An Artificial Bee Colony Algorithm for Assembly Line Worker Assignment and Balancing Problem

Aslihan Karas Celik and Feristah Ozcelik
Department of Industrial Engineering
Eskişehir Osmangazi University
Eskişehir, Turkey
aslihan.karas@ogu.edu.tr, fdurmaz@ogu.edu.tr

Abstract

In the assembly line balancing (ALB) literature, it is seen that the fact that the task processing times vary according to the worker is generally ignored. However, it is known that in real life, worker performances can vary depending on the skills and experience of the worker. Assembly line worker assignment and balancing (ALWAB) problem is a type of ALB problem and addresses the assignment of both tasks and workers to workstations. In this problem, contrary to the assumption that the processing times of the tasks are independent of the worker, each worker has a unique processing time for each task. This study proposes an artificial bee colony (ABC) algorithm to tackle the considered problem. Computational tests are conducted using benchmark problems from the literature and the results show that the presented algorithm achieves promising results.

Keywords
Assembly Lines, Assembly Line Balancing, Worker Assignment, Mathematical Modeling and Artificial Bee Colony Algorithm.

1. Introduction
Assembly lines are components of mass production systems in which parts are assembled by transferring them between workstations. At each workstation, the assembly tasks assigned to the corresponding station are performed, and the assembled parts are left the line as a finished product after the final workstation completes the assigned tasks.

The performance of assembly lines significantly affects the overall performance of the system. In this direction, it is critical that these lines are balanced. The problem of assembly line balancing (ALB) arises at the point of optimizing one or more performance criteria in such a way that the workload at the stations is distributed as evenly as possible, ensuring certain constraints.

In the ALB problem, only the assignment of tasks to workstations is carried out without taking into account the capabilities of the workers. However, the assumption that workers have the same performance and, accordingly, the processing times of assembly tasks are fixed for each worker is not realistic.

In the line balancing literature, differences in skills and experience of workers have started to be considered with the emergence of the assembly line worker assignment and balancing (ALWAB) problem. In this problem, the processing times of the workers differ and, in addition to the assignment of tasks to workstations, workers with different performances are also allocated to the stations where they will perform tasks.

The ALWAB problem can be divided into four groups: type-1, type-2, type-E, and type-F. In the type-1 problem, the number of workstations for a given cycle time is minimized. Type-2, in contrast to the previous one, aims to find the minimum cycle time under the given number of workstations. In the type-E problem, the cycle time and the number of workstations are minimized simultaneously. Type-F investigates whether a feasible solution exists for the given combination of cycle time and the number of workstations (Karas and Ozcelik 2021).

For the ALWAB problem, which occurs by adding the worker assignment problem to the line balancing problem, researchers propose various exact and approximate solution methods.

In this study, the ALWAB type-2 problem for serial and single-model assembly lines is discussed. An artificial bee colony (ABC) algorithm, a meta-heuristic solution method, is proposed to deal with the considered NP-hard problem. The performance of the algorithm was evaluated by using the benchmark instances of different dimensions in the literature.

2. The ALWAB Problem and Literature Review

In the ALWAB problem, it is taken into consideration that the processing times vary according to the assigned worker and some workers do not have the ability to perform some tasks. After the problem was proposed by Miralles et al. (2007), Miralles et al. (2008) implemented an application for sheltered work centers. In the study, an integer programming model and a branch and bound algorithm are developed. They also presented a branch and bound-based heuristic approach for the near-optimal solution to the problem. Table 1 represent the notation of ALWAB type-2 problem. The assumptions made by Miralles et al. (2008) and the mathematical model of the ALWAB type-2 problem are as follows:

- Processing times of tasks are deterministic and known.
- Precedence relations among tasks are deterministic and known.
- Single-model production is carried out on the assembly line.
- A serial-paced assembly line without buffer stocks is considered.
- The processing time of each task depends on the ability of each worker.
- Some workers may or may not have the ability to execute some tasks.
- Each worker can be assigned to a single workstation.
- Each task can be assigned to a single workstation, ensuring that the worker allocated to this station is able to execute the task and that precedence relations among tasks are provided.

