, where the workplaces are located without a rigid station scheme, only defined by start and end times, in order to reduce the line length.

- In practice, it is not possible to define workplaces arbitrarily, because workers must not interfere with each other by working at the same part of the workpiece at the same time. This problem is avoided by defining a grid of *mounting positions* associated with the workpiece and the station space around it. Assigning tasks to a workplace induces a set of required mounting positions flexibly defining the area of the workplace. To avoid interference, it is not allowed that a mounting position is assigned to more than one workplace. Furthermore, to get compact workplaces, the assigned mounting positions should be neighboring or at least be near to each other. Becker & Scholl (2009) call this problem *ALB with variable workplaces (VWALBP)* and additionally include assignment restrictions and extra-long tasks which are to be performed by subsequent stations (multi-station workplaces).
- Sternatz (2014) *simplifies VWALBP* by avoiding the workplace interaction problem. The latter is an NP-hard scheduling subproblem for each station given the task assignment to its workplaces. The simplification is based on an observation from practice: Managers strictly avoid assigning precedence-related tasks to two or more parallel workplaces in order to keep line control simple and error-resistant. Furthermore, in a mixed-model setting, task times vary from cycle to cycle such that a detailed scheduling of tasks within stations seems even counterproductive.

4.1.2. Parallel ALB (PALBP, $\beta_3 = pline$)

Parallel assembly lines are a common lever to increase productivity, flexibility, and robustness of an assembly system. Gökçen et al. (2006) were the first to formulate an independent decision problem (*PALBP* for short), which *simultaneously balances parallel lines*. By installing so-called multi-line stations that share work among lines, the total number of required workers can be reduced. The worker of a multi-line workplace receives tasks from parallel stations of two adjacent lines, so that within her cycle she has to switch among lines. Naturally, this requires that neighboring lines are near to each other, such that the setup times for walking to the other line are negligible. Multi-line workplaces connect the line balancing problems of all parallel lines to a single combined problem. Several extensions have been considered (for more detailed surveys on parallel and multiple line layouts and related decision problems see Aguilar, García-Villoria, & Pastor, 2020; Lusa, 2008):

- Scholl & Boysen (2009) extend PALBP by additionally considering the *assignment of multiple products (models) to the parallel lines* (called *MPALBP*). In this way, the reduction effect of workplaces is increased if compatible models that support a workplace sharing are assigned to neighboring lines.
- Gökçen, Kara, & Atasagun (2010) consider *multiple straight assembly lines* (*MSLB*), which are connected in several ways to obtain flexibility and to reduce the required workforce. Besides the concept of parallel lines, they consider a *consecutive* and *perpendicular arrangement* of straight lines to form multi-line workplaces at the connecting stations where two lines meet. With these arrangements, they aim to achieve flexible assembly systems (following the Shojinka-principle of the well-known Toyota production system) even if U-shaped lines are not possible (see Section 4.1.3).
- The balancing of multiple *parallel two-sided lines* (PTALBP) is considered by Özcan, Gökçen, & Toklu (2010). As discussed for TALBP in Section 4.1.1, additional restrictions for assigning tasks to the sides of the lines have to be observed.

4.1.3. ALB with U-lines (UALBP, $\beta_3 = u$)

A U-shaped line (U-line for short) is another layout type, which is intended to increase flexibility and productivity. The stations are not arranged in a straight line but in a U-form such that workers can perform tasks on two stations within the same cycle in so-called *cross-over workplaces*. The balancing of U-lines as an optimization problem (UALBP for short) was first considered by Miltenburg & Wijngaard (1994). Since then, a large number of papers on U-lines has accumulated. We highlight the main U-line research trends and give selected references in the following:

- *Mixed-model* U-lines face the operational complexity of the varying model-dependent task times. This problem is approached by considering a joint balancing and sequencing problem (e.g., Kara, Ozcan, & Peker, 2007; Özcan, Kellegöz, & Toklu, 2011) and/or extended objective functions that limit or minimize time variations among models (e.g., Hwang & Katayama, 2009; Rabbani, Kazemi, & Manavizadeh, 2012).
- Analogously to straight lines, even U-lines may be connectable with some type of *parallelism*, although it seems rather complicated to nest multiple U-lines into a clear overall layout. Parallel (two-sided) workplaces for U-lines are considered by, e.g., Ağpak, Yegül, & Gökçen (2012) and Delice, Aydoğan, Özcan, & İlkay (2017), parallel stations by Hamzadayi & Yildiz (2012), and parallel U-lines by Kucukkoc & Zhang (2015, 2017).
- The assumption of deterministic and static task times is relaxed for U-lines in alternative ways: *Non-deterministic processing times* are considered, e.g., by Ağpak & Gökçen (2007) (stochastic times, $\alpha_3 = t^{sto}$), Hazır & Dolgui (2015) (interval times, $\alpha_3 = t^{int}$), and Alavidoost, Babazadeh, & Sayyari (2016) (fuzzy times, $\alpha_3 = t^{fuz}$). *Dynamic task times* ($\alpha_3 = t^{dy}$) due to learning effects are integrated into U-lines by Toksarı, İşleyen, Güner, & Baykoç (2008). Based on processing alternatives, Kara, Özgüven, Yalçın, & Atasagun (2011) and Bukchin & Raviv (2018) deal with *resource-dependent task times*. Further aspects of non-deterministic processing times and processing alternatives are discussed in Section 4.4 and Section 4.9, respectively.

4.1.4. ALB with traveling workers (TWALBP)

The basic idea behind U-lines is that workers can share their workload among stations located on different sides of the U. Sharing workload among different stations, however, is also possible in straight lines, which is considered by Sikora, Lopes, & Magatão (2017). Analogously to UALBP, TWALBP assumes that workers can be assigned to more than one station observing cycle time and precedence relations. Due to the considerable distances between stations in a serial line, however, the walking times of workers among stations are considered explicitly. That is, the sum of task times and movement times must not exceed the cycle time for each worker. To determine minimum movement times, a traveling salesman problem has to be solved as a subproblem for each worker. A similar problem in a multi-manned setting is considered by Sahin & Kellegoz (2019).

4.2. Worker assignment (WA)

If the simplifying assumption that all stations are equipped with exactly one homogeneous worker is removed (see Section 2), the assignment of workers to stations becomes part of the decision. Workers mainly differ in terms of processing speed and qualification level ($\alpha_6 = pa$).

- The case where workers have different processing speed (i.e., worker-dependent task times) is considered in the so-called *Assembly Line Worker Assignment and Balancing Problem (ALWABP)* (e.g., Blum & Miralles, 2011; Miralles, García-Sabater, Andrés, & Cardos, 2007; Miralles, García-Sabater, Andrés, & Cardós, 2008; Pereira, 2018). The problem is inspired by sheltered work centers for workers with disabilities (Miralles et al., 2007). Tasks and workers are simultaneously assigned to stations, such that

some efficiency measure, e.g., the cycle time, is optimized ($\beta_4 = equip$). A version of ALWABP with stochastic worker availability is considered by Ritt, Costa, & Miralles (2015).
- A quite similar problem setting (with the same classification tuple $\beta_4 = equip$) is called the *Assembly Line Worker Integration and Balancing Problem (ALWIBP)* (e.g., Moreira, Miralles, & Costa, 2015b; Moreira, Pastor, Costa, & Miralles, 2017). Here, a distinction is only made between two types of workers in terms of processing speed: Either a worker with or one without disabilities has to be assigned to each station and the given workforce of workers with disabilities has to be integrated into the line.
- Different worker qualifications (or skill levels) are considered in the *ALB with Hierarchical Worker Assignment (ALBHWP)* (Sungur & Yavuz, 2015). Each task requires a minimum skill level. The workers have to be assigned to the stations in such a way that they can perform all the assigned tasks based on their qualification ($\beta_4 = equip, res^{\max}$). Since workers with a higher skill level cause higher wage costs, the goal is to find a station-worker-task-assignment that minimizes total costs.

4.3. Robotic ALB (Rob)

In the *Robotic ALB* (*RALBP*, see Rubinovitz, Bukchin, & Lenz, 1993), each station is equipped with a single robot from a set of different robot types ($\beta_4 = equip$). The individual processing times of the tasks depend on the robot type assigned to the respective station ($\alpha_6 = pa$). Thus, the ALB problem is connected with a *robot selection problem*. Concerning the problem structure, similarities to the worker assignment problems exist (see Section 4.2). Different versions of RALBP have been proposed (a more detailed survey is given by Chutima, 2020):

- In a considerable number of papers, the cycle time is minimized (e.g., Borba, Ritt, & Miralles, 2018; Gao, Sun, Wang, & Gen, 2009) in order to efficiently utilize robots and increase the line's *productivity*.
- Many papers focus on alternative *cost functions* that contain one or more of the following cost types: robot installation and life cycle costs (e.g., Pereira, Ritt, & Vásquez, 2018), robot setup costs (e.g., Rabbani, Mousavi, & Farrokhi-Asl, 2016; Yoosefelahi, Aminnayeri, Mosadegh, & Ardakani, 2012), energy costs (e.g., Li, Tang, & Zhang, 2016; Nilakantan, Huang, & Ponnambalam, 2015), robot-dependent task execution costs (Nilakantan, Nielsen, Ponnambalam, & Venkataramanaiah, 2017; Pereira et al., 2018).
- Analogously to manual assembly lines, different further *extensions* such as mixed-model production, different types of parallelism, U-lines, and sequence-dependent setup times can be combined with robots (e.g., Aghajani, Ghodsi, & Javadi, 2014; Çil, Mete, Özceylan, & Ağpak, 2017; Li et al., 2016; Lopes et al., 2017; Nilakantan, Li, Bocewicz, Banaszak, & Nielsen, 2019; Zhang, Tang, Li, & Zhang, 2019).

4.4. Non-deterministic processing times (NDPT)

Due to variation of human work pace, machine failures, or breakdowns, task times can be variable. While former research focused on stochastic processing times based on theoretic probability distributions, more recent research also addresses interval and fuzzy processing times:

- *Stochastic processing times* ($\alpha_3 = t^{sto}$): A considerable number of papers consider SALBP-versions with stochastic task times. Besides proposing solution procedures, stability analyses of solutions are performed (e.g. Gurevsky, Battaïa, & Dolgui, 2012; Lai, Sotskov, Dolgui, & Zatsiupa, 2016). Stochastic ALBPs combined with two-sided workplaces, parallelism and/or U-shaped lines are proposed by Ağpak & Gökçen (2007), Özcan (2010, 2018), Tang, Li, Zhang, & Zhang (2017). An unpaced line setup with synchronized conveyor movement is analyzed for stochastic task times, e.g., by Chiang, Urban, & Xu (2011) and Urban & Chiang (2016). Stochastic unpaced lines with buffers and further extensions such as mixed-model production, parallel stations, and resource selection are considered by, e.g., Tiacci (2015a, 2015b, 2017).
- *Interval processing times* ($\alpha_3 = t^{int}$): As a more convenient way to model variable task times in practice, Hazır & Dolgui (2013) consider interval processing times with a nominal value for normal task execution as the lower bound and a worst case value as the upper bound of the intervals. Based on these interval times, min-max robust optimization versions of ALB are developed with different objective functions. Specifically, Hazır & Dolgui (2013) minimize the cycle time, and Gurevsky, Hazır, Battaïa, & Dolgui (2013) as well as Pereira & Álvarez-Miranda (2018) minimize the number of stations. Moreover, Moreira, Cordeau, Costa, & Laporte (2015a) and Pereira (2018) combine interval times with worker assignment (see Section 4.2), while Hazır & Dolgui (2015) consider U-lines with interval times.
- *Fuzzy processing times* ($\alpha_3 = t^{fuz}$): Non-deterministic processing times can also be modeled by fuzzy numbers (i.e., typically based on a triangular membership function) to define a fuzzy ALB (*FALBP* for short). A basic FALBP with the objective of maximizing the line efficiency is considered by Zacharia & Nearchou (2013). Further problem versiona are obtained by combining FALBP with multiple objectives and/or U-lines (e.g., Alavidoost, Tarimoradi, & Zarandi, 2015; Babazadeh & Javadian, 2019; Zacharia & Nearchou, 2012) as well as with different worker skills (Salehi, Maleki, & Niroomand, 2018).

4.5. Sequence-dependent task time increments (ST)

Usually, tasks times are assumed to be independent of the mounting positions of the tasks and the assembly state of the workpiece. In practice, however, this assumption is often not justified, because walking times and tool changes significantly contribute to station times. Furthermore, parts mounted before may complicate adding a current part and, thus, lead to increased processing times.

