Int J Adv Manuf Technol (2017) 89:1743–1763
DOI 10.1007/s00170-016-9158-3

ORIGINAL ARTICLE

A review of two-sided assembly line balancing problem

Muhammad Razif Abdullah Make [1] · **Mohd Fadzil Faisae Ab. Rashid** [1] · **Muhamad Magffierah Razali** [1]

Received: 27 March 2016 / Accepted: 8 July 2016 / Published online: 2 August 2016

Abstract Assembly line balancing (ALB) is concerned with assigning tasks within an assembly line to meet the required production rate for optimization purposes. On the other hand, two-sided ALB performs double-sided assembly operation on a single assembly line. In this paper, we have focused the survey on two-sided assembly line balancing (2S-ALB) research problems. The numerous factors mentioned in 2S-ALB literature were actually based on problem resolutions, and this paper will quote any preferred literature considering the frequent citation. In particular, this review explores in detail the ALB problems, optimization methods, objective functions, and specific constraints used in solving 2S-ALB problems. Among the purposes of ALB problems is that it traditionally focuses on simple ALB with various engaging approaches. General ALB comes second because of its complexity and nondeterministic polynomial (NP)-hard-classified problems. However, due to the current manufacturing issues, GALB problems, such as 2S-ALB, are forced to be examined and this comprehensive literature will specify anything necessary for the optimization purposes. Finally, future research direction has been discovered and put forward as the suggestion.

Keywords Assembly line balancing · Two-sided · Artificial intelligence

✉ Mohd Fadzil Faisae Ab. Rashid
ffaisae@ump.edu.my

[1] Manufacturing Focus Group, Faculty of Mechanical Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia

1 Introduction

In a modern manufacturing system, assembly line balancing (ALB) plays a vital function, especially in the production line. The installation of an assembly line is a long-term decision and requires large capital investments. It is important that such a system is designed and balanced so that it is able to work as efficiently as possible [1–3]. The assembly line was introduced by Henry Ford in his automobile plants. Since then, many developments through researches have been introduced [4].

Generally, from the feature of the product and technical operational requirement, there have been differences in the line balancing problem classifications made by the researchers. For instance, [5] classified the line balancing problems into simple and general types of problems. The similar classification was also used by [6, 7]. Besides that, the line balancing also was classified according to the model number (single-model and multimodel) and the nature of task times (deterministic and stochastic) by [8–10]. On the other hand, [11, 12] classified the line balancing problems into two types: one-sided and two-sided ALB problems.

Both two types of assembly lines are quite famous among researchers. One-sided assembly, or commonly called single-sided assembly line, was examined extensively in the past few decades. The assembly line is a flow-line production system in which a series of stations are arranged along a conveyor belt or a similar mechanical material handling system [13]. The stations are often prepared in a single line which is long enough to complete the desired product with different types of tasks or assembly processes, as illustrated in Fig. 1. Frequently, every station only has one operator to manage each task and fully run the assembly line. The operator cannot leave the station when the assembly process is running.

Although the focus of researches are always on one-sided assembly lines, two-sided assembly lines are recognized to be

Fig. 1 Single-sided assembly line

crucially important too, especially in the assembly of the large-sized products like cars, busses, or trucks [14]. Two-sided assembly line or often called the double-sided assembly line is absolutely different compared to the one-sided or single-sided assembly line. In the two-sided assembly lines, the operating direction of the assembly tasks will be carried out on the same product in parallel at both the left and right sides of the lines. Due to the use of both sides of the lines, the tasks will have additional operating direction restrictions. The directions can be classified into three types: the left side (L), the right side (R), and either side (E) [13, 15, 16]. Figure 2 illustrates the example of a two-sided assembly line. One of the main differences between the single- and two-sided assemblies is the restriction on the operating directions. Some of the assembly operations can be performed at only one of the sides, while others can be performed on either side of the lines [17]. The two-sided assembly line used both of the lines to enhance the assembly performance of a complex production system such as in automotive industries.

In industry, line balancing is very important in taking advantage of them. Unbalanced lines may incur unnecessary cost [4]. Hence, ALB was raised among researchers in order to satisfy the workload and increase the operational line efficiency [18]. The activity of balancing operations has appeared since 1955 [4]. In order to increase line efficiency, the balancing operations are responsible to determine the set of tasks. ALB is generally classified into two, either simple assembly line balancing (SALB) or general assembly line balancing (GALB) [6, 10]. Figure 3 illustrates the ALB classification and problem-type examples.

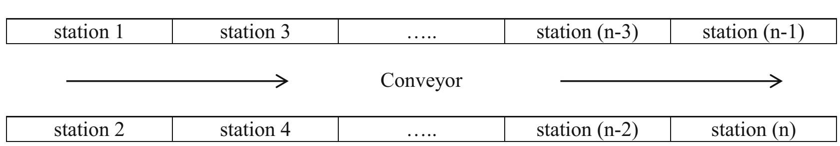
Normally, SALB studies only on a single side of the assembly line (Fig. 1). Even so, SALB problems have been categorized into some classes. First, SALB in type 1 (SALB-1) will perform the minimization of workstation numbers for a given cycle time as the objective [19, 20]. Second, type 2 assembly will consider the minimization of cycle time with the given number of workstations [19, 21]. Next is SALB for type E problems which is different in line configuration from the single-sided ALB [22–25]. This SALB-E type is significantly believed to have its own advantage. Another class in SALB is type F that was categorized by Kriengkorakot and Pianthong (2007) in their studies [10]. SALB type F also has been discussed in the Assembly Line Balancing book by Micieta B. and Stollmann V. (2011) [26].

On the other hand, another ALB class also has been justified in general form (GALB) [27]. First was by the two-sided assembly line balancing (2S-ALB) [28–31], followed by the mixed model of ALB (MALB) [4, 32–35], and then the U-type of ALB [36–40]. The GALB of 2S-ALB will be discussed in detail in another section on the ALB problems. The MALB problem is normally for high production with multiple types of products [41]. However, in GALB, there are many other types of assemblies that have been introduced by other researchers [42]. It was also included with combinatorial problem types. The GALB is always considered as the nondeterministic polynomial (NP)-hard problem due to its complexity.

The development of 2S-ALB in GALB seems more crucial over single-sided assemblies. Two-sided assembly lines were introduced in 1993 by Bartholdi [43], who conducted an iterative program with balancing algorithm using the first fit heuristic. Then, consequently, it was continued by other researchers [44–46], with genetic algorithm as the solving approach. Meanwhile, [47] has proposed an ant colony algorithm to solve two assembly line balancing problems focusing on the minimization of workstations and the maximization of work relatedness. In addition, the simulated annealing for a two-sided assembly problem are successfully presented in [48, 49] and again for mixed-model two-sided assembly line balancing problem in [50].

In previous conducted research, the objective function has also been given considerable attention. In [4, 13, 46, 50–52], the minimizing number of workstations and mated stations are selected as the applied objectives. Besides, [15, 36, 53, 54] have adopted the minimizing number of workstation and line length as their preferred objective function. Apart from that, multiobjective function is also presented in studies by [16, 29, 48, 55, 56]. A continuous evaluation toward 2S-ALB constraint for single [17, 30, 46, 47, 56] and multiple considered constraints [29, 49] is addressed very well. This paper reviews on 2S-ALB problems to address problem types, optimization methods, objective functions, and considered constraints. Many optimization problems have been successfully studied by other researchers. Hence, this review will discuss through their literature and studies.

Fig. 2 Two-sided assembly line



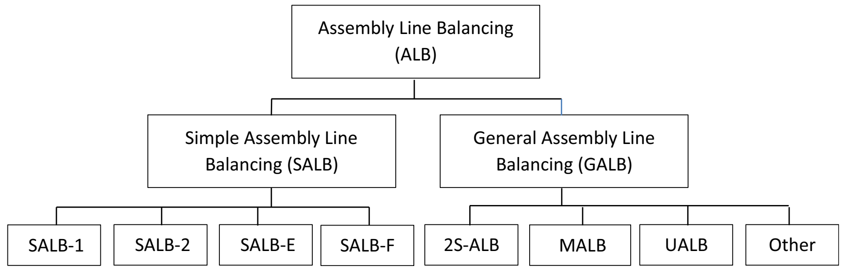


Fig. 3 ALB classification

2 Assembly line balancing problem

The assembly line balancing problem (ALBP) was first mathematically formulated by Salveson in 1955 [26, 29]. ALBP is the problem of assigning tasks to stations in such a way that one or more objectives are optimized, subjected to some specific constraints. Since then, many researches on assembly lines have included the exact solution methods, heuristics, and metaheuristic approaches reported in the literature. The heuristic method is actually an answering method to produce a solution that applies the trial-and-error strategy with a reasonable functional period [57]. On the other hand, the metaheuristic approach is an independent solution strategy that is believed to resolve any optimization problem. The ALBP assigns tasks in an ordered fashion to every workstation by satisfying specific constraints [58–60]. The related studies on ALB are classified in various types of problems. Since this study considers 2S-ALB as the main problem, the other ALB problems will also be discussed.

2.1 General two-sided assembly line balancing

Two-sided assembly line was introduced for the first time by Bartholdi in 1993 [16, 59] to produce high volumes of large-sized products. Typically, the 2S-ALB production of bus and trucks by Kim, Kim, and Kim in 2000 [44], automobile by Lee et al. in 2001 [11], and a domestic product by Baykasoglu and Dereli (2008) [47] studied on the problem to find the best solution. Since then, numerous researches have brought up different methods, either heuristic or metaheuristic as the solution approach.

As shown in Fig. 2, 2S-ALB generally has a pair of workstations/stations facing each other almost in all operation lines. A pair of stations facing each other, e.g., station 1 and station 2, is called "mated station," and one of the stations is called "companion" [44]. Every operating station will perform a different task despite the stations being opposite of each other. Large assembly production industries such as cars and trucks need this type of assembly lines for them to perform the assembly operation at a given time to achieve their productivity target. The operational process in 2S-ALB will provide several advantages compared to the single-sided ALB [14].

The comparisons between 2S-ALB and single-sided ALB are remarkably different. As indicated earlier, 2S-ALB has a more complex layout against single-sided ALB. Figure 4 illustrates the 2S-ALB layout with the left and right sides of the workstation. The number in the boxes indicates the task on every side of the workstation. For a single-sided ALB, the sides are definitely not crucially tough, because it only has a single-side layout of the operation line [29, 61]. In addition, the task distribution between the compared ALB problems differs as well. Having two preferred sides of the assembly line, 2S-ALB definitely needs to distribute every task accordingly. For a single-sided assembly, the precedence relation is considered appropriate with all the tasks assigned to a workstation that can be carried out without any interruption. However, its difference from 2S-ALB is that some of the tasks assigned could be delayed after the assigned task of its companion [30, 43], or commonly known as idle time [48]. The shaded area in Fig. 4 denotes the idle time which is unavoidable in completing the 2S-ALB processing product.