Table 1. Notations of ALWAB type-2 problem

i, j: task
h: worker
s: workstation
N: the set of tasks
H: the set of workers
S: the set of workstations
P_i: the set of immediate predecessors of task i in the precedence network
t_{hi}: processing time of task i in terms of worker h
M: a sufficiently large constant ($\sum_{h \in H} \sum_{i \in N} t_{hi} < M$)
x_{shi}: the binary variable which is equal to 1 if task i is assigned to worker h in workstation s
y_{sh}: the binary variable which is equal to 1 if worker h is assigned to workstation s
CT: cycle time

$$min\ z = CT \tag{1}$$

subject to:

$$\sum_{h \in H} \sum_{s \in s} x_{shi} = 1 \quad \forall\, i \in N \tag{2}$$

$$\sum_{s \in S} y_{sh} \ \leq 1 \quad \forall\, h \in H \tag{3}$$

$$\sum_{h \in H} y_{sh} \ \leq 1 \quad \forall\, s \in S \tag{4}$$

$$\sum_{h \in H} \sum_{s \in s} s.x_{shi} \ \leq \ \sum_{h \in H} \sum_{s \in S} s.x_{shj} \ \forall\, i, j \in N \mid i \in P_j \tag{5}$$

$$\sum_{i \in N} t_{hi} . x_{shi} \leq CT \quad \forall\, s \in S, h \in H \tag{6}$$

$$\sum_{i \in N} x_{shi} \leq M . y_{sh} \quad \forall\, s \in S, h \in H \tag{7}$$

$$x_{shi} \in [0,1] \quad \forall\, s \in S, h \in H, i \in N \tag{8}$$

$$y_{sh} \in [0,1] \quad \forall\, s \in S, h \in H \tag{9}$$

$$CT \geq 0 \tag{10}$$

The objective function (1) of the model is the minimization of the cycle time. Constraint (2) indicates that each task is assigned to a single workstation and a single worker. Constraint (3) ensures that each worker is assigned to only one workstation. Constraint (4) controls that only one worker is allocated to each workstation. Constraint (5) defines precedence relations among tasks. Constraints (6) and (7) indicate that the worker of each station can accomplish more than one task during the cycle time. Constraints (8)-(10) reflect the variable domains.

It is worth noting that most of the researchers aimed to minimize the cycle time when the number of workstations is known (type 2) under the assumption of single model production on a serial line. Chaves et al. (2007) solved the problem by clustering search approach. Chaves et al. (2009) hybridized the clustering search method with iterative local search. Moreira and Costa (2009) updated some constraints of the problem and developed a tabu search algorithm. Blum and Miralles (2011) solved the problem by means of an algorithm based on beam search. Moreira et al. (2012) presented a heuristic approach based on the assignment priority rules and hybridized these rules with a genetic algorithm. Mutlu et al. (2013) tackled the problem by developing an iterative genetic algorithm. Vilà and Pereira (2014) investigated the relationship between the ALWAB type-2 problem and some other problems and derived new lower bounds. In the study, an exact enumeration procedure, which is based on the branch, bound, and remember algorithm was proposed. Borba and Ritt (2014) presented a mixed integer programming model including continuity constraints. A heuristic algorithm based on beam search, and a task-oriented branch and bound procedure are also proposed in the study. Polat et al. (2016) developed a two-phase variable neighborhood search algorithm and Akyol and Baykasoglu (2019a) presented a multiple-rule based constructive randomized search algorithm with the aim of solving the ALWAB type-2 problem.

Several researchers also studied different variants of the ALWAB problem. Janardhanan et al. (2017) and Janardhanan et al. (2019) studied the two-sided ALWAB type-2 problem. Oksuz et al. (2017), Zhang et al. (2019) and Zhang et al. (2020) considered the problem within the framework of U-type lines. While Oksuz et al. (2017) aimed to minimize line efficiency, Zhang et al. (2019) dealt with the type-2 problem. Zhang et al. (2020) conducted a multi-objective study to minimize ergonomic risk and cycle time. Another study that presented a multi-objective approach to the problem while addressing ergonomic risk belongs to Akyol and Baykasoğlu (2019b). Other studies that consider the problem as multi-objective were carried out by Ramezanian and Ezzatpanah (2015) and Zacharia and Nearchou (2016). Ramezanian and Ezzatpanah (2015) aimed at minimizing the cycle time and the total operating cost related to workers in mixed-model assembly lines. Zacharia and Nearchou (2016), on the other hand, optimized the cycle time and workload smoothness on serial lines. Araújo et al. (2012) and Araújo et al. (2015) were taken into account parallel stations in their studies. Costa and Miralles (2009) and Moreira and Costa (2013) included job rotation in the problem. Yilmaz (2021) considered the sequence-dependent setup times and Yang et al. (2021) took into consideration the positional constraints. In terms of the nature of processing times, stochastic (Ritt et al. 2016; Liu et al. 2019a; Liu et al. 2019b and Liu et al. 2021a), robust (Moreira et al. 2015; Pereira 2018; Yilmaz 2020) and fuzzy (Zacharia and Nearchou 2020) ALWAB problems were also studied. Efe et al. (2018) focused on the age and gender-based workload. Katiraee et al. (2021) considered the diversity of workers in terms of perceived physical effort in addition to experience and Liu et al. (2021b) took into consideration energy consumption within the scope of the ALWAB problem.