- *General ALB with setups* (*GALBPS*, $\alpha_4 = \Delta t_{dir}$): Andrés, Miralles, & Pastor (2008) and Pastor, Andrés, & Miralles (2010) define the basic problem, which contains setup times between two tasks executed in direct succession by the same worker. If these tasks do not correspond to the same mounting position or require additional parts or other tools, a sequence-dependent increment to the "normal" task time becomes relevant. Scholl, Boysen, & Fliedner (2013) additionally distinguish between forward setup times if two tasks are executed in the same cycle and backward setup times when a worker has to transfer to the next workpiece in between two tasks. This extended case is also considered by Esmaeilbeigi, Naderi, & Charkhgard (2016) and Zohali, Naderi, & Roshanaei (2021). Different extensions of the problem with respect to mixed-model production, learning effects, and/or parallelism are also proposed (see Akpinar, Elmi, & Bektaş, 2017; Akpinar, Mirac Bayhan, & Baykasoğlu, 2013; Hamta, Fatemi Ghomi, Jolai, & Akbarpour Shirazi, 2013; Özcan, 2019; Özcan & Toklu, 2010).
- *Sequence-dependent ALB* (*SDALBP*, $\alpha_4 = \Delta t_{ind}$): In this problem version, introduced by Scholl, Boysen, & Fliedner (2008), the task time increments do not depend on the direct succession of tasks but consider the assembly state of the workpiece that depends on the previous tasks already executed. The "normal" task time refers to an unobstructed task execution, which how-

ever may be complicated if other parts at the same or near-by mounting positions are already installed. In this case, task time increments are considered.

4.6. Ergonomics (Ergo)

In the last decade, ergonomic aspects have gained increasing attention in practice and research due to an aging workforce, especially in many industrialized countries. Workers are exposed to a tremendous *physical and mental stress* when completing repetitive assembly work over a longer time. Otto & Battaïa (2017) provide a recent survey on research focusing on the integration of ergonomic aspects into ALB and job rotation as another lever to reduce ergonomic stress over time. There exist quite a few alternative approaches depending on the focused stress factor (e.g., repetitiveness, static postures, unhealthy positions, required forces, and energy expenditures) and suitable *methods for risk assessment* (like NIOSH, OCRA, and EAWS). For details we refer to Otto & Battaïa (2017). We concentrate on the following main trends in *ergonomic ALB* (*ErgoALBP*):

- Some papers include ergonomic aspects as *restrictions* ($\alpha_5 = cum$). Based on a given assessment method, the ergonomic risks of a workplace, depending on the assigned tasks, is evaluated and limited by a predefined maximum ergonomic risk value (e.g., Kara, Atasagun, Gökçen, Hezer, & Demirel, 2014; Mutlu & Özgörmüş, 2012; Otto & Scholl, 2011; Sternatz, 2014).
- Alternatively, the ergonomic situation can be improved by including an ergonomic term in the *objective function* (e.g., Battini, Delorme, Dolgui, Persona, & Sgarbossa, 2016b; Choi, 2009; Ozdemir et al., 2021; Rajabalipour Cheshmehgaz, Haron, Kazemipour, & Desa, 2012). In automotive practice, a traffic light system (green: no risk, yellow: moderate risk, red: large risk) is frequently applied to asses the risk level of each workplace (Sternatz, 2014). A straightforward practical system can then exclude red workplaces by restrictions, while minimizing the number of yellow workplaces (Otto & Scholl, 2011).
- Finally, ergonomic aspects are also combined with *other practical aspects*, such as mixed-model production, parallel and/or U-lines (e.g., Mokhtarzadeh, Rabbani, & Manavizadeh, 2021; Sgarbossa, Battini, Persona, & Visentin, 2016; Tiacci & Mimmi, 2018), worker assignment (e.g., Akyol & Baykasoğlu, 2019; Zhang, Tang, Ruiz, & Zhang, 2020c), space constraints (Bautista, Batalla-García, & Alfaro-Pozo, 2016), and part feeding (e.g., Battini, Calzavara, Otto, & Sgarbossa, 2017; Bortolini, Faccio, Gamberi, & Pilati, 2017).

4.7. Additional assignment restrictions (AAR)

In SALBP, the task-station-assignment is only restricted by precedence relations and the cycle time. Real-world assembly lines often require the consideration of further assignment restrictions, such as linked or incompatible tasks, fixed or forbidden stations, and minimum or maximum distances between tasks. These "standard" assignment restrictions are well-known in ALB research (see the overview in Scholl, Fliedner, & Boysen, 2010). In the recent years, these and more specific assignment restrictions have been considered in order to cover the demands of real-world assembly lines in a more detailed manner:

- Scholl et al. (2010) extend SALBP to *ARALBP* (*Assignment-Restricted ALB*, $\alpha_5 = \{link, inc, cum, excl, fix, type, min, max\}$). Here, minimizing the number of stations is restricted by all kinds of additional assignment restrictions.
- In the *Time and Space constrained ALB* (*TSALBP*, $\alpha_5 = cum$), Bautista & Pereira (2007) consider an available space A per station, which is required for material supply. Each task j has a (station-independent) space requirement a_j depending on its part supply mode (e.g., parts are provided in small or large boxes). The space restriction is a cumulated assignment restriction that guarantees that the sum of space required by the assigned tasks per station does not exceed A. Mathematically, it has the same form as the cycle time restriction such that it contains a time and a space "capacity" restriction. Further papers consider different types of TSALBP versions including single or multiple objectives, robustness, multi-manned lines, and a third "capacity" constraint for limiting the ergonomic stress per station (Bautista et al., 2016; Chica, Bautista, Cordón, & Damas, 2016; Chica, Cordón, Damas, & Bautista, 2013; Zhang, Tang, & Chica, 2020b).
- While the above problems are direct extensions of SALBP, different researchers combine additional assignment restrictions *with other problem extensions*, such as parallelism, worker assignment, and mixed-model production (e.g., Aghajani et al., 2014; Miralles et al., 2007; Simaria & Vilarinho, 2009; Sternatz, 2014; Zhang, Hu, & Wu, 2020a).

4.8. Accessibility windows (AW)

In some cases, station areas are not long enough that the entire workpiece can be accessed at the same time. In this case, it is required to consider a so-called moving accessibility window, i.e., the mounting positions that are in reach of the worker at a certain point in time t.

The *Accessibility Windows ALB* (*AWALBP*, $\alpha_2 = spec; \alpha_5 = fix$) of Calleja, Corominas, García-Villoria, & Pastor (2013) connects different decision problems: the assignment of tasks to stations, the movement scheme of the line, the machine configuration in case of automated stations, and the line configuration (line length and type of stations). Different subproblems containing one or more of those interdependent decisions are considered, e.g., by Müller-Hannemann & Weihe (2006), Calleja et al. (2013), Fleszar (2017), and García-Villoria, Corominas, Nadal, & Pastor (2018).

4.9. Processing alternatives (PA)

Traditionally, ALB problems assume that the production process (tasks, their times, and precedence relations) is fixed and given. In practice, however, there are often processing alternatives, which provide different processing times and/or costs. Recent research focused on the following ALB problems with processing alternatives:

- The *Alternative subgraphs ALB* (*ASALBP*, $\alpha_6 = pa^{subgraph}$) considers a precedence graph, which contains alternative subgraphs (i.e., different tasks, task times, and/or precedence relations) for a sub-process to realize a specific product function. In the basic case that directly extends SALBP-1, the ALB decision is connected with the selection of one subgraph per alternative, such that the number of required stations is minimized (e.g., Capacho & Pastor, 2008; Capacho, Pastor, Dolgui, & Guschinskaya, 2009; Scholl et al., 2009).
- The *Resource dependent ALB* (*RDALBP*, $\alpha_6 = pa$) connects ALB with an equipment selection problem, such that the task times are equipment-dependent. In the objective function, the costs for the selected equipment are aggregated with other cost types or alternative (time or capacity-oriented) objectives (e.g., Kara et al., 2011; Triki, Mellouli, & Masmoudi, 2017). These problem versions build up on the previously considered assembly line design problem (see Bukchin & Tzur, 2000; Pinto et al., 1983).
- Hamta, Fatemi Ghomi, Jolai, & Bahalke (2011) propose an bi-criteria extension of SALBP-2, which they call *Flexible task time ALB* (*FTALBP*, $\alpha_6 = pa$). Here, the processing of tasks can be accelerated in a limited manner, i.e., the task times can be

reduced by utilizing better equipment, which increases costs. Thus, reduced task times result in shorter cycle time but come at a higher cost. Hence, the objective function minimizes a weighted sum of cycle time and acceleration costs.

4.10. Objective functions

Early ALB research mainly considers problems with single capacity- or time-related objectives such as minimizing cycle time c or number of stations m. Recently, the following alternative *single objectives* have been focused:

- Originally introduced decades ago (e.g., Moodie & Young, 1965; Rachamadugu & Talbot, 1991), different objectives for *smoothing the workloads* ($\gamma = SSL^{line}$) of the stations have been used as a single (or primary) objective only recently. For instance, quite a few versions of SALBP (*SALBP-SX* for short) with fixed m and c that minimize some smoothness index SX (e.g., the variance of station times) have been considered (e.g., Azizoğlu & İmat, 2018; Hazır, Agi, & Guérin, 2020; 2021; Walter & Schulze, 2021; Walter, Schulze, & Scholl, 2021).
- In connection with different GALBP extensions, such as equipment selection, automated stations, and varying worker skills, a *cost function* ($\gamma = Co$) is minimized. Depending on the actual problem setting, various cost types (fixed installation and investment cost, variable operating cost as well as fixed and variable wage costs) have to be included (see Sections 4.2, 4.3, 4.9). For a recent review on cost- and profit-related ALB problems, we refer to Hazır et al. (2015).

Plenty of recent papers model ALB problems with *multiple objectives* ($\gamma = \text{MO}(*)$). These approaches differ in the way the objectives are aggregated:

- *Weighted sum* of objective functions: A quite common MO-approach is to construct an overall objective function by summing up single objectives. The objectives are given individual weights (often connected with a normalization of values to [0,1]), which enable the decision maker to define the relative importance of goals. A weighted sum aggregation is often applied to connect classical capacity- or time-oriented goals, such as the minimization of c or m, and goals focusing on novel aspects such as workload balancing, ergonomics, or further cost component (e.g., Hamta et al., 2013; Nearchou, 2008; Otto & Scholl, 2011). In other cases, deviations from (ideal) target values are considered in a goal programming approach (e.g., Choi, 2009).
- *Lexicographic ordering* of objectives: A further straightforward and widely-applied approach is to define a primary goal and to balance the line according to this objective function. Only if multiple solutions are optimal, a secondary objective function becomes relevant as a "tie breaker". The above mentioned smoothness index, for instance, is often applied as a secondary goal with "minimize m" or "minimize c" as a primary objective (e.g., Eswaramoorthi, Kathiresan, Jayasudhan, Prasad, & Mohanram, 2012; Mozdgir, Mahdavi, Badeleh, & Solimanpur, 2013). The so-called Lexicographic Bottleneck ALB (LB-ALBP) of Pastor (2011) considers a cascade of goals, which first minimizes the station time of the largest station. In a second step, this maximal time is fixed and the second largest station time is minimized. This is repeated until the station times of $m-1$ stations are fixed in this way. Note that a lexicographic ordering can be modeled via a weighted sum provided that the weights of subordinated objectives are set to sufficiently small values.
- *Pareto-optimality*: If multiple objectives are not combined to construct an overall objective function, the optimization seeks for all (or a subset of) the Pareto-optimal solutions. In this way, the minimization of c is, for instance, considered together with workload balancing (Nearchou, 2011), energy consumption (Nilakantan, Li, Tang, & Nielsen, 2017a), and costs (Yoosefelahi et al., 2012).

We can conclude that a huge variety of new ALB problems and models has been introduced in the last fifteen years. A complete overview on the published papers, as well as their considered problem settings and contributions is provided in the appendix.

5. Algorithms

In the light of the large number of novel problem settings and variations summarized in the preceding sections, it is neither possible nor helpful to give an overview of all applied solution methods on a detail level. Instead, we summarize and classify remarkable novel developments of solution principles and methodologies here. The details on the methodical contributions of all classified articles are given in the Tables of the Online-Appendix. Note that the abbreviations used there to specify the solution methods are also referred to in the following sections. Since almost all ALB problems are NP-hard (see Álvarez-Miranda & Pereira, 2019), it is particularly important to have powerful solution principles on hand, which are applicable to a broader range of problem settings.

5.1. Models and application of default solvers (M)

In the recent years, default solvers for mixed-integer programming have made remarkable progress. Hence, solving ALBPs – even with real-world dimensions – by applying such a solver more and more comes within reach. Therefore, it is a fruitful task for future research to systematically compare the traditional MIP formulations for SALBP, which originally stem from the 1960s (Bowman, 1960; White, 1961), with novel MIP formulations that have been suggested in the recent years. Especially, Pastor & Ferrer (2009), Ho & Emrouznejad (2009), Pastor, Garcia-Villoria, & Corominas (2011), Esmaeilbeigi, Naderi, & Charkhgard (2015), and Ritt & Costa (2018) were successful for different versions of SALBP, such that the solution time of default solvers is considerably reduced.

Also for more general ALBPs, considerable progress in mathematical programming approaches has been made. For instance, the following problem versions have successfully been tackled: GALBPS (Esmaeilbeigi et al., 2016), ALWABP (Fleszar, 2017), TSALBP with time, space and ergonomic constraints (Bautista et al., 2016), UALBP (Fattahi & Turkay, 2015). Furthermore, default solvers for constraint programming approaches have successfully been applied to some ALBPs (e.g., Bukchin & Raviv, 2018; Çil & Kizilay, 2020) and should thus be included into a systematic performance test.