Throughout the 2S-ALB installation, it will lead to some valuable advantages of the assembly lines [45], such as the following:

1. Shortens the assembly line
2. Saves some spaces on the assembly lines
3. Reduces the cost of tools and fixtures
4. Reduces the throughput time
5. Reduces material handling

Based on these advantages, it is considered to have the ability to maximize the productivity of the assembly lines. In reality, high attention has been given for the study of 2S-ALB

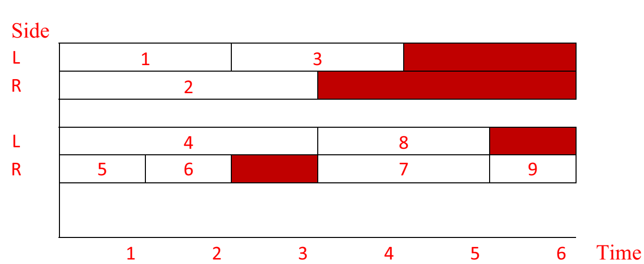


Fig. 4 2S-ALB task distribution layout

problems. Two decades have passed since it was firstly introduced by Bartholdi for his large volumes of vehicle production [43]. Initially, the aim is to simplify and facilitate the manufacturing industries of the 2S-ALB functional capabilities. Up until now, the studies focusing on 2S-ALB problems continue to attract participation among public researchers. Nowadays, 2S-ALB has also been implemented in other manufacturing industries, for instance, in furniture and electrical appliance production [4]. Moreover, the 2S-ALB problem complexity is also growing day after day, with the combination and hybridization of the ALB problems. Therefore, currently, better performance of optimization algorithm is favorably built by researchers in line with the 2S-ALB progress. Besides, introducing and emphasizing a new framework for 2S-ALB problem solutions are also involved.

In 2S-ALB studies, researchers have stated and applied many different kinds of approaches successfully. For example, Purnomo and Wee (2014) [30] proposed the harmony search method to solve 2S-ALB problems. On the other hand, in the [44–46] studies, the genetic algorithm method was performed. The simulated annealing algorithm was also applied in some other studies [48, 49]. Besides, the heuristic approach, which is a problem-dependent method, has been addressed as well for 2S-ALB problems [11, 47, 51, 62].

2.2 Two-sided mixed-model assembly line balancing

The mixed-model line balancing problem was first introduced by Thomopoulos in 1967 [55]. Considering today's competitive market, the MALB has become more advantageous rather compared with a single model assembly line balancing. A single model assembly line only designs a single standardized homogenous product, while the mixed-model assembly lines are widely applied to produce two or more product models depending on the customer needs [4]. In other words, a mixed-model assembly line is designed to produce a similar set of products in the different mixed-ordered model.

The existing research for the MALB problem addressed single- and multi-objective problems under various assembly line considerations. Commonly, in configuring a mixed-model assembly line, a lot of goals and objectives are considered. There are two goals that have been studied by most researchers [63] in balancing the mixed-model assembly lines:

1. Leveling workloads for every station on the line
2. Leveling part usage on the line

The first goal of leveling the workload for all stations on the line is attempting to achieve a balanced workload at specific times for each assembly task, while the second goal is attempting to minimize the variation used by the different parts over time.

In order to fill customer requirement, MALB is applied widely in a range of industries; for instance, in the production of electrical appliances, furniture, and clothing [4]. In automotive industries, the mixed-model assembly line was broadly introduced in combination with the two-sided assembly line. For example, [4, 14, 16, 50] studied a two-sided mixed-model assembly line balancing established with a different solution balancing approach. These optimally gave a positive effect on the large-sized high-volume production industries such as in automobile and appliance factories.

2.3 Two-sided parallel assembly line balancing

The parallel line configuration idea in ALB was started by Suer and Dagli in 1994 [59]. The combination of two or more lines placed parallel to each other became an idea of sharing tools and fixtures to complete an entire job. The balancing idea of P-ALB was studied by Gökçen, Agpak, and Benzer in 2006 [64] with the title Balancing of Parallel Assembly Lines. They proposed a new procedure with a mathematical model on the single-model assembly line balancing problem with parallel lines. Since then, the researcher broadly continued the study on the P-ALB problem. Various approaches and ideas to solve the P-ALB problem then emerged. Cercioglu, Ozcan, Gokcen, and Toklu (2009) proposed a simulated annealing approach for solving the P-ALB [65]. Meanwhile, Ozcan, Cercioglu, Gokcen, and Toklu (2009) firstly utilized a multiobjective Tabu search algorithm method on parallel assembly lines [66]. A novel ant colony optimization (ACO-based algorithm also became one of the methods for solving the P-ALB problems by Baykasoglu, Ozbakir, Gorkemli, and Gorkemli in 2009 [59, 60].

Parallel assembly lines, usually built with two or more lines, are located parallel to each other. This provides the following advantages [64] to the lines:

1. Shortens the assembly lines
2. Steadily runs during breakdown

By installing parallel configuration of the assembly lines, it definitely shortens the assembly lines. Besides, being able to locate only one operator in between the adjacent station helps the operator to perform both tasks. These completely utilize the workers on the assembly lines [60]. Another advantage of parallel assembly lines is that it could still be run steadily even when a workstation faces a problem or breakdown [64]. A single assembly line will stop the assembly operation if any workstation faces a problem, but P-ALB will continue to run and perform the task at the other adjacent lines. The advantages of parallel assembly lines over a single assembly line were also discussed by Ozcan, Gokcen, and Toklu (2010) [66]. It is able to provide much more benefits:

1. It can help to produce similar products or different models of the same production of the adjacent lines.
2. It can reduce the idle time and increase the efficiency of the assembly lines.
3. It is able to make production with a different cycle time for each of the lines.
4. It can improve visibility and communication skills between operators.
5. It is also able to reduce operator requirements.

The combination types of production lines for the parallel and two-sided lines were already studied. The parallel two-sided assembly line balancing problem (PTALBP) was firstly developed by Ozcan, Gokcen, and Toklu in 2010 [66], focusing on the large-sized productions in different industries. Other studies on the PTALB problem were reviewed by Ağpak and Zolfaghari (2015) [53] and Kucukkoc and Zhang (2015) [60].

3 Optimization method

Since 1955 by Salveson, various researches regarding the ALB solving problem were introduced [27]. The researchers focused on improving the assembly line, so that it was able to work efficiently. In 1993, Bartholdi first presented his idea to address the 2S-ALBPs. He discussed some theoretical properties of the 2S-ALB and proposed the first-fit heuristic algorithm method of assigning tasks to workstations [15, 30]. Since then, numerous researches concerning the ALB solution problems using different methods have been introduced. The mathematical model and heuristic and metaheuristic methods were developed to solve the ALB for different problem types. Table 1 summarizes the optimization method used in previous researches of the 2S-ALB problems. Meanwhile, Fig. 5 below shows the number of research papers that have successfully implemented the different types of algorithm by using different soft computational methods.

The number under the graph (Fig. 5) represents the different types of optimization in the 2S-ALB research paper (see Table 1 legend). Mostly all of these metaheuristic algorithm methods were inspired by natural phenomena. Among them, the effectiveness in solving the NP-hard optimization problem became necessarily important. The ALB problem type became complex day after day and the high capability of the algorithm method seems more needed. From the survey, the most (metaheuristic) frequent optimization algorithms used are genetic algorithm (GA) and ant colony optimization (ACO) algorithm (used five times from 30 papers) followed by simulated annealing (SA). The high applied value of GA and ACO in Fig. 5 shows the popularity and stability of these methods in solving the 2S-ALB problems. Others might be less studied because the relatively new algorithm and efficiency of the method was not well proven yet.

3.1 Genetic algorithm

Genetic algorithm was formally introduced in 1970s in the University of Michigan by John Holland. GA has been proven to be very efficient and powerful in a wide variety of applications [44]. It provides a method to find the best sequence of assembly process among the possible sequences that have been generated either in constrained or unconstrained condition. GA is also considered as one of the artificial algorithm methods or artificial intelligence-based algorithms in solving the ALB problems. The accomplishment of GA in solving difficult and complex combinatorial problems is seen to have outperformed the other algorithms in terms of solution quality and convergence speed [46]. Genetic algorithm is believed to be able to find the optimal or nearly optimal assembly plans for the model structure generated by analyzing the small number of possible solutions.

Algorithm starts with a set of solutions (represented by chromosomes) called population. Solutions from one population are taken and used to form a new population [32]. This is motivated by a hope that the new population will be better than the previous one. Solutions which are selected to form new solutions (offspring) are selected according to their fitness—the more suitable they are, the more chances they have to reproduce. This process is repeated until some conditions (improvement of the best solution) are satisfied.

In solving 2S-ALB problems, many researchers have used the GA method [12, 44–46, 60]. The reputation of GA was first addressed by Kim and Kim et al. [44, 46] in solving the 2S-ALB problems. They successfully implemented the GA method with the objective of minimizing the number of stations with a given cycle time. In 2009, Song et al. used a mathematical model and GA for the 2S-ALB problem with different objectives of minimizing the cycle time [45]. Both studies have inspired other researchers to implement GA in solving other types of balancing problems in assemblies.

Many compliments and praises were given to the performance of the GA method in solving different kinds of complex combinatorial problems nowadays [32]. However, some weaknesses arise since GA has been used in the ALB problems [6]. The premature convergence turned into an issue [68, 69]. This appears to be due to that GA sequences heavily depend on the initial generating sequence. Besides, it requires a high amount of computational time in order to find the final solution [70]. Conversely, in a study by [71], the GA method behavior has been discussed to greatly depend on numerous control parameters and only used simple test data. The disadvantage and weaknesses of the GA method should be considered for future research direction.