It is seen from the literature that the ALWAB problem has received a considerable amount of attention from researchers in recent years. This study aims to optimize cycle time, which has a significant impact on system performance, in parallel with many researchers.

3. Proposed Artificial Bee Colony Algorithm

Since the ALWAB problem is in the NP-hard problem class, as the problem size increases, it becomes difficult to obtain solutions with exact methods. Accordingly, many researchers have proposed various heuristic and meta-heuristic approaches to deal with the problem. In this study, an ABC algorithm is developed in order to tackle the problem of ALWAB type-2.

ABC algorithm is a population-based meta-heuristic approach proposed by Karaboga (2005). The algorithm imitates the food-seeking behavior of the bees in nature (Oksuz et al. 2017). The control parameters of the ABC are *the number of food sources, the limit*, and *the termination condition*. *The number of food sources* corresponds to the number of solutions in the population. *The limit* is the maximum number of consecutive trials allowed for each solution to improve. A solution is abandoned if it does not improve at the end of several consecutive attempts equal to the limit. As a *termination condition*, the maximum number of iterations or the maximum number of consecutive unimproved iterations can be used. The maximum number of iterations indicates the number of iterations the algorithm will do. While the most successful solution found in an iteration is the local solution, the most successful solution throughout all past iterations is the global solution. If the global solution does not improve with respect to successive iterations as much as the value of the maximum number of consecutive unimproved iterations parameter, the algorithm terminates (Karas and Ozcelik 2021). In this study, the maximum number of consecutive unimproved iterations parameter is used. To check whether the values of the limit and the maximum number of consecutive unimproved iterations are exceeded, the solution failure counter and the iteration failure counter parameters are employed, respectively. ABC algorithm consists of the initial solution derivation, the employed bee stage, the onlooker bee stage, and the scout bee stage. The flowchart of the algorithm is given in Figure 1.

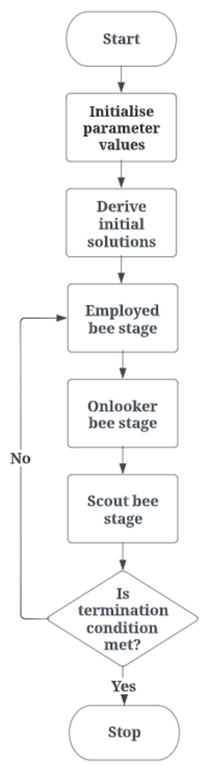

Figure 1. Flowchart of the ABC algorithm

3.1. Inputs of the problem

The inputs of the ALWAB problem are the number of tasks, the number of workstations, the number of workers, the precedence relations among the tasks, and the processing times of tasks for each worker.

3.2. Representation of solutions

In the developed algorithm, the solutions are represented by two vectors. The first vector shows the workstation number of each task. The length of this vector is equal to the number of tasks. The second vector represents the workstation of each worker allocated, and its length is as much as the number of workers. Task & workstation and worker & workstation vectors for an example with 10 tasks and 3 workers are given in Figure 2a and 2b, respectively. According to the task & workstation vector, tasks 1 and 2 are assigned to the 1st station. Tasks 4, 7, and 9 are allocated to the 2nd station, and tasks 3, 5, 6, 8, and 10 are in the 3rd workstation. The worker & workstation vector shows that worker 1 is working at the 2nd workstation. While worker 2 is assigned to workstation 3, worker 3 is to workstation 1.

1	1	3	2	3	3	2	3	2	3

(a)

2	3	1

(b)

Figure 2. Solution vectors

3.3. Deriving initial solutions

ABC algorithm starts with deriving random initial solutions as much as the number of food sources. To derive a feasible solution, a random assignment order is created by providing the precedence relations among the tasks. Then, the number of tasks that will be assigned to each station is randomly determined ensuring that there is at least one task at each station and that the sum of the number of tasks at all stations is equal to the total number of tasks. After a solution is derived, the same steps are applied to generate the next solution. When the number of solutions is equal to the number of food sources, the stage ends.