5.2. Exact solution procedures

Between the 1950s and the 1990s, a lot of exact procedures for different versions of SALBP have been developed, based on dynamic programming (e.g., Held, Karp, & Shareshian, 1963; Jackson, 1956) or branch-and-bound. Noteworthy competitors of the latter class are FABLE of Johnson (1988), EUREKA of Hoffmann (1992), SALOME for SALBP-2 of Klein & Scholl (1996) and for SALBP-1 of Scholl & Klein (1997), as well as the procedures of Sprecher (1999) and Vilà & Pereira (2013). The well-established exact solution methods contain different elaborate enumeration schemes, bounding procedures, and dominance rules. They are able to solve small- to medium-sized problem instances to optimality but still struggle with very large instances.

A remarkable new branch-and-bound procedure, called *branch, bound, and remember* (*BBR*), has been introduced by Sewell & Jacobson (2012) as well as Morrison, Sewell, & Jacobson (2014) for

SALBP-1. It considerably outperforms previous exact methods and clearly increases the problem sizes that can successfully be solved within reasonable computational time. In contrast to former approaches like SALOME, which fail to solve the most challenging instances of the classical benchmark data set of Scholl (1993) in reasonable time, BBR solves all instances with an average computational time of about half a second. Concerning the novel benchmark data set of Otto, Otto, & Scholl (2013), BBR solves all small and medium-sized instances, all but a few of the large-sized ones, and two-thirds of the very large-sized ones in an hour of computational time each. BBR consists of three phases:

1. The well-known heuristic of Hoffmann (1963) is applied in a slightly modified manner to quickly determine a first feasible solution.
2. A station-oriented branching scheme is combined with a so-called cyclic-best-first search (CBFS). Briefly, this search strategy repeatedly follows the most promising path from the root to leaf nodes where feasible solutions can be found, if not fathomed before. In this way, the search process is much more flexible compared to a depth-first search, which is used by most predecessors. CBFS stores all considered partial solutions in a hash table in order to avoid double handling (tree dominance rule). It is combined with further dominance rules (e.g., Jackson dominance rule and maximal load rule, see Scholl & Becker, 2006) and classical lower bounds. As a further, powerful lower bound, the bin packing-relaxation of SALBP-1 is solved (again by another BBR algorithm). Often, the minimal number of bins (or a lower bound on this value) is a very tight bound for SALBP-1. Finally, BBR includes an intelligent decision rule for selecting the planning direction, which is a well-known key factor for a fast solution of SALBP-1 (Scholl & Klein, 1997).
3. In step (2), the maximum number of followers (station loads) per node is restricted, such that an optimal solution has not been found or optimality of the incumbent solution could not be proven yet. In this case, a breadth-first search is applied to form and traverse a new tree to finally find and prove the optimal solution.

An in-depth performance analysis of BBR and its individual components is provided by Li, Kucukkoc, & Tang (2019). BBR is, for instance, adapted to SALBP-2 by Li, Kucukkoc, & Tang (2021c), to TALBP by Li, Kucukkoc, & Zhang (2020), to UALBP by Li, Kucukkoc, & Zhang (2018c), to RALBP by Borba et al. (2018), and to a robust ALB by Pereira & Álvarez-Miranda (2018).

In addition to the mentioned enumerative procedures, *decomposition approaches* more and more become serious competitors for custom-made exact (and heuristic) solution procedures due to the recent progress of default solvers. Promising approaches based on Benders' decomposition (BD) are provided for a robust ALBP (Hazır & Dolgui, 2013; 2015), GALBPS (Akpinar et al., 2017; Zohali et al., 2021), MMALBP (Michels, Lopes, & Magatão, 2020; Michels et al., 2019), and mixed-model TALBP (Huang et al., 2021).

5.3. Construction heuristics

Feasible ALB solutions are required as a starting point for local search procedures or to obtain initial upper (lower) bounds for exact solution procedures in case of minimization (maximization) objectives. Traditionally, such solutions are determined by priority rule-based approaches or by some partial enumeration method like the well-known Hoffmann heuristic (Hoffmann, 1963).

More recently, *flexible procedures*, that are able to solve a broader range of ALBPs, have been proposed. The ant-colony based shortest-path approach (Avalanche) by Boysen & Fliedner (2008) and the enhanced multi-Hoffmann heuristic by Sternatz (2014) are able to solve different relevant GALBP extensions of SALBP even in a combined manner.

Another stream of research attempts to *improve priority-rule based procedures*, because they are well-suited to obtain initial solutions quickly (even for very large-sized instances) and can flexibly be adapted to other problem types. A lot of single priority rules, two basic planning schemes (task and station oriented), and three planning directions (forward, backward, bidirectional) are available with each rule having specific pros and cons (see Otto & Otto, 2014a; Scholl & Voß, 1996). Multi-pass procedures apply different rule-scheme combinations in order to combine their strengths, and additional random components diversify the resulting solution set. Both ideas are combined in the *greedy randomized adaptive search procedure (GRASP)*, a general strategy introduced by Feo & Resende (1995). GRASPs for different ALBPs are, e.g., developed and tested by Andrés et al. (2008), Martino & Pastor (2010), Scholl et al. (2013), Moreira & Costa (2013), Pastor, García-Villoria, Laguna, & Martí (2015), and Chica, Cordón, Damas, & Bautista (2010). Different principles and methods to design well-performing priority rules for different ALB problems and a comprehensive computational study are presented by Otto & Otto (2014a). A quite interesting approach attempts to identify well-performing composite rules: Martino & Pastor (2010) determine a composite rule for GALBPS with the rule weights being determined and fine-tuned by a nonlinear optimization technique (Nelder-Mead) which performs considerable better than single priority rules and other combined rules. Machine learning could be a fruitful approach to adapt the rule weights even more flexibly and to improve over these predecessors.

Naturally, all enumerative exact procedures can be restricted concerning the computational time and/or the search space to obtain heuristics. Among others, Blum & Miralles (2011), Çil et al. (2017), and Li, Janardhanan, & Rahman (2021b), Li et al. (2021c) propose *beam search* procedures for different ALB problems. They restrict the size of the enumeration tree by defining a beam width, i.e., maximal number of transitions from a state at stage k of the tree to the next stage. In the Beam-ACO of Blum (2008), the selection of accepted transitions is done randomly by an ant colony optimization (ACO) approach. *Bounded DP* is a rather similar approach. It is based on a station-oriented DP enumeration scheme, which is additionally bounded by the use of lower bounds (Bautista & Pereira, 2009; Pape, 2015).

5.4. Meta-heuristics

Nowadays, operations research offers a rich arsenal of meta-heuristics that are applied to determine near-optimal solutions wherever possible. In ALB research too, all kinds of meta-heuristics are applied to different problem types. Our summary covers the different types of meta-heuristics that have mainly been applied to ALB problems in the last fifteen years and only can provide a selection of references. Again a detailed breakdown is provided in the column "Contribution" of the Tables in the Online-Appendix.

Local-search-based meta-heuristics: These classical meta-heuristics interact with and steer local improvement (local search) procedures in order to overcome the problem of getting stuck in local optima.

- Though there exist principle difficulties in applying *tabu search (TS)* to ALB (Scholl & Voß, 1996), TS procedures are proposed for different basic and extended ALBPs like: SALBP-1 (Lapierre, Ruiz, & Soriano, 2006; Pape, 2015), parallel and/or two-sided lines (Özcan, 2018; Özcan et al., 2010; Özcan & Toklu, 2009b), LB-ALBP (Pastor et al., 2015), re-balancing (Girit & Azizoğlu, 2021), and ALWABP (Moreira & Costa, 2013).

- *Simulated annealing (SA)* schemes are, for instance, proposed for: RALBP (Aghajani et al., 2014; Li, Janardhanan, Nielsen, & Tang, 2018b), AWALBP (García-Villoria, Corominas, & Pastor, 2015), UALBP with resource dependent task times (Jayaswal & Agarwal, 2014), and mixed-model UALBP (Hamzadayi & Yildiz, 2013).
- *Variable neighborhood search (VNS)* is, for instance applied to: TALBP (Lei & Guo, 2016), ALWABP (Polat, Kalayci, Mutlu, & Gupta, 2015), and mixed-model balancing (Sadeghi, Rebelo, & Ferreira, 2018).

Population-based and nature-inspired meta-heuristics: A vast number of papers propose diverse methods that are based on populations of solutions and mostly are inspired by (strong or loose) analogies to nature. In some cases, the choice of the procedures seems somewhat arbitrary as it is not explained why just this and not another scheme is selected for the considered problem.

- *Genetic algorithms (GA)* and other *evolutionary algorithms (EA)* are, e.g., proposed for: different versions of SALBP (Nearchou, 2008; Yu & Yin, 2010; Zhang, 2019), TSALBP (Chica, Cordón, & Damas, 2011), RALBP (Dalle Mura & Dini, 2019), single and mixed-model UALBP (Hwang, Katayama, & Gen, 2008), MMALBP (Kellegöz & Toklu, 2015), TALBP (Kim, Song, & Kim, 2009), ALWABP (Moreira, Ritt, Costa, & Chaves, 2012), GALBPS (Yolmeh & Kianfar, 2012), and FALBP (Zacharia & Nearchou, 2012).
- Among the numerous papers that propose heuristics that are *inspired by nature or other real-world phenomena*, we select some examples that show the diversity of the approaches and the considered problems:
  - *Ant colony optimization and other ant systems (ANT)*: SALBP-2 (Zheng, Li, Li, & Tang, 2013), TSALBP (Bautista & Pereira, 2007; Chica et al., 2010), UALBP (Sabuncuoglu, Erel, & Alp, 2009), different types of TALBP (Kucukkoc & Zhang, 2014a; Simaria & Vilarinho, 2009).
  - *Particle swarm optimization (PSO)*: different versions of SALBP (Dou, Li, & Su, 2013; Nearchou, 2011; Petropoulos & Nearchou, 2011), stochastic UALBP and/or TALBP (Aydogan, Delice, Ozcan, Gencer, & Bali, 2019; Chiang, Urban, & Luo, 2016; Delice, Kızılkaya Aydoğan, & Özcan, 2016), RALBP (Nilakantan et al., 2015), MMALBP with resources (Şahin & Kellegöz, 2019).
  - *Imperialist competitive algorithm (ICA)*: deterministic and stochastic UALBP (Bagher, Zandieh, & Farsijani, 2011; Nourmohammadi, Zandieh, & Tavakkoli-Moghaddam, 2013), mixed-model line with parallelism (Li, Zhang, Tian, Shao, & Li, 2018a; Ramezanian & Ezzatpanah, 2015), TALBP (Wang, Guan, Li, Zhang, & Chen, 2014).
  - *Bee colony algorithm (BC)*: Mixed-model two-sided RALBP (Çil & Kizilay, 2020), RALBP with setup times (Li, Janardhanan, & Ponnambalam, 2021a), parallel TALBP with walking times (Tapkan, Özbakır, & Baykasoğlu, 2016), U-shaped RALBP (Zhang et al., 2019).
  - Like in other fields of optimization, there are numerous other nature-inspired approaches like *Cuckoo Search (CS)* and *Honey Bee Mating Optimization (HBMO)*, which we do not mention here explicitly as their value for the development of meta-heuristics is questionable (Sörensen, 2015).

Hybrid meta-heuristics: Recently, there is another emerging trend to combine different meta-heuristics to form hybrid meta-strategies such as the following:

- A hybrid of VNS with SA is proposed by Roshani, Paolucci, Giglio, & Tonelli (2021) for MMALBP and a hybrid with PSO is developed by Hamta et al. (2013) for an multi-objective ALBP with setup-times and learning effects.
- Akpinar et al. (2013) consider a hybrid of ACO and GA for a mixed-model ALBP with setup times and other extensions. As mentioned before, Beam-ACO of Blum (2008) combines a beam search with an ant colony evaluation and selection mechanism.
- An emerging trend in heuristic optimization are so-called *matheuristics (MH)* (e.g., see Maniezzo, Stützle, & Voß, 2009). They combine the high performance of default MIP solvers with other heuristic meta-strategies providing a large toolbox of solution approaches. First applications to ALB problems are: an MH-based VNS for MMALBP (Álvarez-Miranda, Chace, & Pereira, 2021) and MH-based approaches for different versions of AWALBP (Calleja, Corominas, García-Villoria, & Pastor, 2014; 2016; García-Villoria et al., 2018).

It is our impression that the ALB research community should not progress and combine each problem version of ALB with any (new, established, or combined) heuristic search scheme. Instead, a rather systematic evaluation, which general solution scheme works good for which problem classes seems more meaningful in this domain.

6. Future research agenda and conclusions

In spite of the manifold research efforts reported in the previous sections, there is still a gap to be bridged between ALB research and the configuration planning of real-world assembly lines. There is thus enough work left for future generations of operations researchers in the ALB domain, and we see special demand in the following three areas:

- Combined models that integrate ALB with related decision tasks,
- ALBPs customized for the different life cycle phases of an assembly line,
- ALBPs for smart factories of the industry 4.0 era.

6.1. Combined models

As we have seen, deciding on the division of labor with the help of an ALBP is already challenging enough. However, improvements in computing power, the impressive development of default MIP solvers, and methodological progress in fields such as decomposition as well as meta- and matheuristics more and more allow to fulfill the demand of practitioners for combined models. The configuration of an assembly line is heavily interdependent with related fields, such as part logistics or personnel planning. Subsequently solved separate planning models, related to each other in a decision hierarchy, neglect these interdependencies and risk worse total planning results. However, combined models still come for a price, namely complex models but also more involved organizational decision processes. Therefore, they are not automatically preferable. Instead, identifying suitable separations among related decision tasks that can nonetheless be decomposed without too much plan deterioration are also a most valuable research result. To identify these decision tasks, however, a combined model is nonetheless inevitable, at least in order to quantify the achievable and to benchmark alternative decision processes.