Table 1 Method of optimization for 2S-ALB problems

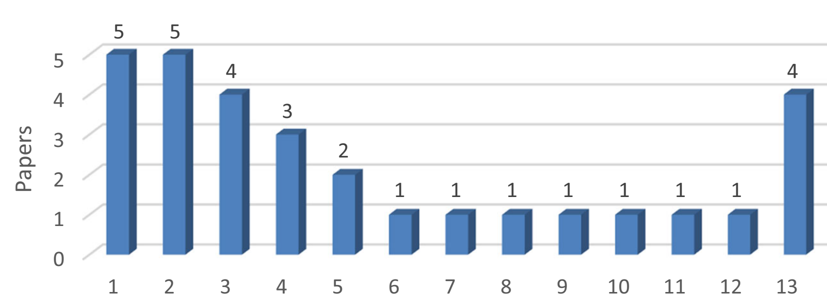
Author/s, Year	Ref.	Optimization method												
		1	2	3	4	5	6	7	8	9	10	11	12	13
Yuan, Zhang et al. 2015	[4]				x									
Kucukkoc and Zhang 2015	[59]		x											
Kucukkoc and Zhang 2015	[60]	x												
Chiang, Urban et al. 2015	[15]					x								
Ağpak and Zolfaghari 2015	[53]													x
Tuncel and Aydin 2014	[29]							x						
Purnomo and Wee 2014	[30]						x							
Kucukkoc and Zhang 2014	[55]		x											
Kucukkoc and Zhang 2014	[54]		x											
Chutima and Naruemitwong 2014	[56]								x					
Purnomo, Wee et al. 2013	[12]	x												
Khorasanian, Hejazi et al. 2013	[48]			x										
Tapkan, Ozbakir et al. 2012	[61]				x									
Roshani, Fattahi et al. 2012	[49]			x										
Chutima and Chimklai 2012	[16]					x								
Ağpak, Yegül et al. 2012	[36]									x				
Taha, El-Kharbotly et al. 2011	[46]	x												
Özbakır and Tapkan 2011	[17]				x									
Xiaofeng, Erfei et al. 2010	[67]										x			
Özcan and Toklu 2010	[51]													x
Özcan, Gökçen et al. 2010	[66]											x		
Özcan 2010	[13]			x										
Simaria and Vilarinho 2009	[14]		x											
Özcan and Toklu 2009	[50]			x										
Özcan and Toklu 2009	[52]												x	
Kim, Song et al. 2009	[45]	x												
Hu, Wu et al. 2008	[62]													x
Baykasoglu and Dereli 2008	[47]		x											
Lee, Kim et al. 2001	[11]													x
Kim, Kim et al. 2000	[44]	x												

Optimization method: 1—genetic algorithm, 2—ant colony optimization, 3—simulated annealing, 4—bee algorithm, 5—particle swarm optimization, 6—harmony search, 7—teaching learning based optimization, 8—Pareto biogeography based optimization, 9—lexicographic optimization method, 10—branch and bound, 11—Tabu search, 12—goal and fuzzy goal, 13—other heuristic methods

3.2 Ant colony optimization

Ant colony algorithm is one of the most famous metaheuristic methods that have already been used successfully for solving various problems in ALB. It was introduced in the early 1990s by Marco Dorigo [72]. The ACO algorithm method studied by [6] was also considered to have high reputation following under the GA fame in solving many types of ALB problems. It

Fig. 5 Number of papers applied different algorithm methods in 2S-ALB



was already assessed to be fit in overcoming and solving even in high combination problems.

The ACO algorithm was inspired by the behavior of a real ant colony finding a path between the food source and its nest. The pheromone trail released by the other ants will be followed. Each ant from the colony will come out with a different path. The ants which pick the shortest path will return to the nest faster; hence, there will be much more pheromone trail on the shortest path. It influences other ants to follow that path [54]. The pheromone trail of an ant path was considered a solution in the algorithm, and the performance will be evaluated according to its quality of accomplishment in obtaining the final execution or solution [55].

Previous studies that successfully presented and used the ACO method from the literature provided essential trend in solving the 2S-ALB problems. The study by Baykasoglu and Dereli (2008) [47] followed by Simaria and Vilarinho (2009) [14] presented the successful achievement in balancing the 2S-ALB problems. While in 2014, Kucukkoc and Zhang became the first pioneer to address the ACO method through the mixed-model parallel two-sided assembly line balancing problem with model variations [54, 55]. They were practically successful in implementing this algorithm method into large-sized products. Then, in 2015, the knowledge of the type-E parallel two-sided assembly line balancing problem was introduced [59] for the first time in literature by Kucukkoc and Zhang in their research; type-E parallel two-sided assembly line balancing problem: Mathematical model and ant colony optimization-based approach with optimized parameters.

From the previous published literature, the ACO algorithm contributed to be competitive in solving different kinds of ALB problems, despite its strong global search ability. However, this evolutionary algorithm also holds its own weakness and limit. For example, the pheromone trail path made by the ant always evaporates and disappears if the path is bad [6]. Therefore, it will also cause premature convergence. Nevertheless, the premature convergence in the ACO algorithm method has been solved by [73].

3.3 Simulated annealing

Numerous studies on ALB performed different approaches on the optimization method for solving the assembly problems. The heuristic, metaheuristic, and also exact approach solutions were introduced and have been reported in literature. The SA algorithm became one of the leading metaheuristic approaches in solving multiple cases or problems of ALB. It was first applied by Kirkpatrick et al. in 1983 in solving a combinatorial optimization [13].

The simulated annealing algorithm was originally inspired by the annealing process in metal works [74]. The heating and cooling process were involved against the material to alter the physical properties due to the changes in the internal structure. In the SA optimization method, it was initially set high and then allowed to cool slowly. The chance of accepting solutions actually gives the algorithm the ability to find early execution before generating the optimal solution.

The simulated annealing algorithm in solving the ALB problem is currently studied by many researchers. The review of such study was given by [13, 48–50]. They successfully implemented the SA method in their studies in minimizing or maximizing something through their objectives. The SA method provides several advantages in the ALB such as the reasonable computational time and good performance in determining the optimal solution on every sized problem [49]. This has outperformed other methods in terms of solution quality.

However, this iterative random search technique (SA) also has its weakness. The SA method is believed to be able to jump into a local optimal solution by accepting the bad solution [13, 50]. This condition will create an opportunity for the bad solution to be selected as the optimal and final solution. Other drawbacks in the SA method are stated as follows:

1. The procedure method will stop when the stopping criterion is reached in getting the optimal solution.
2. The initial solution starts with low solution value.

The two above drawbacks need a proper attention with respect to the 2S-ALB problems and for getting the optimal solution.

3.4 Other optimization methods

Besides the three optimization methods discussed earlier, there are other metaheuristic approaches used by researchers in solving the 2S-ALB problems. These algorithms are the hybrid honey bee mating optimization (HHBMO) algorithm [4], bee algorithm (BA) [17, 61], particle swarm optimization (PSO) [15], teaching learning based optimization (TLBO) [29], harmony search (HS) [30], Pareto biogeography-based optimization (PBBO) [56], particle swarm optimization with negative knowledge (PSONK) [16], lexicographic optimization method (LOM) [36], branch and bound (B&B) [67], Tabu search (TS) [33, 66], and goal and fuzzy goal programming (G&FG) [52]. Otherwise, in some researches, they applied the heuristic (problem-dependent) approach such as in [11, 47, 51, 53, 62].

As far as the search methods are concerned with the popularity efficiency, other optimization methods become neglected. However, this is different with the GA and ACO algorithm methods. They were introduced more than decades ago and the performance of optimization in various kinds of ALB problems are well known. The optimization algorithm recognition is basically based on the performance efficiency and robustness in solving differently sized (small, medium, and large) problems. Hence, it requires a long time for researchers to find and test other methods that are considered relatively new. Nevertheless, the evolutionary combination of the optimization method seems to be able to raise the new algorithm to be getting highlighted through better performance.

3.5 Comparison of different optimization methods

Among the previous researches, the optimization method in ALB strongly gives an impact to the industries. Different methods of optimization either heuristic or metaheuristic successfully develop prior to each research study. In ALB problems, GA, ACO, SA, and other relatively new algorithm optimization methods definitely attempt to balance the assembly line with high values of line efficiency. However, all those optimization methods already serve with their own advantages and some weaknesses.

A successful GA method has been recently presented with a complex combinatorial problem with more numbers of studies possessing the searching method ability. It does not require examination of all the possible solutions but uniquely, it is still able to obtain the best feasible result [6]. In [75], GA is also believed to be able to handle complex and multiple constraint problems very well, even though the premature converge [76, 77] and high amount of computational time [70, 78] became an issue. Therefore, another study has been developed to overcome the raised issues by introducing dynamic partitioning (DPa) in chromosome [76] and the combination with other soft computing algorithms [68, 69]. Kucukkoc and Zhang [60] successfully compared the obtained result with the result of Gökçen et al. [66]. By this, they have obtained a very encouraging performance as shown by GA.

Meanwhile, similar to the ACO method, it also contributes in solving various kinds of discretized ALB problems. Besides, this method is also believed to directly be able to present in a completed ACO graph [6, 79]. Furthermore, the ACO method also appears as the maximum citation paper after GA in five applied journals (Fig. 5). Despite the ACO sensation, it also comes with some confusion. In [80], a premature convergence is stated as a drawback when implementing the rule of the ACO method. For this reason, [73] have introduced a summation updating the rule to overcome this matter. Besides, an adopted particle swarm updating position has also succeeded in solving this outcome matter [80]. The hybridization of ACO and PSO method is able to solve the premature convergence and then significantly reduce the computational time. Baykasoglu et al. [81] have proposed a novel ant colony optimization-based algorithm for PALBP. They compared their test results with three other existing approaches from the literature to prove the efficiency of the proposed algorithm.

This forward to the SA optimization method presented on 1983 which is the reasonable period of computational time being recognized greatly in ALB optimization. This method outperforms the other methods by allowing faster solving solution even for a larger problem [49]. The high reputation of the SA method is practically easy to use and extremely popular in solving practical problems such as job-shop scheduling, traveling salesman, and timetabling problem [82]. Nowadays, the SA method is frequently compared with GA, besides hybridization of these two optimization methods. The main aim of hybridization is to avoid being trapped by a local minima and to have faster convergence. By this, the advantage of both methods could be developed [83]. In Cercioglu et al. [65], a simulated annealing approach in solving the PALBP is proposed. A comparison between the obtained results with the existing heuristic algorithm proposed by Gökçen, Agpak, and Benzer (2006) [64] has also been reported.

Besides the three abovementioned methods, there are many other optimization methods that have successfully shown its accomplishment. Bee algorithm applied by Ozbakir and Tapkan [17] presented for balancing the 2S-AL has effectively compared the optimization result with four other research results in seven differently sized problems. Considerably, it is best to know that the GA method has performed better solutions in computational time than other approaches did including ACO. This is followed by the PSO started by Kennedy and Eberhart in 1995 [84]. Although PSO algorithm is relatively new compared with GA and ACO, this method also holds a good criterion for being selected as an optimization method. Inspired by the social behavior of birds flocking together, the PSO has a simple algorithm with a single velocity formula to evolve and less computational resource compared with GA [6]. Chutima and Chimklai [16] have compared PSO with two different optimizations and significantly showed that results by the PSO method were much better with a simple but robust algorithm performance. This means that the other new algorithms also show a favorable appearance besides those former algorithms.

4 Objective function

Objective function is the computed measure used to evaluate the performance of assembly line. It is widely used mainly in decision analysis, operation research, and optimization studies [85]. The objective function is critically important for a research mainly in the optimization study. Conversely, in the ALB optimization research, objective function becomes necessarily important. These will strictly guide researchers to keep their direction in finding the best solution to their problems. In most studies, the objective function will define the optimization problems and either the tasks or even the installation requirement setup would need to be minimized or maximized.

All the earlier studies possess their own objective through their research. Most of them have been studied and used the multiobjective function approach rather than the single objective function. Table 2 shows the objective function used in the previous researches; Fig. 6 presents the number of researches that successfully implemented the different types of objective functions in the ALB problems.