3.4. Employed bee stage

At the employed bee stage, a neighbor food source is found for each source in the population. The neighbor solution obtained replaces the current one if it is more successful. In the classical ABC algorithm, if the neighbor is not more successful, the current one is preserved. However, in the proposed ABC, a neighbor solution that is worse than the existing solution to a certain ratio is also accepted to avoid getting stuck in the local optimum. If the neighbor is accepted, the solution failure counter of this solution is reset. Otherwise, the value of this counter increases by one unit. Then, the same operations are performed individually for all solutions in the solution population. The neighborhood structures used in the algorithm are described below. All of them provide the constraints of the problem and generate feasible neighbor solutions.

Task Insert: Moving a random task to a different workstation that provides precedence relations.
Task Swap: Exchanging the workstations of two random tasks, ensuring precedence relations.
Worker Swap: Exchanging the workers of two random workstations, ensuring task-worker compatibilities.

3.5. Onlooker bee stage

At this stage, the steps of the previous stage are applied for randomly selected solutions with binary tournament selection. The difference between the onlooker bee stage from the employed bee stage is that a neighbor solution is found for each solution determined by the tournament selection, not for all solutions. At each step of this stage, two solutions are randomly selected from the solution population. Then, a neighbor solution is attained for the solution whose objective function value is more successful. The same neighborhood structures and neighbor solution acceptance criteria in the previous stage are applied in this stage. The same steps are repeated as many times as the number of solutions (number of food sources) in the population.

3.6. Scout bee stage

At the scout bee stage, the solution failure counter values of all solutions in the population are checked. The solutions, the solution failure counter of which exceeds the value of the limit parameter, are abandoned. In the classical ABC algorithm structure, a random feasible solution is derived instead of each abandoned solution. In the ABC algorithm developed within the scope of this study, to quickly reach better solutions, each abandoned solution is replaced by a neighbor of the most successful solution in the population. In addition, even if the solution failure counter value of the most successful solution in the population exceeds the limit value, it is ensured that this solution remains in the population in order not to lose this solution.

3.7. Evaluation of the most successful solution obtained in an iteration

An iteration ends with the completion of the scout bee stage. The most successful solution obtained in the iteration (local solution) is kept in memory. This solution is compared with the most successful solution found throughout all iterations (global solution). If the local solution is more successful than the global one, the iteration failure counter is reset, and this solution replaces the global solution. Otherwise, the counter is increased by one unit. If the counter value is not reached the maximum number of consecutive unimproved iterations, a new iteration starts from the employed bee stage. In the contrary case, the algorithm terminates.

4. Test Problems

A total of 320 ALWAB benchmark instances presented by Chaves et. al (2007) are used for the experimental evaluation of the proposed ABC algorithm. These instances are grouped into four families: Roszieg, Heskia, Tonge, and Wee-Mag. Each family has 8 groups, and each group contains 10 problems. The task and worker numbers of each instance are given in Table 2.

Table 2. Task and worker numbers of ALWAB benchmark instances

Family	The number of tasks	The number of workers
Roszieg	25	4 (Group 1-4) and 6 (Group 5-8)
Heskia	28	4 (Group 1-4) and 7 (Group 5-8)
Tonge	70	10 (Group 1-4) and 17 (Group 5-8)
Wee-Mag	75	11 (Group 1-4) and 19 (Group 5-8)

5. Results and Discussion

The proposed ABC algorithm is coded in the MATLAB 2021a programming language. In the employed bee and onlooker bee stages, neighbor solutions are better or a maximum of 5% worse than the current solution are accepted. The ABC is applied 20 times to each problem. The other parameter values used in the algorithm are given in Table 3.

Table 3. The parameter values used in the proposed ABC algorithm

Parameter	Value
The number of food sources	50
Limit	50
The maximum number of consecutive unimproved iterations	2000

The average of the objective function values found for 10 problems in each group of each family is shown in Table 4. For calculating the group average for each group, the most successful solution value obtained within 20 runs for each problem in that group is taken into account. For the same problems, the results obtained in the other ALWAB type-2 studies in the literature are also presented in Table 4. The most successful results in each group for the considered problem are highlighted in bold. The CPU times of the most successful solutions within 20 runs for the proposed algorithm for all problems are presented in Table 5 in seconds.