Traditionally, combined models in the ALB domain mainly unify balancing and sequencing. Recent contributions of the past decade are, for instance, provided by Kucukkoc & Zhang (2014a,b) and Lopes, Michels, Sikora, Molina, & Magatão (2018), Lopes, Sikora, Michels, & Magatao (2020b). Mixed-model assembly lines also give rise to a short-term sequencing problem, which decides on the production sequence of workpieces launched down the line. Since different products demand different assembly times in the stations, sequence planning aims to avoid a work overload of assembly workers (for a survey, see Boysen et al., 2009b). Certainly, balancing and sequencing are heavily interdependent. Specifically, the former determines the workload at the stations and thus provides the basic input data for the sequencing problem. Vice versa, the result of the sequencing problem, namely the amount of work overload, is also an important performance measure for a line balance. However, the main obstacle of solving both problems jointly in the same model are their different time frames. When balancing a line, typically, the exact production program is yet unknown. Hence, some researchers suggest to integrate stochastic sequence problems with line balancing (e.g., see Özcan et al., 2011). This results in very complex decision tasks, so that it may be preferable to rather roughly anticipate the impact of the line balance on sequencing with a workload smoothing approach (e.g., see Emde, Boysen, & Scholl, 2010). The latter is a well-researched approach that has shown successful in many applications. Consequently, we suggest to focus future research rather on the integration of alternative interdependent decision tasks:

- *Part logistics:* Depending on the exact positioning of the bins, racks, and containers for parts in a workstation, assembly work contains a considerable amount of unproductive walking for fetching parts (Sedding, 2020). If the line balance assigns multiple tasks to a single station that demand large part containers and compete for a similar positioning regarding a worker's optimal walking triangle along the moving conveyor, then unproductive walking can increase considerably. Therefore, deciding between a more costly, yet less space consuming small-lot supply, e.g., via a close-by parts supermarket, or bulk delivery by forklift (for more details, see Boysen et al., 2015) as well as the positioning of part containers along the line is heavily interdependent with ALB. First combined models integrating part feeding are, for instance, provided by Sternatz (2015), Battini, Calzavara, Otto, & Sgarbossa (2016a), Nourmohammadi & Eskandari (2017), and Nourmohammadi, Eskandari, & Fathi (2018). However, including the large variety of real-world part-feeding policies still requires further research.
- *Quality:* The line balance determines the stations' work contents, so that it also decides on the complexity and amount of work to be handled by each worker. There are plenty of empirical findings that complexity and lack of idle time increase the occurrence of quality defects in assembly work (e.g. Eklund, 1995; Fast-Berglund, Fässberg, Hellman, Davidsson, & Stahre, 2013). Thus, deriving suitable work contents for workers that fairly distribute stress, complexities, and idle time in order to reduce quality defects are an important field for future research. This not only requires an adaption of ALB models but also interdisciplinary work in order to develop proactive measures to evaluate work contents concerning these qualitative factors. Also, where to place inspection and rework stations to detect and repair quality defects, respectively, are important decision tasks that impact the line balance. Since some assembly tasks depend on predecessor tasks, it can be advantageous to remove neuralgic quality defects within the line and not just at the end in extra rework areas (Boysen, Scholl, & Wopperer, 2012). Then, dependent tasks can be completed in subsequent stations as planned and do not add to the amount of rework to be accomplished after assembly.
- *Personnel:* As already elaborated in Section 4.2, integrating personnel planning and worker assignment into line balancing is already a current trend to be witnessed in the past decade. For instance, previous research considers that workers may vary in their processing speeds (e.g. Miralles et al., 2007; Miralles et al., 2008) and qualification for specific tasks (e.g. Sungur & Yavuz, 2015). Other human factors such as learning effects over time are among the classics of the ALB community for several decades (for some recent contributions see, e.g., Otto & Otto, 2014b; Toksarı, İşleyen, Güner, & Baykoç, 2010). Even in this established area, however, new trends or current challenges demand new combined ALB models. For instance, integrating minimum distances among human workers during all the positional changes within a production cycle is an important aspect in times of a pandemic to stop virus spread.
- *Model assignment:* Another important decision that is heavily related with line balancing is the decision which models should be produced on a mixed-model assembly line. The main trade-off this decision faces is between flexibility and costs (see Jordan & Graves, 1995). The more assembly lines are equipped to produce a specific product, the more flexibly production quantities can be adapted to unforeseen shifts in customer demands. However, equipping a line for a specific product produces flexibility costs, such as additional worker qualification and product-specific machinery. There is already some research on balancing parallel lines, and there is also a special survey paper dedicated to this area (Lusa, 2008). However, most of these predecessors only focus on one specific aspect, e.g., the usage of workers splitting their work content among two adjacent lines in the same factory (e.g. Gökçen et al., 2006; Scholl & Boysen, 2009). Today's car manufacturers, however, offer more and more models and variants, so that a systematic evaluation of the pros and cons of different partitions of the product program among assembly lines, along with the resulting equipment decisions and the impact on the line balances remain very challenging tasks for future research.

6.2. ALBPs for different phases of the life cycle

Future research should address varying ALBP models that are required for the different phases of the life cycle of an assembly line and the products it manufactures. Existing research mainly focuses on the initial line balance, where all tasks are to be placed for the first time and can be arbitrarily moved along the line. However, it much more often occurs that an existing assembly line is to be re-balanced (Boysen et al., 2008), and already several months or even years before the start of production the very first line balances need to be derived. We thus suggest that future research not exclusively focuses on initial line balancing. Instead, there are, for instance, the following triggers for ALB that require the solution of substantially different ALBPs:

- *Several months before start of production:* A modern assembly line with all its conveyor technology, machinery, and manifold relations to other departments is a complex entity by itself. Hence, several months up to two years before the start of production, the first line balances have to be determined. The result is, for instance, required to dimension and position racks, supermarkets, and automated storage and retrieval systems for part logistics (e.g. Boysen et al., 2015) and to layout conveyor paths and wiring that strongly depend on the positions of important machinery, such as the wedding station where engine and chassis are united. Naturally, during this time the product to be produced is still in the design phase by itself and thus ALB faces a lot of uncertainties. In this phase, especially the data retrieval is a huge challenge. Since the product is not finally specified, precedence relations are also widely uncertain. Developing suitable data retrieval methods that, e.g., take over (most likely) precedence relations from predecessor or related products is an important task for future research. Moreover, line balancing should also be allowed to influence product development. If a specific precedence constraint severely reduces the flexibility of ALB and, for instance, leads to more stations, it may still be possible to remove this constraint by an adaption of the product design. More research on how such a feed-back loop can successfully be established is required.

- *Change of model mix:* After first time installation, assembly lines are frequently re-balanced. Such a re-balance might be triggered by a considerable demand shift or an altered model mix of products to be assembled. Since workers with given qualification and some immobile machinery already exist, the flexibility to move tasks to other stations is restricted. This may help to decompose the line into sections that can be re-balanced separately. Since real-world automotive assembly lines with their hundreds of stations are still a huge algorithmic challenge, future research should evaluate where a line can be separated at a justifiable loss of planning quality.
- *Change of cycle time:* If customer demands alter only moderately, it may be sufficient to adapt the cycle time. In this case, we generally face a SALBP-1 setting, where we aim to minimize the number of stations (and the workforce to staff them) for the new given cycle time. Moreover, it seems advantageous to alter the work content for the assembly workers least as possible (Sancı & Azizoğlu, 2017). Otherwise, additional qualification costs and losses of previous learning effects might occur.

We can conclude that there is a need for substantially different ALBPs in different phases of the life cycle. A systematic evaluation of each phase and all its basic requirements, including the ALBPs to be solved and suitable data retrieval methods, is still a challenging task for future research.

6.3. ALBPs for smart factories

Ongoing technological developments such as digitization, autonomous robots, and Internet-of-Things (IoT) technology influence not only the appearance and capabilities of assembly lines, but they also require an adaption of the ALB problems to be solved. Two exemplary settings of future smart factories that demand novel models and solution approaches are briefly described in the following:

- *Flexible shift models:* Many car producers, especially in industrialized countries, face an aging workforce and are thus anxious to establish more flexible work shift models, even in final assembly. The established modus operandi is a fixed shift scheme, where the complete workforce starts and ends each work shift, e.g., of seven hours, unanimously. In a more flexible environment, there is no unique start point but workers can start work at different (yet potentially predefined) points in time. Workers identify themselves directly on the shop floor and are assigned their current workload on the fly. This requires a re-balancing under real-time restrictions that withdraws work content from the established workforce and reassigns it to the new workers. The resulting re-balancing problem under real-time constraints has to consider the qualification of workers, should only assign nearby tasks to each new worker to reduce the resulting walking distances. Furthermore, it should not reassign too much workload among the established workforce in order to ensure a smooth adaption process. In this way, the cycle time can flexibly be readjusted during a shift, depending on the current workforce.
- *Flexible assembly paths:* If the workpieces are no longer transported by a conveyor but each by its own autonomous transport robot, the traditional assembly line concept dating back to Henry Ford is altered in two fundamental ways: First, the workpieces need not follow all the same linear path through the assembly stations. Instead, they can visit the stations in a more flexible manner, and robots can adapt their paths according the stations' current occupation, as long as no precedence constraints are violated. Second, since workpieces are no longer continuously conveyed through the stations, there is no need for a strict cycle time that is identical in all stations. In spite of these severe alterations, however, a balancing of the workload among stations is still inevitable, although not necessarily in every production cycle. But as soon as one station continuously receives more workload over a longer period of time, it becomes the unique bottleneck and leads to inefficiencies. Hence, while the general ALB structure is still relevant, adaptions are required to consider alternative paths through the stations, which are arranged in a non-linear manner and variable transportation times among stations are to be considered. A first approach in this direction is provided by Hottenrott & Grunow (2019), but there is certainly more research required until the new flexibility offered by smart factories can be fully utilized.

6.4. Concluding remarks

We want to end with a few personal remarks on lessons-learned from an implementation project of a tailor-made line balancing software at a major German automotive producer. Finally, the project was successful but it was also a huge effort, and we accompanied it with scientific advice for more than a decade.

- First, we conclude that identifying suitable ALBPs and deriving efficient solution algorithms is an important, yet by far not the most time-consuming task. Gaining reliable data (foremost the precedence graph) especially during early life-cycle phases of new car models and keeping it updated with economically justifiable effort is a tremendous task not only for software development but especially for the management to overcome all organizational burdens. Moreover, designing a suitable graphical user interface that displays the results in a well-arranged manner and allows planners to conveniently adapt the planning result in order to integrate special human knowledge that is hardly codable for an algorithm is of utmost importance for user acceptance.
- The latter leads us to our second point. A line balance gained by an algorithm cannot be expected to represent the final solution that is actually realized. An ALB software only delivers decision support and is a starting point for the inevitable adaptations of human planners. In line with this finding, it is our experience that it is often not the quality of the solution that is considerably improved by an ALB optimization procedure. Experienced planners typically reach the solution quality of an optimization procedure, it only takes them several weeks of manual readjustments. Hence, it is our experience that the main advantage of an optimization-based ALB software is a dramatic reduction of planning time and effort. However, given the steady decrease of today's product life cycles, this is an important competitive advantage.
- Finally, it is our experience that once a suitable decision support tool for line balancing exists and has proven its usefulness, further organizational units in charge of related decision tasks such as part logistics, ergonomics, and personnel planning formulate their demands for more and more extensions and additional optimization support. Hence, the line balancing software can become an indispensable central planning instrument of today's assembly processes.

To conclude, there are still many important and interesting future research tasks to be investigated. Hence, we are positive that in another fifteen years a further survey paper that has to cope with a comparable number of new scientific contributions can be written.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.ejor.2021.11.043.

References

Aghajani, M., Ghodsi, R., & Javadi, B. (2014). Balancing of robotic mixed-model two-sided assembly line with robot setup times. *The International Journal of Advanced Manufacturing Technology, 74*(5–8), 1005–1016.

Ağpak, K., & Gökçen, H. (2007). A chance-constrained approach to stochastic line balancing problem. *European Journal of Operational Research, 180*(3), 1098–1115.

Ağpak, K., Yegül, M. F., & Gökçen, H. (2012). Two-sided U-type assembly line balancing problem. *International Journal of Production Research, 50*(18), 5035–5047.

Aguilar, H., García-Villoria, A., & Pastor, R. (2020). A survey of the parallel assembly lines balancing problem. *Computers & Operations Research, 124*, 105061.

Akpinar, S., Elmi, A., & Bektaş, T. (2017). Combinatorial benders cuts for assembly line balancing problems with setups. *European Journal of Operational Research, 259*(2), 527–537.

Akpinar, S., Mirac Bayhan, G., & Baykasoğlu, A. (2013). Hybridizing ant colony optimization via genetic algorithm for mixed-model assembly line balancing problem with sequence dependent setup times between tasks. *Applied Soft Computing, 13*(1), 574–589.

Akyol, S. D., & Baykasoğlu, A. (2019). ErgoALWABP a multiple-rule based constructive randomized search algorithm for solving assembly line worker assignment and balancing problem under ergonomic risk factors. *Journal of Intelligent Manufacturing, 30*(1), 291–302.