The numbers under the graph represent different types of objective function of 2S-ALB problem (see legend of Table 2). The most popular objective function in 2S-ALB problems is to minimize the number of workstations with 21 counts from 30 papers, while the minimization of the mated station number has taken the second place in the objective function popularity.

4.1 Minimizing the workstation number

During the last decade, researchers have begun to study the 2S-ALB problems recognized to be crucially important in real life. They have developed numerous techniques and assumptions to fulfill their objective function. Even in the simple assembly lines, the number of workstations has been taken into consideration and already classified into two types [53]:

Type 1:Minimizes number of workstations for a given cycle time

Type 2:Minimizes the cycle time for a given number of workstation

Some researchers believe these two classifications can be applied into other ALB problems. Özbakır and Tapkan (2011) [17] found that the bee algorithm method in 2S-ALB has taken the Type-1 group in minimizing the number of workstations into their research objective. While Özcan and Toklu (2010) [51] also considered Type-1 objective function for their heuristic approach method.

The evaluations on minimizing the workstation number were discussed in some researches. As in Kim et al. (2000) [44] study, they have successfully determined the fitness of potential solution.

$$\text{Eval} = \sum_{j \in J} WSj \tag{1}$$

where J is the set of workstations and $0, \quad \text{if } F_j = 0,\ 1, \text{if } 0 < F_j \leq CT,\ 1 + \left(\frac{F_j}{CT} + 1\right), \text{if} F_j > CT$.

The evaluation measure (Eq. 1) intends to select more fit individual characteristics for the next generation.

The related studies that applied the objective function to minimize the number of work stations in the ALB problems have been reported in literature [4, 13–17, 36, 44, 46–48, 50–55, 59–61, 66]. However, there is one objective function that seems to be related to the above function in minimizing the workstation number (i.e., minimize mated station number) as discussed in the following section.

4.2 Minimizing the mated station number

Formally, in two-sided assembly line, there will be a pair of lines placed opposite each other such as that shown in Fig. 2. In the 2S-ALB, both sides of the lines either right or left will perform its individual task. A mated station is represented by a pair of station or workstation that faces each other [47, 48]. In some researches, it is also called as the companion [44, 47]. Therefore, the 2S-ALB minimization of mated station number is generally able to reduce the number of stations as well. Most of the researches will take the minimization of station number into consideration when assigning the minimization of the mated station number as their objective function [46, 48, 51, 52].

In Özcan and Toklu (2009) [50] study, Eq. 2 becomes the mathematical formulation model for minimizing the mated station number besides being able to assist in minimizing the number of stations or workstations.

$$\text{Minimize} = \sum_{j \in J} \left(F_j + G_j\right) + \mathscr{E}.\sum_{j \in J}.\sum_{k=1,2} U_{jk} \tag{2}$$

where

$j$mated station

$k$side of the line; $k = \{1, \text{indicates a left} \quad 2, \text{indicates a right}$

Jset of mated station; $J = \{1, 2,. .., j$

F_j1, if mated station j is utilized for both sides of the line; 0, otherwise

G_j1, if mated station j is utilized for only side of the line; 0, otherwise

$\mathscr{E}$a small positive value, $0 < \mathscr{E} \leq 1/ (2 * \text{nms} + 1)$

U_{jk}1, if stations j is utilized for only side of the line; 0, otherwise

The researches that choose the minimization of mated station number as their objective function have been reported in literature [4, 13, 16, 46, 48, 50–52]. The significant result has proven their accomplishment in the ALB studies.

4.3 Minimizing line length

Formerly, in the SALB production line, the longer and larger space are actually needed as only one side of assembly is used since 2S-ALB looks more reliable in dealing with this kind of problem. The 2S-ALB provides shorter length of line length than single-sided ALB [15]. This is due to the workstation dispensed on both sides of the assembly production systems, as shown in Fig. 2. A set of 2S-ALB assemblies will distribute

Table 2 Objective function for 2S-ALB problems

Author/s, year	Ref.	Objective Function											
		1	2	3	4	5	6	7	8	9	10	11	12
Yuan, Zhang et al. 2015	[4]	x	x										
Kucukkoc and Zhang 2015	[59]	x			x								
Kucukkoc and Zhang 2015	[60]	x											
Chiang, Urban et al. 2015	[15]	x		x									
Ağpak and Zolfaghari 2015	[53]	x		x									
Tuncel and Aydin 2014	[29]				x	x							x
Purnomo and Wee 2014	[30]										x		
Kucukkoc and Zhang 2014	[55]	x		x		x							
Kucukkoc and Zhang 2014	[54]	x		x									
Chutima and Naruemitwong 2014	[56]				x	x		x					
Purnomo, Wee et al. 2013	[12]				x								
Khorasanian, Hejazi et al. 2013	[48]	x	x			x							
Tapkan, Ozbakir et al. 2012	[61]	x											
Roshani, Fattahi et al. 2012	[49]											x	
Chutima and Chimklai 2012	[16]	x	x			x	x						
Ağpak, Yegül et al. 2012	[36]	x		x									
Taha, El-Kharbotly et al. 2011	[46]	x	x										
Özbakır and Tapkan 2011	[17]	x											
Xiaofeng, Erfei et al. 2010	[67]			x									
Özcan and Toklu 2010	[51]	x	x										
Özcan, Gökçen et al. 2010	[66]	x											
Özcan 2010	[13]	x	x										
Simaria and Vilarinho 2009	[14]	x											
Özcan and Toklu 2009	[50]	x	x										
Özcan and Toklu 2009	[52]	x	x										
Kim, Song et al. 2009	[45]				x								
Hu, Wu et al. 2008	[62]								x				
Baykasoglu and Dereli 2008	[47]	x					x						
Lee, Kim et al. 2001	[11]						x			x			
Kim, Kim et al. 2000	[44]	x											

Objective function: 1—min. number of workstation, 2—min. number of mated station, 3—min. line length, 4—min cycle time, 5—workload/task smoothness, 6—max. work relatedness, 7—min. production variance, 8—min. idle time, 9—max. slackness, 10—max. production rate, 11—cost oriented, 12—optm. specific constraints

all of the tasks in between a mated station; therefore, it allows the length of the lines to be shortened [54]. Besides, a shortened line may provide other additional benefits [15] like the following:

1. Reduces the cost of material handling
2. Able to reduce the equipment of tools and fixture by implementing tool sharing approach for opposite workstation

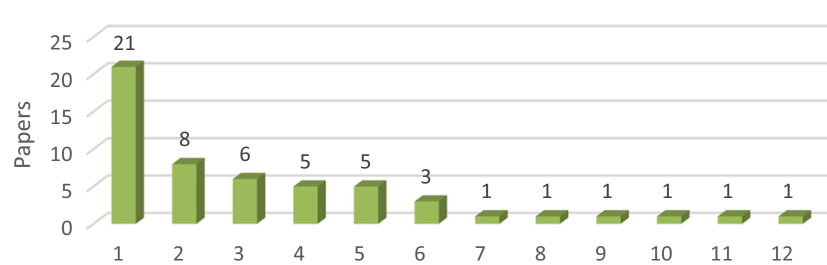


Fig. 6 Number of researches that used different objective functions in 2S-ALB

3. Reduce the overhead cost

The formulation of line length minimization was presented in a study by Urban et al. (2015) [15]. They have formulated their objective function on minimizing the line length with w_1 and w_2 as the weight-associated parameters.

$$\min_{x_{jk}}\left\{w_1{*}\max_k\left[\left(\cup_j x^j_{jk}\right)*\left[\frac{k}{2}\right]\right]+w_2{*}\sum_k\left(\cup_j x^j_{jk}\right)\right\} \quad (3)$$

where

$j=1,2,\ldots,n$ tasks

$k=1,2,\ldots,m$ station

w_1, w_2 objective function weights for the line length and the number of stations, respectively

x_{jk}assignment variable, equal to one if task j is assigned to station k; equal to zero otherwise

In some studies, the minimization of line length was also called as the minimization of position number [36, 67]. The position number actually indicates the workstation in which it will reallocate and open in a row order [53]. Ağpak and Zolfaghari also succeeded in introducing a different evaluation in minimizing the line length. The formulation is as follows:

$$\text{Minimize } Z_2=\sum_{k=1}^{K}k.\ P_k \text{ or } Z_2=\sum_{k=1}^{K}P_k \quad (4)$$

where

$k$position, $k=1,2,\ldots,K$

P_k1, if any station at position k is open; 0, otherwise

The minimization of the line length could be suggested as the additional objective function in 2S-ALB or could be the secondary objective. In certain researches, the minimization of line length was performed after the minimization of workstation or number of mated station [53, 55]. Hence, a different idea and formulation has been built to represent 2S-ALB with success.

4.4 Minimizing cycle time

According to [59], the duration of cycle time is greatly related with workstation. In ALB, the minimization number of cycle time alternatively classifies performance as a type-2 ALB problem. The relation between the number of cycle time and the number of workstations in objective function has influenced various studies throughout their researches. Cycle time is commonly defined as the maximum time to complete any task allowed on each line of workstation. Such in 2S-ALB problem which definitely has two sides, either left or right of workstation, the cycle time will be strictly set as to not exceed the limit value of the processing task time. However, in many cases, the cycle time could not be filled by the task and it has created some gap on the workstation due to some restriction. Thus, the processing task time will not be equal to the assessed cycle time. In such cases, the gap associated with a void space is naturally called idle time.

The minimization of cycle time has been discussed in some previous researches. As in [45], they have set the minimization of cycle time as the single objective function. Eq. 6 is presented as the restriction for Eq. 5 to achieve the cycle time minimization.

$$\text{Minimize ct} \quad (5)$$

$$t_i^f \le ct \quad (6)$$

where

ct cycle time

t_i^f finish time of the task i

The summation of processing and idle time were actually performed as the general operation of calculating the number of cycle time [12], and it must be equal or smaller than the actual value [11, 36], which cannot exceed the designated cycle time. The minimization of cycle time is mentioned as equivalent to maximization of the assembly line efficiency (workstation efficiency) that reduces the idle time value. Workstation efficiency (WE) in Eq. 7 is defined by the total processing time of all tasks divided into time allocated in the workstation (cycle time) [12]. In measuring the workstation efficiency, the number of cycle time is also needed and becomes a factor for calculating the efficiency value.

$$\text{WE}=\frac{\sum_{i=1}^{n}t_i}{2.\text{m.}\ \text{CT}};\quad \text{i}\in\text{I} \quad (7)$$

where

st_isetup time

$t_i$processing time

Ia set of tasks assigned to the workstation

CTcycle time

In 2S-ALB, each line, left (L) and right (R), may have different numbers of cycle times [46]. Hence, it may have different throughput rates too. In addition, the sum of processing and idle time was actually performed as the general operation of calculating the number of cycle time [12] and it must be equal or smaller than the actual value [11, 36], which cannot exceed the designated cycle time. In measuring the workstation efficiency, the number of cycle time is also needed and becomes a factor for calculating the efficiency value [12].