The presented ABC attains optimal solutions for all problems of the Roszieg and Heskia families. The most successful group average values in the accessible literature are obtained for the problems in the first four groups of the Tonge family, one of the large-sized problem families. The obtained results for the last four groups of the Tonge family and the problems belonging to the Wee-Mag family are close to the most successful solutions in the literature. The algorithm achieves all solutions in a reasonable time.

Table 4. Comparison of the results obtained with the proposed ABC algorithm and the other approaches

Problem family	Group	Blum and Miralles (2011)	Moreira et al. (2012)	Mutlu et al. (2013)	Borba and Ritt (2014)	Polat et al. (2016)	Proposed ABC
Roszieg	1	**20.1**	**20.1**	**20.1**	**20.1**	**20.1**	**20.1**
	2	**31.5**	**31.5**	**31.5**	**31.5**	**31.5**	**31.5**
	3	**28.1**	**28.1**	**28.1**	**28.1**	**28.1**	**28.1**
	4	**28.0**	**28.0**	**28.0**	**28.0**	**28.0**	**28.0**
	5	**9.7**	**9.7**	**9.7**	**9.7**	**9.7**	**9.7**
	6	**11.0**	11.1	**11.0**	**11.0**	**11.0**	**11.0**
	7	**16.0**	**16.0**	**16.0**	**16.0**	**16.0**	**16.0**
	8	**15.1**	**15.1**	**15.1**	**15.1**	**15.1**	**15.1**
Heskia	1	**102.3**	**102.3**	**102.3**	**102.3**	**102.3**	**102.3**
	2	**122.6**	122.7	**122.6**	**122.6**	**122.6**	**122.6**
	3	**172.5**	**172.5**	**172.5**	**172.5**	**172.5**	**172.5**
	4	**171.2**	**171.2**	**171.2**	**171.2**	**171.2**	**171.2**
	5	**34.9**	**34.9**	**34.9**	**34.9**	**34.9**	**34.9**
	6	**42.6**	**42.6**	**42.6**	42.7	**42.6**	**42.6**
	7	**75.2**	**75.2**	**75.2**	**75.2**	**75.2**	**75.2**
	8	**67.2**	**67.2**	**67.2**	**67.2**	**67.2**	**67.2**
Tonge	1	94.9	92.8	94.1	91.3	91.4	**90.2**
	2	110.2	109.3	110.2	107.8	107.4	**106.7**
	3	165.0	162.2	165.2	160.8	160.3	**159.0**
	4	170.0	168.4	170.1	165.9	166.0	**163.9**
	5	33.1	34.1	33.1	32.2	**31.9**	33.9
	6	40.0	40.2	40.4	38.9	**38.1**	39.5
	7	66.4	66.6	66.4	64.5	**64.2**	66.8
	8	64.7	65.8	64.8	63.1	**62.6**	64.7
Wee-Mag	1	28.7	**26.7**	**26.7**	27.1	**26.7**	28.1
	2	33.6	32.2	32.7	32.1	**31.1**	33.2
	3	50.1	47.6	48.2	47.5	**47.0**	48.7
	4	48.6	45.6	46.0	45.4	**45.0**	47.1
	5	10.3	10.5	10.4	**9.9**	10.1	11.7
	6	11.9	12.3	12.1	**11.4**	11.6	13.2
	7	18.2	18.6	18.5	**17.7**	17.8	20.3
	8	18.1	18.4	18.4	**17.7**	**17.7**	20.6