Alavidoost, M. H., Babazadeh, H., & Sayyari, S. T. (2016). An interactive fuzzy programming approach for bi-objective straight and U-shaped assembly line balancing problem. *Applied Soft Computing, 40*, 221–235.

Alavidoost, M. H., Tarimoradi, M., & Zarandi, M. F. (2015). Fuzzy adaptive genetic algorithm for multi-objective assembly line balancing problems. *Applied Soft Computing, 34*, 655–677.

Altemeier, S. (2009). *Kostenoptimale Kapazitätsabstimmung in einer getakteten Variantenfließlinie*. Dissertation, University of Paderborn. https://d-nb.info/994983549/34

Altemeier, S., Brodkorb, D., & Dangelmaier, W. (2010). A top-down approach for an automatic precedence graph construction under the influence of high product variety. In B. Vallespir, & T. Alix (Eds.), *Advances in production management systems. New challenges, new approaches* (pp. 73–80). Berlin: Springer.

Álvarez-Miranda, E., Chace, S., & Pereira, J. (2021). Assembly line balancing with parallel workstations. *International Journal of Production Research, 59*(21), 6486–6506.

Álvarez-Miranda, E., & Pereira, J. (2019). On the complexity of assembly line balancing problems. *Computers & Operations Research, 108*, 182–186.

Amen, M. (2000). An exact method for cost-oriented assembly line balancing. *International Journal of Production Economics, 64*(1–3), 187–195.

Amen, M. (2001). Heuristic methods for cost-oriented assembly line balancing: A comparison on solution quality and computing time. *International Journal of Production Economics, 69*(3), 255–264.

Andrés, C., Miralles, C., & Pastor, R. (2008). Balancing and scheduling tasks in assembly lines with sequence-dependent setup times. *European Journal of Operational Research, 187*(3), 1212–1223.

Antani, K. R., Pearce, B., Kurz, M. E., Mears, L., Funk, K., & Mayorga, M. E. (2013). Manual precedence mapping and application of a novel precedence relationship learning technique to real-world automotive assembly line balancing. In *Volume 2: Systems; Micro and nano technologies; Sustainable manufacturing*. ASME. https://doi.org/10.1115/MSEC2013-1235.

Antani, K. R., Pearce, B., Mears, L., Renu, R., Kurz, M. E., & Schulte, J. (2014). Application of system learning to precedence graph generation for assembly line balancing. In *Volume 1: Materials; Micro and nano technologies; Properties, applications and systems; Sustainable manufacturing*. ASME. https://doi.org/10.1115/MSEC2014-3906.

Aydogan, E. K., Delice, Y., Ozcan, U., Gencer, C., & Bali, O. (2019). Balancing stochastic U-lines using particle swarm optimization. *Journal of Intelligent Manufacturing, 30*(1), 97–111.

Azizoğlu, M., & İmat, S. (2018). Workload smoothing in simple assembly line balancing. *Computers & Operations Research, 89*, 51–57.

Babazadeh, H., & Javadian, N. (2019). A novel meta-heuristic approach to solve fuzzy multi-objective straight and U-shaped assembly line balancing problems. *Soft Computing, 23*(17), 8217–8245.

Bagher, M., Zandieh, M., & Farsijani, H. (2011). Balancing of stochastic U-type assembly lines: An imperialist competitive algorithm. *The International Journal of Advanced Manufacturing Technology, 54*(1–4), 271–285.

Bard, J. F. (1989). Assembly line balancing with parallel workstations and dead time. *International Journal of Production Research, 27*(6), 1005–1018.

Bartholdi, J. (1993). Balancing two-sided assembly lines: A case study. *International Journal of Production Research, 31*(10), 2447–2461.

Battaïa, O., & Dolgui, A. (2013). A taxonomy of line balancing problems and their solution approaches. *International Journal of Production Economics, 142*(2), 259–277.

Battini, D., Boysen, N., & Emde, S. (2013). Just-in-time supermarkets for part supply in the automobile industry. *Journal of Management Control, 24*(2), 209–217.

Battini, D., Calzavara, M., Otto, A., & Sgarbossa, F. (2016a). The integrated assembly line balancing and parts feeding problem with ergonomics considerations. *IFAC-PapersOnLine, 49*(12), 191–196.

Battini, D., Calzavara, M., Otto, A., & Sgarbossa, F. (2017). Preventing ergonomic risks with integrated planning on assembly line balancing and parts feeding. *International Journal of Production Research, 55*(24), 7452–7472.

Battini, D., Delorme, X., Dolgui, A., Persona, A., & Sgarbossa, F. (2016b). Ergonomics in assembly line balancing based on energy expenditure: A multi-objective model. *International Journal of Production Research, 54*(3), 824–845.

Bautista, J., Batalla-García, C., & Alfaro-Pozo, R. (2016). Models for assembly line balancing by temporal, spatial and ergonomic risk attributes. *European Journal of Operational Research, 251*(3), 814–829.

Bautista, J., & Pereira, J. (2007). Ant algorithms for a time and space constrained assembly line balancing problem. *European Journal of Operational Research, 177*(3), 2016–2032.

Bautista, J., & Pereira, J. (2009). A dynamic programming based heuristic for the assembly line balancing problem. *European Journal of Operational Research, 194*(3), 787–794.

Baybars, I. (1986). A survey of exact algorithms for the simple assembly line balancing problem. *Management Science, 32*(8), 909–932.

Becker, C., & Scholl, A. (2006). A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research, 168*(3), 694–715.

Becker, C., & Scholl, A. (2009). Balancing assembly lines with variable parallel workplaces: Problem definition and effective solution procedure. *European Journal of Operational Research, 199*(2), 359–374.

Blum, C. (2008). Beam-ACO for simple assembly line balancing. *INFORMS Journal on Computing, 20*(4), 618–627.

Blum, C., & Miralles, C. (2011). On solving the assembly line worker assignment and balancing problem via beam search. *Computers & Operations Research, 38*(1), 328–339.

Borba, L., Ritt, M., & Miralles, C. (2018). Exact and heuristic methods for solving the robotic assembly line balancing problem. *European Journal of Operational Research, 270*(1), 146–156.

Bortolini, M., Faccio, M., Gamberi, M., & Pilati, F. (2017). Multi-objective assembly line balancing considering component picking and ergonomic risk. *Computers & Industrial Engineering, 112*, 348–367.

Bowman, E. H. (1960). Assembly-line balancing by linear programming. *Operations Research, 8*(3), 385–389.

Boysen, N., Emde, S., Hoeck, M., & Kauderer, M. (2015). Part logistics in the automotive industry: Decision problems, literature review and research agenda. *European Journal of Operational Research, 242*(1), 107–120.

Boysen, N., & Fliedner, M. (2008). A versatile algorithm for assembly line balancing. *European Journal of Operational Research, 184*(1), 39–56.

Boysen, N., Fliedner, M., & Scholl, A. (2007). A classification of assembly line balancing problems. *European Journal of Operational Research, 183*(2), 674–693.

Boysen, N., Fliedner, M., & Scholl, A. (2008). Assembly line balancing: Which model to use when? *International Journal of Production Economics, 111*(2), 509–528.

Boysen, N., Fliedner, M., & Scholl, A. (2009a). Assembly line balancing: Joint precedence graphs under high product variety. *IIE Transactions, 41*(3), 183–193.

Boysen, N., Fliedner, M., & Scholl, A. (2009b). Sequencing mixed-model assembly lines: Survey, classification and model critique. *European Journal of Operational Research, 192*(2), 349–373.

Boysen, N., Scholl, A., & Wopperer, N. (2012). Resequencing of mixed-model assembly lines: Survey and research agenda. *European Journal of Operational Research, 216*(3), 594–604.

Bukchin, J., & Tzur, M. (2000). Design of flexible assembly line to minimize equipment cost. *IIE Transactions, 32*(7), 585–598.

Bukchin, Y., & Raviv, T. (2018). Constraint programming for solving various assembly line balancing problems. *Omega, 78*, 57–68.

Burns, L. D., & Daganzo, C. F. (1987). Assembly line job sequencing principles. *International Journal of Production Research, 25*(1), 71–99.

Buxey, G. (1974). Assembly line balancing with multiple stations. *Management Science, 20*(6), 1010–1021.

Buzacott, J. A. (1967). Prediction of the efficiency of production systems without internal storage. *International Journal of Production Research, 6*(3), 173–188.

Calleja, G., Corominas, A., García-Villoria, A., & Pastor, R. (2013). A MILP model for the accessibility windows assembly line balancing problem (AWALBP). *International Journal of Production Research, 51*(12), 3549–3560.

Calleja, G., Corominas, A., García-Villoria, A., & Pastor, R. (2014). Combining matheuristics and MILP to solve the accessibility windows assembly line balancing problem level 2 (AWALBP-L2). *Computers & Operations Research, 48*, 113–123.

Calleja, G., Corominas, A., García-Villoria, A., & Pastor, R. (2016). Hybrid metaheuristics for the accessibility windows assembly line balancing problem level 2 (AWALBP-L2). *European Journal of Operational Research, 250*(3), 760–772.

Capacho, L., & Pastor, R. (2008). Asalbp: The alternative subgraphs assembly line balancing problem. *International Journal of Production Research, 46*(13), 3503–3516.

Capacho, L., Pastor, R., Dolgui, A., & Guschinskaya, O. (2009). An evaluation of constructive heuristic methods for solving the alternative subgraphs assembly line balancing problem. *Journal of Heuristics, 15*(2), 109–132.

Caputo, F., Greco, A., Fera, M., & Macchiaroli, R. (2019). Digital twins to enhance the integration of ergonomics in the workplace design. *International Journal of Industrial Ergonomics, 71*, 20–31.

Chase, R. B. (1974). Survey of paced assembly lines. *Industrial Engineering, 6*(2), 14–18.

Chiang, W.-C., Urban, T. L., & Luo, C. (2016). Balancing stochastic two-sided assembly lines. *International Journal of Production Research, 54*(20), 6232–6250.

Chiang, W.-C., Urban, T. L., & Xu, X. (2011). A bi-objective metaheuristic approach to unpaced synchronous production line-balancing problems. *International Journal of Production Research, 50*(1), 293–306.

Chica, M., Bautista, J., Cordón, Ó., & Damas, S. (2016). A multiobjective model and evolutionary algorithms for robust time and space assembly line balancing under uncertain demand. *Omega, 58*, 55–68.

Chica, M., Cordón, Ó., & Damas, S. (2011). An advanced multiobjective genetic algorithm design for the time and space assembly line balancing problem. *Computers & Industrial Engineering, 61*(1), 103–117.

Chica, M., Cordón, Ó., Damas, S., & Bautista, J. (2010). Multiobjective constructive heuristics for the 1/3 variant of the time and space assembly line balancing problem: ACO and random greedy search. *Information Sciences, 180*(18), 3465–3487.

Chica, M., Cordón, Ó., Damas, S., & Bautista, J. (2013). A robustness information and visualization model for time and space assembly line balancing under uncertain demand. *International Journal of Production Economics, 145*(2), 761–772.

Choi, G. (2009). A goal programming mixed-model line balancing for processing time and physical workload. *Computers & Industrial Engineering, 57*(1), 395–400.

Chutima, P. (2020). A comprehensive review of robotic assembly line balancing problem. *Journal of Intelligent Manufacturing,*. https://doi.org/10.1007/s10845-020-01641-7.

Çil, Z. A., & Kizilay, D. (2020). Constraint programming model for multi-manned assembly line balancing problem. *Computers & Operations Research, 124*, 105069.

Çil, Z. A., Mete, S., Özceylan, E., & Ağpak, K. (2017). A beam search approach for solving type II robotic parallel assembly line balancing problem. *Applied Soft Computing, 61*, 129–138.

Dalle Mura, M., & Dini, G. (2019). Designing assembly lines with humans and collaborative robots: A genetic approach. *CIRP Annals - Manufacturing Technology, 68*(1), 1–4.

Delice, Y., Aydoğan, E. K., Özcan, U., & İlkay, M. S. (2017). Balancing two-sided U–type assembly lines using modified particle swarm optimization algorithm. *4OR, 15*(1), 37–66.

Delice, Y., Kızılkaya Aydoğan, E., & Özcan, U. (2016). Stochastic two-sided U-type assembly line balancing: A genetic algorithm approach. *International Journal of Production Research, 54*(11), 3429–3451.

Dimitriadis, S. 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.

Dobson, G., & Yano, C. A. (1994). Cyclic scheduling to minimize inventory in a batch flow line. *European Journal of Operational Research, 75*(2), 441–461.

Dolgui, A., Sgarbossa, F., & Simonetto, M. (2021). Design and management of assembly systems 4.0: systematic literature review and research agenda. *International Journal of Production Research (to appear)*.

Dou, J., Li, J., & Su, C. (2013). A novel feasible task sequence-oriented discrete particle swarm algorithm for simple assembly line balancing problem of type 1. *The International Journal of Advanced Manufacturing Technology, 69*(9-12), 2445–2457.

Eghtesadifard, M., Khalifeh, M., & Khorram, M. (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.

Eklund, J. A. (1995). Relationships between ergonomics and quality in assembly work. *Applied Ergonomics, 26*(1), 15–20.