The cycle time formulation determined by the researchers is normally connected with some restriction. In [29, 45, 56], they have prescribed the cycle time value into an amount which could not be greater. By this, the cycle time will not be exceeded and might be dropped. However, [59] has successfully applied a strict expression which combines two objective functions, that is, the cycle time and workstation minimization. The expression has also built together certain restriction constraints.

4.5 Workload/task smoothness

For any workstation on ALB problems, the assigned workload will not be same. Considering that, between the distributed tasks in industrial problems, the processing tasks are normally not equal. The assigned workload/task to the workstation initially is unbalanced. As in [29], the main goal is to improve the line balance implemented by the company for the given cycle time. Thus, considering the workload smoothness comes as another additional aim [14, 54]. Referring to [54, 55] that minimizes weighted idle times (WITs) also means ensuring a smooth workload among the workstations. Equation 8 below shows the expression of WIT.

$$WIT = \sum_{\varphi=1}^{\phi} \sum_{h=1}^{H} \sum_{k=1}^{K_h} \sum_{x\in\{0,1\}} \left(C - \sum_{j=1}^{M_h} \sum_{i=1}^{T_{hj}} op_{hj}\, pt_{hji} Y^{\varphi}_{\text{hjikx}} \right) \quad (8)$$

where

xSide of the line; $x=0$, which indicates left side of relevant line; 1, indicates right side of relevant line

φproduction cycle ($\varphi=1,\ldots,\phi$), where $\phi=LCM(S_1,\ldots,S_H)$

Ccommon cycle time for all lines

op_{hj}overall proportion of the demand of assembled product model

pt_{hji} processing time of task of t_{hji} model m_{hj} on line L_h

$Y^{\varphi}_{\text{hjikx}}$ 1, if task t_{hji} of model m_{hj} is assigned to station W_{hkx} on side x of line $L_h$in the production cycle φ; 0, otherwise

Smoothing the workload evenly is able to balance the workstation and the assembly line. In fact, it is successfully presented in [29], which balances using the Teaching–learning-based optimization (TLBO) algorithm. The smoothness index (Eq. 9) among the workstation is calculated within ranged value. The formulation is as follows:

$$C_b = \sum_{k=1}^{K} \left[\left(\frac{I_k}{T} \right) - (1/K) \right]^2 \quad (9)$$

where

$C_b$line smoothness index

Ktotal number of workstations utilized on the line

Ttotal idle time of all workstation

I_k idle time at workstation k

Once the calculation of smoothness index C_b is done, the line efficiency is shown to improve as well. Another equation contributed to the balance workload is illustrated in [16] whereby they assigned workload plus idle time for any workstation. Moreover, a uniform distribution across open workstation trusted has the same meaning as uniform idle time distribution. Therefore, the balance workload can be calculated from the following equation:

$$\text{Minimize } B_b = \frac{N_w}{N_w-1} \sum_{k=1}^{LL} \sum_{b=L}^{R} \left(\frac{S_{kb}}{WIT} - \frac{1}{N_w} \right)^2 \quad (10)$$

where

B_b workload balance between workstations

N_w the number of operators

S_{kb} the average idle time of workstation k on side b

WIT weighted idle time

The workload balance distribution among workstation is taken as counted measure since it is recommended in [14]. The recommendation is also highlighted in [16] and has been successfully presented in B_b formulation equation in terms of balancing the workload among the workstations.

4.6 Other objective functions

Another significant objective function prescribed by the researcher besides the above five examined earlier could have a big potential. They have noticed other critical objective function that could be used and applied in optimizing the assembly line, for instance, the minimization of the production variance [56] and the minimization of idle time [62]. Besides, the maximizing work relatedness [11, 16, 47], maximizing slackness [11], maximizing production rate [30], optimization of specific constraints [29], and cost oriented [49] can also bring great influence to other research.

5 Constraint

Normally, for every research on 2S-ALB, it will consist of feasible assignment with certain restrictions. In order to acquire more sensible and effective solution, the presented studies have considered the real non-obligatory relationships between tasks in assigning them to the workstations on the assembly lines [35, 48] such as the precedence relation constraint that indicates each operational process for every assigned task. It practically could not be considered because of the influences against all of the assembly operations. However, other constraints used in the 2S-ALB as the restriction will be discussed based on its popularity. Table 3 shows the constraints considered on the previous research of the 2S-ALB problems.

Figure 7 above has illustrated the frequency for each different type of constraints in 2S-ALB. The number under the graph represents the difference between optimization method types (see legend of Table 3). Zoning constraint leads the frequency graph (Fig. 7) by 16 counts followed by 13 cycle times and operation direction constraints with only 9 counts.

5.1 Zoning constraint

Large numbers of researches on ALB problems have been considered both by the academics and industry. While in 2S-ALB problems, various types of solution approach were suggested by the researcher in solving the faced problems. In order to reach the specified objective and succeed in the studies, most of the researchers strictly applied certain restrictions or constraints. As an example, Baykasoglu and Dereli (2008) studied the minimizing of the number of workstations in the 2S-ALB problems [47]. Some restrictions and constraints were built such as zoning. In some other researches, the zoning constraint was also known as the positional constraint.

Zoning constraint actually is a preference of task to be assigned on which workstation [16] on the assembly line. Respectively, the zoning constraints are divided into two, either positive or negative, zoning [30, 53, 60]. In general, positive zoning is a restriction for assigning more than one task into a workstation, while negative zoning strictly controls to not to be assigned with any set of tasks into the same workstation. Positive zoning is usually related to the common tools

Table 3 Constraints in 2S-ALB problems

Author/s, Year	Ref.	Constraint											
		1	2	3	4	5	6	7	8	9	10	11	12
Yuan, Zhang et al. 2015	[4]			x									
Kucukkoc and Zhang 2015	[59]				x		x			x			
Kucukkoc and Zhang 2015	[60]	x	x		x								
Chiang, Urban et al. 2015	[15]		x	x								x	
Ağpak and Zolfaghari 2015	[53]	x	x										
Tuncel and Aydin 2014	[29]	x				x	x	x					
Purnomo and Wee 2014	[30]	x											
Kucukkoc and Zhang 2014	[55]	x			x								
Kucukkoc and Zhang 2014	[54]	x		x	x				x	x			
Chutima and Naruemitwong 2014	[56]												x
Purnomo, Wee et al. 2013	[12]	x	x			x	x	x	x		x		
Khorasanian, Hejazi et al. 2013	[48]							x			x		
Tapkan, Ozbakir et al. 2012	[61]	x				x							
Roshani, Fattahi et al. 2012	[49]		x	x	x								
Chutima and Chimklai 2012	[16]	x											
Ağpak, Yegül et al. 2012	[36]	x	x										
Taha, El-Kharbotly et al. 2011	[46]			x									
Özbakır and Tapkan 2011	[17]	x											
Xiaofeng, Erfei et al. 2010	[67]			x									
Özcan and Toklu 2010	[51]		x				x						
Özcan, Gökçen et al. 2010	[66]		x										
Özcan 2010	[13]		x										
Simaria and Vilarinho 2009	[14]	x		x	x	x							
Özcan and Toklu 2009	[50]	x				x							
Özcan and Toklu 2009	[52]	x											
Kim, Song et al. 2009	[45]		x						x				
Hu, Wu et al. 2008	[62]		x	x									
Baykasoglu and Dereli 2008	[47]	x											
Lee, Kim et al. 2001	[11]		x	x									
Kim, Kim et al. 2000	[44]	x	x										

Constraints: 1—zoning, 2—cycle time, 3—operation direction, 4—capacity, 5—synchronous task, 6—workstation; 7—resource, 8—occurrence, 9—assignment, 10—distance, 11—completion probability, 12—sequence dependent task time

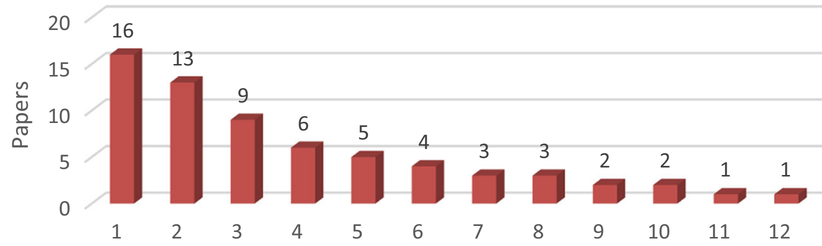


Fig. 7 Number of constraints used in the 2S-ALB reference paper

and fixture; therefore, the operational process could be assigned on the same workstation [29]. Meanwhile, negative zoning is something that is related to technology and equipment. Hence, it could not be assigned to the same workstation due to safety reasons or different required equipment [30].

There are two types of formulation in the zoning constraint, positive and negative. In studies by Simaria and Vilarinho (2009) [14] and Tapkan, Ozbakir et al. (2012) [61], they successfully performed both positive and negative zonings respectively for the same and different workstations. Equation 11 represents the positive zoning constraint where the set of tasks must be assigned at the same workstation, while Eq. 12 is the negative zoning constraint that the task must not be assigned to the same workstation.

$$\sum_K k(x_{ik1}+x_{ik2})-\sum_K k\left(x_{jk1}+x_{jk2}\right)=0 \qquad (i,j)\in ZP_{ij} \quad (11)$$

$$\sum_K k(x_{ik1}+x_{ik2})-\sum_K k\left(x_{jk1}+x_{jk2}\right)\neq 0 \qquad (i,j)\in ZN_{ij} \quad (12)$$

where

K the set of workstations ($k=1,\ldots,I$)
ZP_{ij} the set of pairs of tasks that must be assigned to the same workstation
ZN_{ij} the set of pairs of tasks that cannot be assigned to the same workstation
x_{ikb} $\{1$, if task i is assigned to workstation k at side b; 0, otherwise

However, different formulations of zoning constraint have been used by Wee et al. (2013) [12] and Yegül et al. (2012) [36] in their studies. Both formulations of Eqs. 13 and 14 were constructed based on the positive and negative zoning constraints (PZ and NZ) as well.

$$x_{gjk}-x_{ijk}=0 \qquad (g,i)\in \mathrm{PZ} \quad (13)$$

$$x_{gjk}+x_{ijk}\leq 1 \qquad (g,i)\in \mathrm{NZ} \quad (14)$$

where

x_{ijk} a decision variable
(g,i) distance between task g and i

The positive and negative formulations of the zoning constraints have been applied by numerous researchers throughout the significant ALB in different problem types [12, 14, 16, 17, 29, 30, 36, 44, 47, 50, 52–55, 60, 61].

5.2 Cycle time constraint

Another significant constraint to the line system is the duration of the entire processing time or usually called cycle time. These constraints were considered as one of the most important criteria to successfully balance the two-sided assemblies. Commonly, the cycle time is subjected for balancing purposes such as in simple ALB problems of type 1 and type 2. Both of these problems seriously considered cycle time as their objective.