Emde, S., Boysen, N., & Scholl, A. (2010). Balancing mixed-model assembly lines: A computational evaluation of objectives to smoothen workload. *International Journal of Production Research, 48*(11), 3173–3191.

Esmaeilbeigi, R., Naderi, B., & Charkhgard, P. (2015). The type E simple assembly line balancing problem: A mixed integer linear programming formulation. *Computers & Operations Research, 64*, 168–177.

Esmaeilbeigi, R., Naderi, B., & Charkhgard, P. (2016). New formulations for the setup assembly line balancing and scheduling problem. *OR Spectrum, 38*(2), 493–518.

Eswaramoorthi, M., Kathiresan, G. R., Jayasudhan, T. J., Prasad, P., & Mohanram, P. V. (2012). Flow index based line balancing: A tool to improve the leanness of assembly line design. *International Journal of Production Research, 50*(12), 3345–3358.

Faber, M., Przybysz, P., Latos, B. A., Mertens, A., Brandl, C., Finsterbusch, T., … Nitsch, V. (2019). Empirical validation of the time accuracy of the novel process language human work design (MTM-HWD®). *Production & Manufacturing Research, 7*(1), 350–363.

Fast-Berglund, Å., Fässberg, T., Hellman, F., Davidsson, A., & Stahre, J. (2013). Relations between complexity, quality and cognitive automation in mixed-model assembly. *Journal of Manufacturing Systems, 32*(3), 449–455.

Fathi, M., & Ghobakhloo, M. (2014). A technical comment on "a review on assembly sequence planning and assembly line balancing optimisation using soft computing approaches". *The International Journal of Advanced Manufacturing Technology, 71*(9-12), 2033–2042.

Fattahi, A., & Turkay, M. (2015). On the MILP model for the U-shaped assembly line balancing problems. *European Journal of Operational Research, 242*(1), 343–346.

Feo, T. A., & Resende, M. G. C. (1995). Greedy randomized adaptive search procedures. *Journal of Global Optimization, 6*(2), 109–133.

Fleszar, K. (2017). A new MILP model for the accessibility windows assembly line balancing problem level 2 (AWALBP-L2). *European Journal of Operational Research, 259*(1), 169–174.

Gao, J., Sun, L., Wang, L., & Gen, M. (2009). An efficient approach for type II robotic assembly line balancing problems. *Computers & Industrial Engineering, 56*(3), 1065–1080.

García-Villoria, A., Corominas, A., Nadal, A., & Pastor, R. (2018). Solving the accessibility windows assembly line problem level 1 and variant 1 (AWALBP-L1-1) with precedence constraints. *European Journal of Operational Research, 271*(3), 882–895.

García-Villoria, A., Corominas, A., & Pastor, R. (2015). Heuristics and simulated annealing procedures for the accessibility windows assembly line problem level 1 (AWALBP-L1). *Computers & Operations Research, 62*, 1–11.

Girit, U., & Azizoğlu, M. (2021). Rebalancing the assembly lines with total squared workload and total replacement distance objectives. *International Journal of Production Research, 59*(22), 6702–6720.

Gökçen, H., Ağpak, K., & Benzer, R. (2006). Balancing of parallel assembly lines. *International Journal of Production Economics, 103*(2), 600–609.

Gökçen, H., Kara, Y., & Atasagun, Y. (2010). Integrated line balancing to attain Shojinka in a multiple straight line facility. *International Journal of Computer Integrated Manufacturing, 23*(5), 402–411.

Graves, S. C., & Lamar, B. W. (1983). An integer programming procedure for assembly system design problems. *Operations Research, 31*(3), 522–545.

Gurevsky, E., Battaïa, O., & Dolgui, A. (2012). Balancing of simple assembly lines under variations of task processing times. *Annals of Operations Research, 201*(1), 265–286.

Gurevsky, E., Hazır, Ö., Battaïa, O., & Dolgui, A. (2013). Robust balancing of straight assembly lines with interval task times. *Journal of the Operational Research Society, 64*(11), 1607–1613.

Hamta, N., Fatemi Ghomi, S., Jolai, F., & Akbarpour Shirazi, M. (2013). A hybrid PSO algorithm for a multi-objective assembly line balancing problem with flexible operation times, sequence-dependent setup times and learning effect. *International Journal of Production Economics, 141*(1), 99–111.

Hamta, N., Fatemi Ghomi, S., Jolai, F., & Bahalke, U. (2011). Bi-criteria assembly line balancing by considering flexible operation times. *Applied Mathematical Modelling, 35*(12), 5592–5608.

Hamzadayi, A., & Yildiz, G. (2012). A genetic algorithm based approach for simultaneously balancing and sequencing of mixed-model U-lines with parallel workstations and zoning constraints. *Computers & Industrial Engineering, 62*(1), 206–215.

Hamzadayi, A., & Yildiz, G. (2013). A simulated annealing algorithm based approach for balancing and sequencing of mixed-model U-lines. *Computers & Industrial Engineering, 66*(4), 1070–1084.

Hazır, Ö., Agi, M. A., & Guérin, J. (2020). A fast and effective heuristic for smoothing workloads on assembly lines: algorithm design and experimental analysis. *Computers & Operations Research, 115*, 104857.

Hazır, Ö., Agi, M. A. N., & Guérin, J. (2021). An efficient branch and bound algorithm for smoothing the workloads on simple assembly lines. *International Journal of Production Research, 59*(2), 617–634.

Hazır, Ö., Delorme, X., & Dolgui, A. (2015). A review of cost and profit oriented line design and balancing problems and solution approaches. *Annual Reviews in Control, 40*, 14–24.

Hazır, Ö., & Dolgui, A. (2013). Assembly line balancing under uncertainty: Robust optimization models and exact solution method. *Computers & Industrial Engineering, 65*(2), 261–267.

Hazır, Ö., & Dolgui, A. (2015). A decomposition based solution algorithm for U-type assembly line balancing with interval data. *Computers & Operations Research, 59*, 126–131.

Held, M., Karp, R. M., & Shareshian, R. (1963). Assembly-line balancing—dynamic programming with precedence constraints. *Operations Research, 11*(3), 442–459.

Hillier, F. S., So, K. C., & Boling, R. W. (1993). Toward characterizing the optimal allocation of storage space in production line systems with variable processing times. *Management Science, 39*(1), 126–133.

Ho, W., & Emrouznejad, A. (2009). A mathematical model for assembly line balancing model to consider disordering sequence of workstations. *Assembly Automation, 29*(1), 49–51.

Hochrein, S., Glock, C. H., et al. (2013). Systematic literature reviews in purchasing and supply management research: a tertiary study. *International Journal of Integrated Supply Management, 7*(4), 215–245.

Hoffmann, T. R. (1963). Assembly line balancing with a precedence matrix. *Management Science, 9*(4), 551–562.

Hoffmann, T. R. (1992). Eureka: A hybrid system for assembly line balancing. *Management Science, 38*(1), 39–47.

Hottenrott, A., & Grunow, M. (2019). Flexible layouts for the mixed-model assembly of heterogeneous vehicles. *OR Spectrum, 41*(4), 943–979.

Huang, D., Mao, Z., Fang, K., & Yuan, B. (2021). Combinatorial Benders decomposition for mixed-model two-sided assembly line balancing problem. *International Journal of Production Research*. (to appear)

Hudson, S., McNamara, T., & Shaaban, S. (2014). Unbalanced lines: Where are we now? *International Journal of Production Research, 53*(6), 1895–1911.

Hwang, R., & Katayama, H. (2009). A multi-decision genetic approach for workload balancing of mixed-model U-shaped assembly line systems. *International Journal of Production Research, 47*(14), 3797–3822.

Hwang, R. K., Katayama, H., & Gen, M. (2008). U-shaped assembly line balancing problem with genetic algorithm. *International Journal of Production Research, 46*(16), 4637–4649.

Jackson, J. R. (1956). A computing procedure for a line balancing problem. *Management Science, 2*(3), 261–271.

Jayaswal, S., & Agarwal, P. (2014). Balancing U-shaped assembly lines with resource dependent task times: A simulated annealing approach. *Journal of Manufacturing Systems, 33*(4), 522–534.
Johnson, R. V. (1983). A branch and bound algorithm for assembly line balancing problems with formulation irregularities. *Management Science, 29*(11), 1309–1324.
Johnson, R. V. (1988). Optimally balancing large assembly lines with “Fable”. *Management Science, 34*(2), 240–253.
Jordan, W. C., & Graves, S. C. (1995). Principles on the benefits of manufacturing process flexibility. *Management Science, 41*(4), 577–594.
Kara, Y., Atasagun, Y., Gökçen, H., Hezer, S., & Demirel, N. (2014). An integrated model to incorporate ergonomics and resource restrictions into assembly line balancing. *International Journal of Computer Integrated Manufacturing, 27*(11), 997–1007.
Kara, Y., Ozcan, U., & Peker, A. (2007). Balancing and sequencing mixed-model just-in-time U-lines with multiple objectives. *Applied Mathematics and Computation, 184*(2), 566–588.
Kara, Y., Özgüven, C., Yalçın, N., & Atasagun, Y. (2011). Balancing straight and U-shaped assembly lines with resource dependent task times. *International Journal of Production Research, 49*(21), 6387–6405.
Kellegöz, T., & Toklu, B. (2012). An efficient branch and bound algorithm for assembly line balancing problems with parallel multi-manned workstations. *Computers & Operations Research, 39*(12), 3344–3360.
Kellegöz, T., & Toklu, B. (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.
Kim, Y. K., Song, W. S., & Kim, J. H. (2009). A mathematical model and a genetic algorithm for two-sided assembly line balancing. *Computers & Operations Research, 36*(3), 853–865.
Klein, R., & Scholl, A. (1996). Maximizing the production rate in simple assembly line balancing — A branch and bound procedure. *European Journal of Operational Research, 91*(2), 367–385.
Klindworth, H., Otto, C., & Scholl, A. (2012). On a learning precedence graph concept for the automotive industry. *European Journal of Operational Research, 217*(2), 259–269.
Kottas, J. F., & Lau, H.-S. (1976). A total operating cost model for paced lines with stochastic task times. *AIIE Transactions, 8*(2), 234–240.
Kucukkoc, I., Li, Z., Karaoglan, A. D., & Zhang, D. Z. (2018). Balancing of mixed–model two-sided assembly lines with underground workstations: A mathematical model and ant colony optimization algorithm. *International Journal of Production Economics, 205*, 228–243.
Kucukkoc, I., & Zhang, D. Z. (2014a). Mathematical model and agent based solution approach for the simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines. *International Journal of Production Economics, 158*, 314–333.
Kucukkoc, I., & Zhang, D. Z. (2014b). Simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines. *International Journal of Production Research, 52*(12), 3665–3687.
Kucukkoc, I., & Zhang, D. Z. (2015). Balancing of parallel U-shaped assembly lines. *Computers & Operations Research, 64*, 233–244.
Kucukkoc, I., & Zhang, D. Z. (2017). Balancing of mixed-model parallel U-shaped assembly lines considering model sequences. *International Journal of Production Research, 55*(20), 5958–5975.
Kunz, A., Zank, M., Nescher, T., & Wegener, K. (2016). Virtual reality based time and motion study with support for real walking. *Procedia CIRP, 57*, 303–308.
Lai, T.-C., Sotskov, Y. N., Dolgui, A., & Zatsiupa, A. (2016). Stability radii of optimal assembly line balances with a fixed workstation set. *International Journal of Production Economics, 182*, 356–371.
Lapierre, S. D., Ruiz, A., & Soriano, P. (2006). Balancing assembly lines with tabu search. *European Journal of Operational Research, 168*(3), 826–837.
Lei, D., & Guo, X. (2016). Variable neighborhood search for the second type of two-sided assembly line balancing problem. *Computers & Operations Research, 72*, 183–188.
Li, D., Zhang, C., Tian, G., Shao, X., & Li, Z. (2018a). Multiobjective program and hybrid imperialist competitive algorithm for the mixed-model two-sided assembly lines subject to multiple constraints. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 48*(1), 119–129.
Li, Z., Janardhanan, M. N., Nielsen, P., & Tang, Q. (2018b). Mathematical models and simulated annealing algorithms for the robotic assembly line balancing problem. *Assembly Automation, 38*(4), 420–436.
Li, Z., Janardhanan, M. N., & Ponnambalam, S. G. (2021a). Cost-oriented robotic assembly line balancing problem with setup times multi-objective algorithms. *Journal of Intelligent Manufacturing, 32*(4), 989–1007.
Li, Z., Janardhanan, M. N., & Rahman, H. F. (2021b). Enhanced beam search heuristic for U-shaped assembly line balancing problems. *Engineering Optimization, 53*(4), 594–608.
Li, Z., Kucukkoc, I., & Nilakantan, J. M. (2017). Comprehensive review and evaluation of heuristics and meta-heuristics for two-sided assembly line balancing problem. *Computers & Operations Research, 84*, 146–161.
Li, Z., Kucukkoc, I., & Tang, Q. (2019). A comparative study of exact methods for the simple assembly line balancing problem. *Soft Computing, 142*(2), 259.
Li, Z., Kucukkoc, I., & Tang, Q. (2021c). Enhanced branch-bound-remember and iterative beam search algorithms for type II assembly line balancing problem. *Computers & Operations Research, 131*, 105235.
Li, Z., Kucukkoc, I., & Zhang, Z. (2018c). Branch, bound and remember algorithm for U-shaped assembly line balancing problem. *Computers & Industrial Engineering, 124*, 24–35.
Li, Z., Kucukkoc, I., & Zhang, Z. (2020). Branch, bound and remember algorithm for two-sided assembly line balancing problem. *European Journal of Operational Research, 284*(3), 896–905.
Li, Z., Tang, Q., & Zhang, L. (2016). Minimizing energy consumption and cycle time in two-sided robotic assembly line systems using restarted simulated annealing algorithm. *Journal of Cleaner Production, 135*, 508–522.
Lopes, T. C., Michels, A. S., Sikora, C. G. S., Molina, R. G., & Magatão, L. (2018). Balancing and cyclically sequencing synchronous, asynchronous, and hybrid unpaced assembly lines. *International Journal of Production Economics, 203*, 216–224.
Lopes, T. C., Pastre, G. V., Michels, A. S., & Magatão, L. (2020a). Flexible multi-manned assembly line balancing problem: Model, heuristic procedure, and lower bounds for line length minimization. *Omega, 95*, 102063.
Lopes, T. C., Sikora, C., Molina, R. G., Schibelbain, D., Rodrigues, L., & Magatão, L. (2017). Balancing a robotic spot welding manufacturing line: An industrial case study. *European Journal of Operational Research, 263*(3), 1033–1048.
Lopes, T. C., Sikora, C. G. S., Michels, A. S., & Magatao, L. (2020b). An iterative decomposition for asynchronous mixed-model assembly lines combining balancing, sequencing, and buffer allocation. *International Journal of Production Research, 58*(2), 615–630.
Lusa, A. (2008). A survey of the literature on the multiple or parallel assembly line balancing problem. *European Journal of Industrial Engineering, 2*(1), 50–72.
Macaskill, J. L. C. (1972). Production-line balances for mixed-model lines. *Management Science, 19*(4), 423–434.
Make, M. R. A., Rashid, M. F. F. A., & Razali, M. M. (2017). A review of two-sided assembly line balancing problem. *The International Journal of Advanced Manufacturing Technology, 89*(5-8), 1743–1763.
Maniezzo, V., Stützle, T., & Voß, S. (2009). *Matheuristics: Hybridizing Metaheuristics and Mathematical Programming*.
Martino, L., & Pastor, R. (2010). Heuristic procedures for solving the general assembly line balancing problem with setups. *International Journal of Production Research, 48*(6), 1787–1804.
Maynard, H., Stegemerten, G., & Schwab, J. (1948). *Methods-time measurement*. New York: McGraw-Hill.
Michels, A. S., Lopes, T. C., & Magatão, L. (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.
Michels, A. S., Lopes, T. C., Sikora, C. G. S., & Magatão, L. (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.
Miltenburg, G., & Wijngaard, J. (1994). The U-line line balancing problem. *Management Science, 40*(10), 1378–1388.
Minzu, V., Bratcu, A., & Henrioud, J.-M. (1999). Construction of the precedence graphs equivalent to a given set of assembly sequences. In *Towards flexible and agile assembly and manufacturing* (pp. 14–19). Piscataway, NJ: IEEE Customer Service.
Miralles, C., García-Sabater, J. P., Andrés, C., & Cardos, M. (2007). Advantages of assembly lines in sheltered work centres for disabled. a case study. *International Journal of Production Economics, 110*(1-2), 187–197.
Miralles, C., García-Sabater, J. P., Andrés, C., & Cardós, M. (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.
Mokhtarzadeh, M., Rabbani, M., & Manavizadeh, N. (2021). A novel two-stage framework for reducing ergonomic risks of a mixed-model parallel U-shaped assembly-line. *Applied Mathematical Modelling, 93*, 597–617.
Moodie, C. L., & Young, H. (1965). A heuristic method of assembly line balancing for assumptions of constant or variable work element times. *The Journal of Industrial Engineering, 16*(6), 23–29.
Moreira, M. C. O., Cordeau, J.-F., Costa, A. M., & Laporte, G. (2015a). Robust assembly line balancing with heterogeneous workers. *Computers & Industrial Engineering, 88*, 254–263.
Moreira, M. C. O., & Costa, A. M. (2013). Hybrid heuristics for planning job rotation schedules in assembly lines with heterogeneous workers. *International Journal of Production Economics, 141*(2), 552–560.
Moreira, M. C. O., Miralles, C., & Costa, A. M. (2015b). Model and heuristics for the assembly line worker integration and balancing problem. *Computers & Operations Research, 54*, 64–73.
Moreira, M. C. O., Pastor, R., Costa, A. M., & Miralles, C. (2017). The multi-objective assembly line worker integration and balancing problem of type-2. *Computers & Operations Research, 82*, 114–125.
Moreira, M. C. O., Ritt, M., Costa, A. M., & Chaves, A. A. (2012). Simple heuristics for the assembly line worker assignment and balancing problem. *Journal of Heuristics, 18*(3), 505–524.
Morrison, D. R., Sewell, E. C., & Jacobson, S. H. (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.
Mozdgir, A., Mahdavi, I., Badeleh, I. S., & Solimanpur, M. (2013). Using the Taguchi method to optimize the differential evolution algorithm parameters for minimizing the workload smoothness index in simple assembly line balancing. *Mathematical and Computer Modelling, 57*(1-2), 137–151.
Müller-Hannemann, M., & Weihe, K. (2006). Moving policies in cyclic assembly line scheduling. *Theoretical Computer Science, 351*(3), 425–436.