Type 1:To minimize the number of workstations for a given cycle time

Type 2:To minimize the cycle time for a given number of workstations

From the two objectives above on the ALB problem, the cycle time could be best regarded as purely important either for retention (Type 1) or for reduction (Type 2).

Cycle time becomes important especially in measuring workstation efficiency (Eq. 7). For instance, it is required and becomes a factor for calculating the efficiency value [12]. Workstation efficiency (WE) in Eq. 7 is defined by the total processing time of all the tasks divided into the time allocated in the workstation (cycle time), while the response of changes on cycle time is formulated as the equation below [49]. Equation 15 is represented as the constraint that ensures the task will be finished before the cycle time ends.

$$st_i+t_i\leq CT, \qquad \mathrm{i}\forall\in \mathrm{I} \quad (15)$$

where

st_i setup time
t_i processing time

I a set of tasks assigned to the workstation
CT cycle time

The other cycle time formulation that takes idle time as the measure is shown in the equation below [12]. Equation 17 is the formulation of measuring the total of idle time in the workstation; meanwhile, the sum of processing and idle time is show in Eq. 16. The summation in Eq. 16 must be smaller than or equal to the cycle time while performing the cycle time restriction.

$$\sum_{i=1}^{n} t_i x_{ijk} + s_{jk} \le CT \tag{16}$$

$$s_{jk} = \sum_{u=1}^{U} x_{ujk}\left(t_{u+1}^{s} - t_{u}^{f}\right) + \left(CT - t_{u}^{f}\right) \quad \mathrm{u} \in Q_{jk} \tag{17}$$

where

t_i processing time for task i
x_{ijk} a decision variable
s_{jk} total idle time at workstation j, side k
t_u^s the starting time of task u
t_u^f the finishing time of task u
CT cycle time
Q_{jk} a set of task that is assigned in workstation j side k

Recently, the cycle time constraint has become imperative to researchers in balancing purposes [11, 36, 44, 45, 51]. The high recommendations in every future study impacts the diverse utilization of the formulation. These are subjected to the different types of ALB problems that brought different ideas by different researchers in calculating the cycle time.

5.3 Operation direction constraint

A feasible balance line in 2S-ALB will be assigned with preferred sides of the line [15]. Therefore, allocating the task to the preferred workstation becomes crucial. This is the most challenging issue before completely running the assembly line throughout the desired task due to the 2S-ALB problems which are already categorized into three groups: left side (L), right side (R), and either sides (E) of the line [46, 49]. For this reason, the selection of the side was studied by the researcher. These are commonly called the operation direction constraint, and it should be fulfilled by the relations between every task. For example, the automotive assembly line which consists of the two-sided assembly operation. The left side usually will perform the task which prefers the left-hand handling, while the right side will perform the right-handed task. However, there are some tasks that do not have any preferred operation direction [49]. Hence, the proper selection of (left, right, or either) sides was greatly studied by researchers for optimization purposes.

The three equations below are the rule and formulation in selecting the preferred side (left, right, or either side) of the 2S-ALB problems [67]. The first equation, Eq. 18, will perform either side as the selection after calculating the total processing task t_i and t_j (cycle time). Then, Eqs. 19 and 20 will perform the left or right side of the selection after either side is filled (Eq. 18).

$$D(i) = E, \ \text{if} \quad t_i + t_j > C \quad \forall j \in CTI_i, \tag{18}$$

then t_i is increased to C;

$$D(i) = L, \quad \text{if} \quad t_i + t_j > C \quad \forall j \in CTI_i, \mathrm{D}(j) \in \{L, E\}, \tag{19}$$

then t_i is increased to C;

$$D(i) = R, \quad \text{if} \quad t_i + t_j > C \quad \forall j \in CTI_i, \mathrm{D}(j) \in \{R, E\}, \tag{20}$$

then t_i is increased to C;

where

C cycle time
i, j task number
t_i processing time of task i
t_j processing time of task j
$D(i)$ operation direction of task i
$D(j)$ operation direction of task j

The operation direction constraint has been formulated with a definite purpose to allocate the preferred workstation whether left, right, or either side of the assembly line. Another research by Urban et al. (2015) also formulated its operation direction constraints in three main rules [15]. Each task will be assigned to only one station either left or right, starting with the left side. In Eq. 21, which is for the left operation side, the formulation is labeled with odd numbers $(1, 3, 5, \ldots, m-1)$ and even numbers $(2, 4, 6, \ldots, m)$ for the right-sided operation (Eq. 22). Therefore, Eq. 23 will be choosing either side of the assembly after both sides are filled.

$$\sum_{k \in \left\{1,3,5,\ldots,m-1\right.} x_{jk} = 1 \ \ \forall j \in L \tag{21}$$

$$\sum_{k \in \{2,4,6,\ldots,m\}} x_{jk} = 1 \ \ \forall j \in R \tag{22}$$

$$\sum_{k=1}^{m} x_{jk} = 1 \quad \forall j \in E \tag{23}$$

where

x_{jk} assignment variable, equal to one if task j is assigned to station k, equal to zero otherwise

j 1, 2, …, n tasks
k 1, 2, …, m stations

The researchers who have taken the operation direction constraint into consideration significantly have been reported in the literature [4, 11, 15, 46, 49, 55, 62, 67]. Most of the studies that considered operation direction into their constraint from general ALB problem were due to the additional line to the assemblies.

5.4 Other constraints

Some other significant constraints that were used other than those reviewed above also possess their own abilities and advantages such as the capacity constraints that are commonly used in the line balancing problems, and they need to be satisfied. Commonly, the capacity constraint is developed by the total processing time of the assigned tasks to the workstation. If the next sequence task does not satisfy the restriction of the capacity constraint, a new workstation will be opened for the next assigned task [59, 60]. The capacity constraint also will ensure that the execution of each task is within the cycle time [55]. Other considered studies on capacity constraint are reviewed in other literature [14, 49, 54].

Besides that, the workstation constraints are also considered by some of the researchers in the 2S-ALB problems as the restriction. This constraint means, for each specific task, it will be assigned to a specific workstation. Therefore, the assigned task is strictly for a workstation where the task is really required [12, 86]. In a study by Tuncel and Aydin (2014), they have associated the workstation with particular equipment and material for the assembly operation. Thus, it also means a specific task could only be assigned to a certain and required workstation [29]. By this, workstation constraint seems absolutely essential in all ALB problem optimizations.

The assignment constraint to determine which task/assignment could be assigned in which location of the workstation was also studied. It also determines the duration of time in which the assignment must be executed at the same workstation when the current side task is lower than the opposite side task [54]. The assignment constraint could also ensure that each task will be assigned exactly once in completing the 2S-ALB operation [59]. The mated station in 2S-ALB always becomes another factor that may affect the completion probability. The completion probability constrain is constantly related with time. It is necessary for a mated station to complete the task given within the cycle time [15]. Therefore, the completion of time distribution must be determined explicitly in 2S-ALB for optimization purposes.

The synchronism of task in the single-sided ALB may not be very important, but in 2S-ALB, it is different. If a synchronization constrain is considered in 2S-ALB, the task will be divided to satisfy the synchronization constrain [29]. The identical task time for every mated station will perform the synchronism working experience, and it is only presented in 2S-ALB operating line. By doing this, the idle time of product processing will be minimized [12]. Other considered researches on the synchronization constraint are [14, 50, 61]. Another remarkable constraint in the 2S-ALB problems is the sequence-dependent task time. The existence of this type of constraint is to allocate each task into a workstation based on the preferred operational directions and precedence relationships [56]. The mated station factor could also affect the performance in the 2S-ALB; hence, this sequence-dependent task time constraint will allocate the task within the limited setup time.

The distance between tasks is formally able to become as a constraint, due to the important and affected factor regarding space and cycle time. Distance could be measured by the length of space, duration of time, and even workstation position, from the initial task to the next preferred task [12, 48]. Distance constraint will be set to the maximum or minimum with the aim to get prepared for the next task. For example, a painting process that needs a high duration of time to dry. Therefore, maximizing the distance will come into a way of preparing the paint before further tasks are done.

Different ALB operation requires different machines and tools. These will turn into an issue in ALB. Any equipment and tools for conducting a task are considered as a resource. Some researchers take resources as a restriction and consider them as constraints. Resource constraints might be in many forms such as space and operator [29]. An operator could be a resource to take the action to conduct the assembly operation. In Wee et al. (2013), the resource restriction is believed to be able to reveal the inadequate space for allocating the required machines on a workstation [12]. Meanwhile, several studies have highlighted the occurrence constraint in 2S-ALB. Occurrence constraint is to ensure that every task is assigned just only for one workstation [12, 45]. The researchers come with some formulation to control and act as desired for the optimization purposes.

Literally, different constraints actually come with a certain objective function. To meet the desired objective function, it needs a strict condition of constraint as a constraint will be set according to the considered objective function. For instance, the minimization number of workstations and the minimization of line length [61, 67] are also affected. The methods of choosing constraints are definitely different from one another. This is because the constraints will act as a strict condition to acquire the objective function value. Hence, the selection of a suitable constrain is seriously needed.

Besides, for more complex objective function, the constraint is significantly complex too. In [15, 53], the assessed objective function is presented in biobjective which aims at different targets. Although both studies searched for a similar objective function, the methods of choosing constrains are still quite different. Each study has provided their own way to specify how each selected constraint should behave in meeting the needs of objective function. Moreover, every constraint lives with a strength indicating how important it is to satisfy the considered objective function. A single constraint might still be a weak restriction to enforce the direction of the choosing objective function. Therefore, by picking a different constraint, it will ensure that the constraint becomes stronger and influences the other constraints to provide a good result in the objective function.

6 Discussion and research potential

Assembly line balancing (ALB) is a production planning problem concerned with allocating each task to the workstations on the assembly line. The purpose is to balance and optimize the assembly line to increase the productivity measure. Various methods have been established with different useful optimization techniques, and it has been expending a lot with new improvement and additional capability of solving methods. In identifying the research issue, four main specifications have been summarized and discussed accordingly. Considering the growing number of publication in solving ALB problems, this study was focusing only on the two-sided assembly line balancing (2S-ALB) problems because of its benefit and priority in solving large-manufactured products. The researches which deal with 2S-ALB problems on the algorithm optimization are classified as the NP-hard problems. However, as time goes by, more complicated types of problems appear. A different solving approach of different complex problems was presented. The complex combinatorial problems between two or three or even more problems were vigorously studied by the researcher to diversify the problem types and inspire other researchers for future studies. However, there are still a few unfilled potential and gaps through the ALB problem studies.

In ALB, there are two problem types which are simple and general that have been questioned by the researcher. Simple ALB is quite famous and extremely studied because of its ability to balance with only a single side of the operation line. Besides, it is able to optimize the operation line by simply allocating for the workstation without much hesitation. However, in manufacturing industries nowadays, a complex combination of problems is introduced. Because of the market and customer needs, the researcher has been extremely dedicated to the invention of a new ALB combinatorial problems, known as the general ALB, such as in Chutima and Chimklai (2012) and Zhang et al. (2015), who presented the combination of mixed-model and two-sided problems in ALB [4, 16]. Meanwhile, Kucukkoc and Zhang (2014) introduced the mixed-model parallel two-sided assembly line balancing problem [54, 55]. Figure 8 presents the problem combination graphs of 2S-ALB that have been considered by researchers since 2008. These absolutely change the assembly configuration but successfully provide more beneficial advantages. From these, it was noted that researchers were more interested in combining the 2S-ALB with the other ALB models. Hence, this paper focused on the study of 2S-ALB problems.

From most of the published works, the genetic algorithm (GA) has successfully dominated the 2S-ALB problem. As presented in Fig. 5, GA is found competitive in solving the different types of combinatorial ALB problems. The successfulness of the GA method in solving complex problems is well known, with the presented control parameter, such as population size and crossover probability. Besides GA, the popularity in solving the complex combinatorial 2S-ALB problem is followed by the ant colony optimization (ACO) algorithm method. The pheromone trail of the ant path by the ACO method became the idea to find the solution and solving method. The simulated annealing (SA) followed after both the above methods. Even so, numerous other metaheuristics such as bee algorithm (BA) and particle swarm optimization (PSO)

Fig. 8 2S-ALB problem combination trend by year

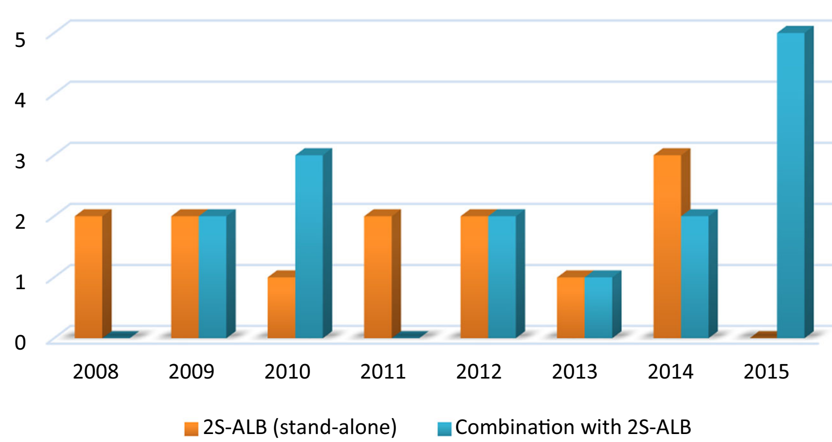
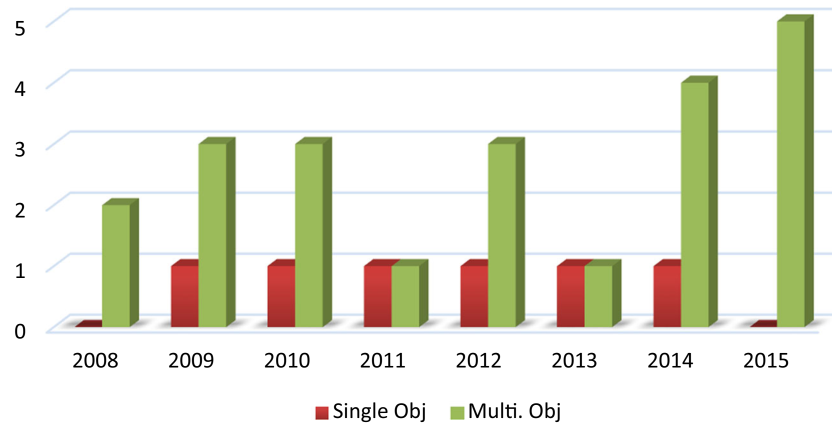


Fig. 9 The trend of 2S-ALB multiobjective by year



greatly had a big potential in solving the 2S-ALB combinatorial problems. From the assembly line balancing perspective, we also noted that the implementation of advanced computational method such as metaheuristic approaches are broadly studied compared with the heuristic, a problem-dependent solution approach.

Another issue that has been highlighted by researchers is the objective function consideration. It is a necessity for an ALB optimization research to indicate and point the objective function itself. This precisely will make sure the direction of the optimization research study. Most of the literature considers either maximizing or minimizing the appropriate task in fully optimizing the assembly line. Figure 6 illustrates the trend of objective function considered by researchers. Practically, in 2S-ALB, the majority considered the minimization number of workstation as the main objective. It is followed by the minimization of mated station number. The minimization number of workstation always looks significantly able to optimize the assembly line; however, it is not that simple. In other cases, the minimization of the number of workstations also might delay certain processes that would seriously reduce the productivity. By this, the automation and integration of assembly optimization has potential. Nowadays, a multiobjective function is widely studied, rather than a single-objective function. Figure 9 illustrates the trend of the multiobjectives that have been studied. This is due to the modern manufacturing system which always tries to fulfill the demands.

Besides that, in filling the needs of the objective function and plant configuration, some constraints will be counted. Therefore, normally, in every research of 2S-ALB, it will consist of certain restrictions. The considered constraint among the 2S-ALB researchers is shown in Fig. 7. The trend is to start with the zoning constraint, followed by cycle time and operation direction constraint. With this restriction on every constraint, it will effectively be presented by a certain rule or mathematical formulation. Because of the existence of a complex combination problem type with different designs of plant configuration, various constraints have been introduced recently. The presented constraints will act as the restriction to completely acquire the main objective of the research.

In dealing with various types of complex combination problems, various approaches of the optimization method and objective have been established. This is because the ALB problems are getting more complicated day after day. Hence, the growing of 2S-ALB researches in solving the manufacturing issues shall be supported by the manufacturing industry itself. Besides, the researcher might be able to manage and introduce more combinations or hybridized optimization methods on the ALB problems.

Since 2S-ALB is classified as an NP-hard combinatorial optimization problem, it consequently possesses a big concern to the researcher in finding the best solution. NP in fact stands for nondeterministic polynomial time, commonly informed as the hardest problem to solve and needs a large amount of time [44, 54]. Therefore, nowadays, the ALB problem has received widespread attention among researchers and industrial practitioners. At once, the hybridization technique between two algorithms successfully minimizes the computational time and makes the solving period faster. Most of the time, none of the researchers have guaranteed an optimal solution for the ALB problems, but they always relatively offer a good solution in reducing the computing period. Therefore, different solving approaches have been found, including metaheuristics, as the deficiency of the possible solving method should be considered for the enhancement as the potential research study.

7 Conclusions

Lately, the studies on ALB problems are growing very rapidly if compared to the previous decades, and most of the studies are focusing on the manufacturing industries. Industries, especially those manufacturing large-sized products, like cars, busses, or trucks, always need to keep improving, and the productivity will act as the measure. Thus, one of the ways to increase the company profit is by optimizing the assembly line. It is crucially important, especially in the assembly of large-sized products. In order to develop and optimize the ALB problems that are becoming complex day after day, different methods and solution approaches should be presented. There are still plenty of chances even in the algorithm development which is not being exposed yet.

In future research direction, various kinds of improvements that are precisely able to modify and develop the ALB problems are needed. The combination of certain general assembly lines are found to be competitive for being the best on making a closer model to the actual industrial circumstances. Besides that, the hybridization of an algorithm method also seems to be able to improve the assembly line problem. This method should also be implemented in 2S-ALB problems which formerly have been applied to the simple ALB problem. The potential of other heuristic or metaheuristic solving approaches also require a lot of attention. This can provide more alternative ways and methods in solving the ALB problems and thus able to hybridize and compare the effectiveness of the new algorithm method. Another direction should be focused to solve and optimize the customized problem/combination problem because the assembly line/production becomes more complex. Besides, the importance of automation is seen to be crucial. Further study related to ALB and automation is believed to be able to help to improve and provide more confidence to the industrial practitioner.

Acknowledgments The authors would like to acknowledge the Ministry of Higher Education, Malaysia and Universiti Malaysia Pahang for supporting this research under the FRGS grant RDU140103.

References

1. Becker C, Scholl A (2006) A survey on problems and methods in generalized assembly line balancing. Eur J Oper Res 168(3):694–715
2. Lam NT et al. (2016) Lean line balancing for an electronics assembly line. Procedia CIRP 40:437–442
3. Caggiano A, Marzano A, Teti R (2016) Resource efficient configuration of an aircraft assembly line. Procedia CIRP 41:236–241
4. Yuan B et al. (2015) An effective hybrid honey bee mating optimization algorithm for balancing mixed-model two-sided assembly lines. Comput Oper Res 53:32–41
5. Boysen N, Fliedner M, Scholl A (2007) A classification of assembly line balancing problems. Eur J Oper Res 183(2):674–693
6. Rashid MFF, Hutabarat W, Tiwari A (2011) A review on assembly sequence planning and assembly line balancing optimisation using soft computing approaches. Int J Adv Manuf Technol 59(1):335–349
7. Scholl A, Becker C (2006) State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. Eur J Oper Res 168(3):666–693
8. Erel E, Sarin SC (1998) A survey of the assembly line balancing procedures. Prod Plan Control 9(5):414–434
9. Sivasankaran P, Shahabudeen P (2014) Literature review of assembly line balancing problems. Int J Adv Manuf Technol 73(9):1665–1694
10. Kriengkorakot N, Pianthong N (2007) The assembly line balancing problem. KKU Eng J 34(2):133–140
11. Lee TO, Kim Y, Kim YK (2001) Two-sided assembly line balancing to maximize work relatedness and slackness. Comput Ind Eng 40(3):273–292
12. Purnomo HD, Wee H-M, Rau H (2013) Two-sided assembly lines balancing with assignment restrictions. Math Comput Model 57(1–2):189–199
13. Özcan U (2010) Balancing stochastic two-sided assembly lines: a chance-constrained, piecewise-linear, mixed integer program and a simulated annealing algorithm. Eur J Oper Res 205(1):81–97
14. Simaria AS, Vilarinho PM (2009) 2-ANTBAL: an ant colony optimisation algorithm for balancing two-sided assembly lines. Comput Ind Eng 56(2):489–506
15. Chiang W-C, Urban TL, Luo C (2015) Balancing stochastic two-sided assembly lines. Int J Prod Res:1–19
16. Chutima P, Chimklai P (2012) Multi-objective two-sided mixed-model assembly line balancing using particle swarm optimisation with negative knowledge. Comput Ind Eng 62(1):39–55
17. Özbakır L, Tapkan P (2011) Bee colony intelligence in zone constrained two-sided assembly line balancing problem. Expert Syst Appl 38(9):11947–11957
18. Boysen N, Fliedner M, Scholl A (2009) Sequencing mixed-model assembly lines: survey, classification and model critique. Eur J Oper Res 192(2):349–373
19. Sotskov YN et al. (2015) Enumerations and stability analysis of feasible and optimal line balances for simple assembly lines. Comput Ind Eng 90:241–258
20. Baykasoglu A (2006) Multi-rule multi-objective simulated annealing algorithm for straight and U type assembly line balancing problems. J Intell Manuf 17(2):217–232
21. Sungur B, Yavuz Y (2015) Assembly line balancing with hierarchical worker assignment. J Manuf Syst 37(Part 1):290–298
22. Esmaeilbeigi R, Naderi B, Charkhgard P (2015) The type E simple assembly line balancing problem: a mixed integer linear programming formulation. Comput Oper Res 64:168–177
23. García-Villoria A, Pastor R (2013) Erratum to "a solution procedure for type E simple assembly line balancing problem". Comput Ind Eng 66(1):201–202
24. Zacharia PT, Nearchou AC (2013) A meta-heuristic algorithm for the fuzzy assembly line balancing type-E problem. Comput Oper Res 40(12):3033–3044
25. Wei N-C, Chao IM (2011) A solution procedure for type E simple assembly line balancing problem. Comput Ind Eng 61(3):824–830
26. Micieta, B. and V. Stollomann, (2011) Assembly Line Balancing. DAAAM International p 257–264
27. Pearce, B.W. (2015) A Study on General Line Balancing Modeling Method and Techniques. Clemson University p 239
28. Tapkan P, Özbakır L, Baykasoğlu A (2016) Bee algorithms for parallel two-sided assembly line balancing problem with walking times. Appl Soft Comput 39:275–291
29. Tuncel G, Aydin D (2014) Two-sided assembly line balancing using teaching–learning based optimization algorithm. Comput Ind Eng 74:291–299
30. Purnomo HD, Wee H-M (2014) Maximizing production rate and workload balancing in a two-sided assembly line using harmony search. Comput Ind Eng 76:222–230
31. Sepahi, A. and S.G.J. Naini, Two sided assembly line balancing problem with parallel performing properties. Applied Mathematical Modelling, 2016.
32. Simaria AS, Vilarinho PM (2004) A genetic algorithm based approach to the mixed-model assembly line balancing problem of type II. Comput Ind Eng 47(4):391–407
33. Buyukozkan K et al. (2016) Lexicographic bottleneck mixed-model assembly line balancing problem: artificial bee colony and