Mutlu, Ö., & Özgörmüş, E. (2012). A fuzzy assembly line balancing problem with physical workload constraints. *International Journal of Production Research, 50*(18), 5281–5291.
Nearchou, A. C. (2008). Multi-objective balancing of assembly lines by population heuristics. *International Journal of Production Research, 46*(8), 2275–2297.
Nearchou, A. C. (2011). Maximizing production rate and workload smoothing in assembly lines using particle swarm optimization. *International Journal of Production Economics, 129*(2), 242–250.
Nilakantan, J. M., Huang, G. Q., & Ponnambalam, S. G. (2015). An investigation on minimizing cycle time and total energy consumption in robotic assembly line systems. *Journal of Cleaner Production, 90*, 311–325.
Nilakantan, J. M., Li, Z., Bocewicz, G., Banaszak, Z., & Nielsen, P. (2019). Metaheuristic algorithms for balancing robotic assembly lines with sequence-dependent robot setup times. *Applied Mathematical Modelling, 65*, 256–270.
Nilakantan, J. M., Li, Z., Tang, Q., & Nielsen, P. (2017a). Multi-objective co-operative co-evolutionary algorithm for minimizing carbon footprint and maximizing line efficiency in robotic assembly line systems. *Journal of Cleaner Production, 156*, 124–136.
Nilakantan, J. M., Nielsen, I., Ponnambalam, S. G., & Venkataramanaiah, S. (2017b). Differential evolution algorithm for solving RALB problem using cost- and time-based models. *The International Journal of Advanced Manufacturing Technology, 89*(1-4), 311–332.
Nourmohammadi, A., & Eskandari, H. (2017). Assembly line design considering line balancing and part feeding. *Assembly Automation, 37*(1), 135–143.
Nourmohammadi, A., Eskandari, H., & Fathi, M. (2018). Design of stochastic assembly lines considering line balancing and part feeding with supermarkets. *Engineering Optimization, 51*(1), 63–83.
Nourmohammadi, A., Zandieh, M., & Tavakkoli-Moghaddam, R. (2013). An imperialist competitive algorithm for multi-objective U-type assembly line design. *Journal of Computational Science, 4*(5), 393–400.
Otto, A., & Battaïa, O. (2017). Reducing physical ergonomic risks at assembly lines by line balancing and job rotation: A survey. *Computers & Industrial Engineering, 111*, 467–480.
Otto, A., & Otto, C. (2014a). How to design effective priority rules: Example of simple assembly line balancing. *Computers & Industrial Engineering, 69*, 43–52.
Otto, A., Otto, C., & Scholl, A. (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.
Otto, A., & Scholl, A. (2011). Incorporating ergonomic risks into assembly line balancing. *European Journal of Operational Research, 212*(2), 277–286.
Otto, C., & Otto, A. (2014b). Extending assembly line balancing problem by incorporating learning effects. *International Journal of Production Research, 52*(24), 7193–7208.
Özcan, U. (2010). Balancing stochastic two-sided assembly lines: A chance-constrained, piecewise-linear, mixed integer program and a simulated annealing algorithm. *European Journal of Operational Research, 205*(1), 81–97.
Özcan, U. (2018). Balancing stochastic parallel assembly lines. *Computers & Operations Research, 99*, 109–122.
Özcan, U. (2019). Balancing and scheduling tasks in parallel assembly lines with sequence-dependent setup times. *International Journal of Production Economics, 213*, 81–96.
Özcan, U., Gökçen, H., & Toklu, B. (2010). Balancing parallel two-sided assembly lines. *International Journal of Production Research, 48*(16), 4767–4784.
Özcan, U., Kellegöz, T., & Toklu, B. (2011). A genetic algorithm for the stochastic mixed-model U-line balancing and sequencing problem. *International Journal of Production Research, 49*(6), 1605–1626.
Özcan, U., & Toklu, B. (2009a). Balancing of mixed-model two-sided assembly lines. *Computers & Industrial Engineering, 57*(1), 217–227.
Özcan, U., & Toklu, B. (2009b). A tabu search algorithm for two-sided assembly line balancing. *The International Journal of Advanced Manufacturing Technology, 43*(7–8), 822–829.
Özcan, U., & Toklu, B. (2010). Balancing two-sided assembly lines with sequence-dependent setup times. *International Journal of Production Research, 48*(18), 5363–5383.
Özceylan, E., Kalayci, C. B., Güngör, A., & Gupta, S. M. (2019). Disassembly line balancing problem: A review of the state of the art and future directions. *International Journal of Production Research, 57*(15-16), 4805–4827.
Ozdemir, R., Sarigol, I., AlMutairi, S., AlMeea, S., Murad, A., Naqi, A., & Al-Nasser, N. (2021). Fuzzy multi-objective model for assembly line balancing with ergonomic risks consideration. *International Journal of Production Economics, 239*, 108188.
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.
Pastor, R. (2011). LB-ALBP: The lexicographic bottleneck assembly line balancing problem. *International Journal of Production Research, 49*(8), 2425–2442.
Pastor, R., Andrés, C., & Miralles, C. (2010). Corrigendum to "balancing and scheduling tasks in assembly lines with sequence-dependent setup" [European Journal of Operational Research 187 (2008) 1212–1223]. *European Journal of Operational Research, 201*(1), 336.
Pastor, R., & Ferrer, L. (2009). An improved mathematical program to solve the simple assembly line balancing problem. *International Journal of Production Research, 47*(11), 2943–2959.
Pastor, R., Garcia-Villoria, A., & Corominas, A. (2011). Comparing ways of breaking symmetries in mathematical models for SALBP-1. *Assembly Automation, 31*(4), 385–387.
Pastor, R., García-Villoria, A., Laguna, M., & Martí, R. (2015). Metaheuristic procedures for the lexicographic bottleneck assembly line balancing problem. *Journal of the Operational Research Society, 66*(11), 1815–1825.
Pereira, J. (2018). The robust (minmax regret) assembly line worker assignment and balancing problem. *Computers & Operations Research, 93*, 27–40.
Pereira, J., & Álvarez-Miranda, E. (2018). An exact approach for the robust assembly line balancing problem. *Omega, 78*, 85–98.
Pereira, J., Ritt, M., & Vásquez, Ó. C. (2018). A memetic algorithm for the cost-oriented robotic assembly line balancing problem. *Computers & Operations Research, 99*, 249–261.
Petropoulos, D. I., & Nearchou, A. C. (2011). A particle swarm optimization algorithm for balancing assembly lines. *Assembly Automation, 31*(2), 118–129.
Pinto, P., Dannenbring, D. G., & Khumawala, B. M. (1975). A branch and bound algorithm for assembly line balancing with paralleling. *International Journal of Production Research, 13*(2), 183–196.
Pinto, P. A., Dannenbring, D. G., & Khumawala, B. M. (1983). Assembly line balancing with processing alternatives: an application. *Management Science, 29*(7), 817–830.
Polat, O., Kalayci, C. B., Mutlu, Ö., & Gupta, S. M. (2015). 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.
Powell, S. G. (1994). Buffer allocation in unbalanced three-station serial lines. *International Journal of Production Research, 32*(9), 2201–2217.
Rabbani, M., Kazemi, S. M., & Manavizadeh, N. (2012). Mixed model U-line balancing type-1 problem: A new approach. *Journal of Manufacturing Systems, 31*(2), 131–138.
Rabbani, M., Mousavi, Z., & Farrokhi-Asl, H. (2016). Multi-objective metaheuristics for solving a type II robotic mixed-model assembly line balancing problem. *Journal of Industrial and Production Engineering, 33*(7), 472–484.
Rachamadugu, R., & Talbot, B. (1991). Improving the equality of workload assignments in assembly lines. *International Journal of Production Research, 29*(3), 619–633.
Rajabalipour Cheshmehgaz, H., Haron, H., Kazemipour, F., & Desa, M. I. (2012). Accumulated risk of body postures in assembly line balancing problem and modeling through a multi-criteria fuzzy-genetic algorithm. *Computers & Industrial Engineering, 63*(2), 503–512.
Ramezanian, R., & Ezzatpanah, A. (2015). Modeling and solving multi-objective mixed-model assembly line balancing and worker assignment problem. *Computers & Industrial Engineering, 87*, 74–80.
Rashid, M. F. F., Hutabarat, W., & Tiwari, A. (2012). A review on assembly sequence planning and assembly line balancing optimisation using soft computing approaches. *The International Journal of Advanced Manufacturing Technology, 59*(1-4), 335–349.
Ritt, M., & Costa, A. M. (2018). Improved integer programming models for simple assembly line balancing and related problems. *International Transactions in Operational Research, 25*(4), 1345–1359.
Ritt, M., Costa, A. M., & Miralles, C. (2015). The assembly line worker assignment and balancing problem with stochastic worker availability. *International Journal of Production Research, 54*(3), 907–922.
Rosenblatt, M. J., & Carlson, R. C. (1985). Designing a production line to maximize profit. *IIE Transactions, 17*(2), 117–122.
Roshani, A., & Giglio, D. (2016). Simulated annealing algorithms for the multi–manned assembly line balancing problem: Minimising cycle time. *International Journal of Production Research, 55*(10), 2731–2751.
Roshani, A., Paolucci, M., Giglio, D., & Tonelli, F. (2021). A hybrid adaptive variable neighbourhood search approach for multi-sided assembly line balancing problem to minimise the cycle time. *International Journal of Production Research, 59*(12), 3696–3721.
Rubinovitz, J., Bukchin, J., & Lenz, E. (1993). RALB – a heuristic algorithm for design and balancing of robotic assembly lines. *CIRP Annals - Manufacturing Technology, 42*(1), 497–500.
Sabuncuoglu, I., Erel, E., & Alp, A. (2009). Ant colony optimization for the single model U-type assembly line balancing problem. *International Journal of Production Economics, 120*(2), 287–300.
Sadeghi, P., Rebelo, R. D., & Ferreira, J. S. (2018). Balancing mixed-model assembly systems in the footwear industry with a variable neighbourhood descent method. *Computers & Industrial Engineering, 121*, 161–176.
Sahin, M., & Kellegoz, T. (2019). 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.
Şahin, M., & Kellegöz, T. (2019). 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.
Salehi, M., Maleki, H. R., & Niroomand, S. (2018). A multi-objective assembly line balancing problem with worker's skill and qualification considerations in fuzzy environment. *Applied Intelligence, 48*(8), 2137–2156.
Salveson, M. E. (1955). The assembly line balancing problem. *The Journal of Industrial Engineering*, 18–25.
Sancı, E., & Azizoğlu, M. (2017). Rebalancing the assembly lines: exact solution approaches. *International Journal of Production Research, 55*(20), 5991–6010.
Schaub, K., Caragnano, G., Britzke, B., & Bruder, R. (2013). The European assembly worksheet. *Theoretical Issues in Ergonomics Science, 14*(6), 616–639.