tabu search approaches with optimised parameters. Expert Syst Appl 50:151–166
34. Vilarinho PM, Simaria AS (2002) A two-stage heuristic method for balancing mixed-model assembly lines with parallel workstations. Int J Prod Res 40(6):1405–1420
35. Merengo C, Nava F, Pozzetti A (1999) Balancing and sequencing manual mixed-model assembly lines. Int J Prod Res 37(12):2835–2860
36. Ağpak K, Yegül MF, Gökçen H (2012) Two-sided U-type assembly line balancing problem. Int J Prod Res 50(18):5035–5047
37. Kucukkoc I, Zhang DZ (2015) Balancing of parallel U-shaped assembly lines. Comput Oper Res 64:233–244
38. Alavidoost MH, Babazadeh H, Sayyari ST (2016) An interactive fuzzy programming approach for bi-objective straight and U-shaped assembly line balancing problem. Appl Soft Comput 40: 221–235
39. Ogan D, Azizoglu M (2015) A branch and bound method for the line balancing problem in U-shaped assembly lines with equipment requirements. J Manuf Syst 36:46–54
40. Hazır Ö, Dolgui A (2015) A decomposition based solution algorithm for U-type assembly line balancing with interval data. Comput Oper Res 59:126–131
41. Mosadegh H, Zandieh M, Ghomi SMTF (2012) Simultaneous solving of balancing and sequencing problems with station-dependent assembly times for mixed-model assembly lines. Appl Soft Comput 12(4):1359–1370
42. Manavizadeh N et al. (2013) A simulated annealing algorithm for a mixed model assembly U-line balancing type-I problem considering human efficiency and just-in-time approach. Comput Ind Eng 64(2):669–685
43. Bartholdi JJ (1993) Balancing two-sided assembly lines: a case study. Int J Prod Res 31(10):2447–2461
44. Kim YK, Kim Y, Kim YJ (2000) Two-sided assembly line balancing: a genetic algorithm approach. Prod Plan Control 11(1): 44–53
45. Kim YK, Song WS, Kim JH (2009) A mathematical model and a genetic algorithm for two-sided assembly line balancing. Comput Oper Res 36(3):853–865
46. Taha RB et al. (2011) A genetic algorithm for solving two-sided assembly line balancing problems. Ain Shams Eng J 2(3–4):227–240
47. Baykasoglu A, Dereli T (2008) Two-sided assembly line balancing using an ant-colony-based heuristic. Int J Adv Manuf Technol 36(5–6):582–588
48. Khorasanian D, Hejazi SR, Moslehi G (2013) Two-sided assembly line balancing considering the relationships between tasks. Comput Ind Eng 66(4):1096–1105
49. Roshani A et al. (2012) Cost-oriented two-sided assembly line balancing problem: a simulated annealing approach. Int J Comput Integr Manuf 25(8):689–715
50. Özcan U, Toklu B (2009) Balancing of mixed-model two-sided assembly lines. Comput Ind Eng 57(1):217–227
51. Özcan U, Toklu B (2010) Balancing two-sided assembly lines with sequence-dependent setup times. Int J Prod Res 48(18):5363–5383
52. Özcan U, Toklu B (2009) Multiple-criteria decision-making in two-sided assembly line balancing: a goal programming and a fuzzy goal programming models. Comput Oper Res 36(6):1955–1965
53. Ağpak K, Zolfaghari S (2015) Mathematical models for parallel two-sided assembly line balancing problems and extensions. Int J Prod Res 53(4):1242–1254
54. Kucukkoc I, Zhang DZ (2014) Simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines. Int J Prod Res 52(12):3665–3687
55. Kucukkoc I, Zhang DZ (2014) Mathematical model and agent based solution approach for the simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines. Int J Prod Econ 158:314–333
56. Chutima P, Naruemitwong W (2014) A Pareto biogeography-based optimisation for multi-objective two-sided assembly line sequencing problems with a learning effect. Comput Ind Eng 69:89–104
57. Yang X-S (2010) Engineering optimization an introduction with metaheuristic applications. John Wiley & Sons, Inc, Hoboken
58. Kucukkoc I, Karaoglan AD, Yaman R (2013) Using response surface design to determine the optimal parameters of genetic algorithm and a case study. Int J Prod Res 51(17):5039–5054
59. Kucukkoc I, Zhang DZ (2015) Type-E parallel two-sided assembly line balancing problem: mathematical model and ant colony optimisation based approach with optimised parameters. Comput Ind Eng 84:56–69
60. Kucukkoc I, Zhang DZ (2015) A mathematical model and genetic algorithm-based approach for parallel two-sided assembly line balancing problem. Prod Plan Control 26(11):874–894
61. Tapkan P, Ozbakir L, Baykasoglu A (2012) Modeling and solving constrained two-sided assembly line balancing problem via bee algorithms. Appl Soft Comput 12(11):3343–3355
62. Hu X, Wu E, Jin Y (2008) A station-oriented enumerative algorithm for two-sided assembly line balancing. Eur J Oper Res 186(1):435–440
63. Ding F-Y, Zhu J, Sun H (2006) Comparing two weighted approaches for sequencing mixed-model assembly lines with multiple objectives. Int J Prod Econ 102(1):108–131
64. Gökçen H, Ağpak K, Benzer R (2006) Balancing of parallel assembly lines. Int J Prod Econ 103(2):600–609
65. Çerçioğlu H et al. (2009) A simulated annealing approach for parallel assembly line balancing problem. J Fac Eng Archit Gazi Univ 24(2):331–341
66. Özcan U, Gökçen H, Toklu B (2010) Balancing parallel two-sided assembly lines. Int J Prod Res 48(16):4767–4784
67. Xiaofeng H et al. (2010) A branch-and-bound algorithm to minimize the line length of a two-sided assembly line. Eur J Oper Res 206(3):703–707
68. Li, S.-x. and H.-b. Shan. GSSA and ACO for assembly sequence planning: a comparative study. in Automation and Logistics, 2008. ICAL 2008. IEEE International Conference on. 2008.
69. Shan, H., et al. (2006) Genetic simulated annealing algorithm-based assembly sequence planning. in Technology and Innovation Conference. ITIC 2006. International 2006.
70. Moon I, Logendran R, Lee J (2009) Integrated assembly line balancing with resource restrictions. Int J Prod Res 47(19):5525–5541
71. Tasan SO, Tunali S (2007) A review of the current applications of genetic algorithms in assembly line balancing. J Intell Manuf 19(1): 49–69
72. Dorigo, M. and C. Blum, Ant colony optimization theory: a survey. Theor Comput Sci, 2005. 344(2–3): p. 243–278.
73. Zhang, Z.-q., et al. (2007) Ant algorithm with summation rules for assembly line balancing problem. In Management Science and Engineering. ICMSE 2007. International Conference on. 200
74. Spinellis DD, Papadopoulos CT (2000) A simulated annealing approach for buffer allocation in reliable production lines. Ann Oper Res 93(1):373–384
75. Chen S-F, Liu Y-J (2001) An adaptive genetic assembly-sequence planner. Int J Comput Integr Manuf 14(5):489–500
76. Sabuncuoglu I, Erel E, Tanyer M (2000) Assembly line balancing using genetic algorithms. J Intell Manuf 11(3):295–310
77. Zhao YZ, de Souza R (2000) Genetic production line-balancing for the hard disk drive industry. Int J Adv Manuf Technol 16(4):297–302
78. Gonçalves JF, de Almeida JR (2002) A hybrid genetic algorithm for assembly line balancing. J Heuristics 8(6):629–642
79. Wang JF, Liu JH, Zhong YF (2005) A novel ant colony algorithm for assembly sequence planning. Int J Adv Manuf Technol 25(11): 1137–1143

80. Shuang B, Chen J, Li Z (2008) Microrobot based micro-assembly sequence planning with hybrid ant colony algorithm. Int J Adv Manuf Technol 38(11):1227–1235
81. Baykasoglu, A., et al. (2009) Balancing parallel assembly lines via ant colony optimization. In Computers and Industrial Engineering. CIE 2009. International Conference on. 2009
82. Aydin ME, Fogarty TC (2004) A distributed evolutionary simulated annealing algorithm for combinatorial optimisation problems. J Heuristics 10(3):269–292
83. Mahfoud SW, Goldberg DE (1995) Parallel recombinative simulated annealing: a genetic algorithm. Parallel Comput 21(1):1–28
84. Kennedy, J. and R. Eberhart. (1995) Particle swarm optimization. in neural networks. Proceedings, IEEE International Conference on. 1995
85. Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11(4):341–359
86. Lapierre SD, Ruiz AB (2004) Balancing assembly lines: an industrial case study. J Oper Res Soc 55(6):589–597