Schaub, K. G., Mühlstedt, J., Illmann, B., Bauer, S., Fritzsche, L., Wagner, T., … Bruder, R. (2012). Ergonomic assessment of automotive assembly tasks with digital human modelling and the 'ergonomics assessment worksheet' (EAWS). *International Journal of Human Factors Modelling and Simulation, 3*(3/4), 398.
Scholl, A. (1993). Data of assembly line balancing problems. *Schriften zur Quantitativen Betriebswirtschaftslehre, TH Darmstadt, 16/93*. https://assembly-line-balancing.de/salbp/benchmark-data-sets-1993/
Scholl, A., & Becker, C. (2006). State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research, 168*(3), 666–693.
Scholl, A., & Boysen, N. (2009). Designing parallel assembly lines with split workplaces: Model and optimization procedure. *International Journal of Production Economics, 119*(1), 90–100.
Scholl, A., Boysen, N., & Fliedner, M. (2008). The sequence-dependent assembly line balancing problem. *OR Spectrum, 30*(3), 579–609.
Scholl, A., Boysen, N., & Fliedner, M. (2009). Optimally solving the alternative subgraphs assembly line balancing problem. *Annals of Operations Research, 172*(1), 243–258.
Scholl, A., Boysen, N., & Fliedner, M. (2013). The assembly line balancing and scheduling problem with sequence-dependent setup times: Problem extension, model formulation and efficient heuristics. *OR Spectrum, 35*(1), 291–320.
Scholl, A., Fliedner, M., & Boysen, N. (2010). Absalom: Balancing assembly lines with assignment restrictions. *European Journal of Operational Research, 200*(3), 688–701.
Scholl, A., & Klein, R. (1997). Salome: A bidirectional branch-and-bound procedure for assembly line balancing. *INFORMS Journal on Computing, 9*(4), 319–334.
Scholl, A., & Voß, S. (1996). Simple assembly line balancing - heuristic approaches. *Journal of Heuristics, 2*(3), 217–244.
Schöniger, J., & Spingler, J. (1989). Planung der Montageanlage. *Technica, 14*, 27–32.
Sedding, H. A. (2020). Line side placement for shorter assembly line worker paths. *IISE Transactions, 52*(2), 181–198.
Sewell, E. C., & Jacobson, S. H. (2012). A branch, bound, and remember algorithm for the simple assembly line balancing problem. *INFORMS Journal on Computing, 24*(3), 433–442.
Sgarbossa, F., Battini, D., Persona, A., & Visentin, V. (2016). Including ergonomics aspects into mixed-model assembly line balancing problem. In R. Goonetilleke, & W. Karwowski (Eds.), *Advances in physical ergonomics and human factors* (pp. 991–1001). Cham and s.l.: Springer International Publishing. Advances in Intelligent Systems and Computing
Sikora, C. G. S., Lopes, T. C., & Magatão, L. (2017). Traveling worker assembly line (re)balancing problem: Model, reduction techniques, and real case studies. *European Journal of Operational Research, 259*(3), 949–971.
Simaria, A. S., & Vilarinho, P. M. (2009). 2-ANTBAL: An ant colony optimisation algorithm for balancing two-sided assembly lines. *Computers & Industrial Engineering, 56*(2), 489–506.
Sörensen, K. (2015). Metaheuristics-the metaphor exposed. *International Transactions in Operational Research, 22*(1), 3–18.
Sprecher, A. (1999). A competitive branch-and-bound algorithm for the simple assembly line balancing problem. *International Journal of Production Research, 37*(8), 1787–1816.
Sternatz, J. (2014). Enhanced multi-Hoffmann heuristic for efficiently solving real–world assembly line balancing problems in automotive industry. *European Journal of Operational Research, 235*(3), 740–754.
Sternatz, J. (2015). The joint line balancing and material supply problem. *International Journal of Production Economics, 159*, 304–318.
Sungur, B., & Yavuz, Y. (2015). Assembly line balancing with hierarchical worker assignment. *Journal of Manufacturing Systems, 37*, 290–298.
Tang, Q., Li, Z., Zhang, L., & Zhang, C. (2017). Balancing stochastic two-sided assembly line with multiple constraints using hybrid teaching-learning-based optimization algorithm. *Computers & Operations Research, 82*, 102–113.
Tapkan, P., Özbakır, L., & Baykasoğlu, A. (2016). Bee algorithms for parallel two-sided assembly line balancing problem with walking times. *Applied Soft Computing, 39*, 275–291.
Tasan, S. O., & Tunali, S. (2008). A review of the current applications of genetic algorithms in assembly line balancing. *Journal of Intelligent Manufacturing, 19*(1), 49–69.
Thomopoulos, N. T. (1967). Line balancing-sequencing for mixed-model assembly. *Management Science, 14*(2), B–59.
Thomopoulos, N. T. (1970). Mixed model line balancing with smoothed station assignments. *Management Science, 16*(9), 593–603.
Tiacci, L. (2015a). Coupling a genetic algorithm approach and a discrete event simulator to design mixed-model un-paced assembly lines with parallel workstations and stochastic task times. *International Journal of Production Economics, 159*, 319–333.
Tiacci, L. (2015b). Simultaneous balancing and buffer allocation decisions for the design of mixed-model assembly lines with parallel workstations and stochastic task times. *International Journal of Production Economics, 162*, 201–215.
Tiacci, L. (2017). Mixed-model U-shaped assembly lines: Balancing and comparing with straight lines with buffers and parallel workstations. *Journal of Manufacturing Systems, 45*, 286–305.
Tiacci, L., & Mimmi, M. (2018). Integrating ergonomic risks evaluation through ocra index and balancing/sequencing decisions for mixed model stochastic asynchronous assembly lines. *Omega, 78*, 112–138.
Toksarı, M. D., İşleyen, S. K., Güner, E., & Baykoç, Ö. F. (2008). Simple and U-type assembly line balancing problems with a learning effect. *Applied Mathematical Modelling, 32*(12), 2954–2961.
Toksarı, M. D., İşleyen, S. K., Güner, E., & Baykoç, Ö. F. (2010). Assembly line balancing problem with deterioration tasks and learning effect. *Expert Systems with Applications, 37*(2), 1223–1228.
Triki, H., Mellouli, A., & Masmoudi, F. (2017). A multi-objective genetic algorithm for assembly line resource assignment and balancing problem of type 2 (ALRABP-2). *Journal of Intelligent Manufacturing, 28*(2), 371–385.
Urban, T. L. (1998). Optimal balancing of U-shaped assembly lines. *Management Science, 44*(5), 738–741.
Urban, T. L., & Chiang, W. C. (2016). Designing energy-efficient serial production lines: The unpaced synchronous line-balancing problem. *European Journal of Operational Research, 248*(3), 789–801.
Vilà, M., & Pereira, J. (2013). An enumeration procedure for the assembly line balancing problem based on branching by non-decreasing idle time. *European Journal of Operational Research, 229*(1), 106–113.
Walter, R., & Schulze, P. (2021). On the performance of task-oriented branch-and-bound algorithms for workload smoothing in simple assembly line balancing. *International Journal of Production Research*. (to appear)
Walter, R., Schulze, P., & Scholl, A. (2021). SALSA: Combining branch-and-bound with dynamic programming to smoothen workloads in simple assembly line balancing. *European Journal of Operational Research, 295*, 857–873.
Wang, B., Guan, Z., Li, D., Zhang, C., & Chen, L. (2014). Two-sided assembly line balancing with operator number and task constraints: a hybrid imperialist competitive algorithm. *The International Journal of Advanced Manufacturing Technology, 74*(5-8), 791–805.
White, W. W. (1961). Letter to the editor – comments on a paper by Bowman. *Operations Research, 9*(2), 274–276.
Yolmeh, A., & Kianfar, F. (2012). An efficient hybrid genetic algorithm to solve assembly line balancing problem with sequence-dependent setup times. *Computers & Industrial Engineering, 62*(4), 936–945.
Yoosefelahi, A., Aminnayeri, M., Mosadegh, H., & Ardakani, H. D. (2012). Type II robotic assembly line balancing problem: An evolution strategies algorithm for a multi-objective model. *Journal of Manufacturing Systems, 31*(2), 139–151.
Yu, J., & Yin, Y. (2010). Assembly line balancing based on an adaptive genetic algorithm. *The International Journal of Advanced Manufacturing Technology, 48*(1–4), 347–354.
Zacharia, P., & Nearchou, A. C. (2013). A meta-heuristic algorithm for the fuzzy assembly line balancing type-e problem. *Computers & Operations Research, 40*(12), 3033–3044.
Zacharia, P. T., & Nearchou, A. C. (2012). Multi-objective fuzzy assembly line balancing using genetic algorithms. *Journal of Intelligent Manufacturing, 23*(3), 615–627.
Zandin, K. B. (2002). *MOST work measurement systems*. Boca Raton: CRC Press.
Zäpfel, G. (1975). *Ausgewählte fertigungswirtschaftliche Optimierungsprobleme von Fließfertigungssystemen*. Beuth.
Zhang, H.-y. (2019). An immune genetic algorithm for simple assembly line balancing problem of type 1. *Assembly Automation, 39*(1), 113–123.
Zhang, Y., Hu, X., & Wu, C. (2020a). Improved imperialist competitive algorithms for rebalancing multi-objective two-sided assembly lines with space and resource constraints. *International Journal of Production Research, 58*(12), 3589–3617.
Zhang, Z., Tang, Q., & Chica, M. (2020b). Multi-manned assembly line balancing with time and space constraints a MILP model and memetic ant colony system. *Computers & Industrial Engineering,, 150*, 106862.
Zhang, Z., Tang, Q., Li, Z., & Zhang, L. (2019). Modelling and optimisation of energy-efficient U-shaped robotic assembly line balancing problems. *International Journal of Production Research, 57*(17), 5520–5537.
Zhang, Z., Tang, Q., Ruiz, R., & Zhang, L. (2020c). Ergonomic risk and cycle time minimization for the U-shaped worker assignment assembly line balancing problem: A multi-objective approach. *Computers & Operations Research, 118*, 104905.
Zheng, Q., Li, M., Li, Y., & Tang, Q. (2013). Station ant colony optimization for the type 2 assembly line balancing problem. *The International Journal of Advanced Manufacturing Technology, 66*(9-12), 1859–1870.
Zohali, H., Naderi, B., & Roshanaei, V. (2021). Solving the type-2 assembly line balancing with setups using logic-based Benders decomposition. *INFORMS Journal on Computing*. (to appear)