Table 5. Solution times (s) of the proposed ABC algorithm

Family	Group \ Instance	1	2	3	4	5	6	7	8	9	10
Roszieg	1	0.04	0.04	0.19	0.08	0.02	0.02	0.03	0.05	0.10	0.06
	2	0.12	0.06	0.84	0.07	0.05	0.04	0.12	0.03	0.06	0.07
	3	0.12	0.02	0.04	0.07	0.02	0.06	0.07	0.06	0.09	0.05
	4	0.03	0.03	0.09	0.07	0.05	0.05	0.05	0.05	0.03	0.06
	5	0.06	0.10	0.13	0.12	0.12	0.10	0.02	0.07	0.14	0.10
	6	0.03	0.08	0.09	0.08	0.16	0.20	0.17	0.10	0.51	0.21
	7	0.14	0.15	0.33	0.15	0.12	0.07	0.18	0.12	0.13	0.25
	8	0.31	0.09	0.24	0.18	0.14	0.09	0.76	0.19	0.07	0.15
Heskia	1	0.73	0.55	3.36	0.46	2.44	0.51	0.65	0.54	0.23	0.14
	2	0.70	0.31	0.26	0.69	0.22	0.45	0.17	0.19	1.02	0.28
	3	1.14	0.16	0.28	0.71	0.79	0.17	0.35	1.59	0.34	0.25
	4	1.21	0.34	0.29	0.64	0.42	0.40	0.79	0.38	0.41	0.86
	5	1.38	0.76	1.90	3.84	2.50	1.16	3.00	0.58	0.22	0.98
	6	0.56	0.46	0.15	1.57	3.53	0.46	9.40	1.07	0.35	0.46
	7	1.06	1.21	2.35	0.06	0.13	0.76	0.17	3.00	0.43	0.36
	8	0.10	2.30	1.43	0.34	1.58	0.75	0.53	1.13	0.56	0.68
Tonge	1	58.41	67.09	37.14	100.8	78.04	36.01	43.86	69.27	19.53	10.91
	2	28.40	47.36	26.16	25.19	58.82	23.94	21.25	22.08	70.69	21.57
	3	38.70	81.17	10.51	21.12	25.82	13.04	10.48	29.55	25.83	33.39
	4	56.19	8.62	96.95	22.67	44.64	43.40	14.87	27.68	31.72	9.29
	5	33.13	31.06	18.66	75.13	24.67	65.10	52.81	82.06	14.93	13.69
	6	55.10	41.37	13.26	35.89	18.45	48.02	42.69	50.39	20.90	20.54
	7	58.64	40.48	37.98	2.25	55.20	80.83	77.46	49.40	42.11	95.05
	8	109.60	72.66	15.63	35.46	30.47	33.66	45.84	24.52	40.43	77.09
Wee-Mag	1	90.74	41.56	47.4	34.23	92.64	17.73	23.02	77.98	26.44	61.89
	2	19.51	22.09	53.81	55.73	132.7	68.19	81.40	52.16	103.70	23.73
	3	78.11	29.88	39.88	46.92	200.50	79.00	118.5	74.12	107.60	48.61
	4	77.61	36.75	211.90	42.30	158.30	45.12	37.23	62.36	110.60	42.99
	5	53.55	17.43	53.19	26.20	37.51	44.65	61.84	20.91	69.63	62.12
	6	139.50	32.03	36.76	41.24	55.87	54.56	59.67	93.06	35.55	56.03
	7	65.84	51.09	40.43	24.83	184.70	73.05	89.49	47.80	80.90	49.74
	8	84.18	45.09	68.08	82.67	51.15	77.41	41.10	48.93	27.76	49.69

6. Conclusion

In industrial applications, due to the fact that workers in production environments have different levels of experience and abilities, processing times are not equal to each other. In the ALB literature, the type of problem that takes into account the differences in processing time between workers is known as ALWAB. It is costly to solve large-sized examples of this NP-hard problem with exact solution methods. In this study, an ABC algorithm is employed to tackle ALWAB type-2 problem. In order to increase the speed of reaching successful solutions, some features are added to the neighbor solution acceptance structure and the scout bee stage of the classical algorithm. The results show that the presented algorithm is able to achieve promising results. In future studies, it is planned to hybridize the ABC algorithm with different approaches so order to obtain more successful results in large-sized problems.

References

Akyol, S.D. and Baykasoglu, A., A multiple-rule based constructive randomized search algorithm, *Journal of Intelligent Manufacturing*, vol. 30, no. 2, pp. 557-573, 2019a.

Akyol, S.D. and Baykasoglu, A., ErgoALWABP: a multiple-rule based constructive randomized search algorithm for solving assembly line worker assignment and balancing problem under ergonomic risk factors, *Journal of Intelligent Manufacturing*, vol. 30, no. 1, pp. 291-302, 2019b.

Araújo, F.F.B., Costa, A.M. and Miralles, C., Two extensions for the ALWABP: Parallel stations and collaborative approach, *International Journal of Production Economics*, vol. 140, no. 1, pp. 483-495, 2012.

Araújo, F.F.B., Costa, A.M. and Miralles, C., Balancing parallel assembly lines with disabled workers, *European Journal of Industrial Engineering*, vol. 9, no. 3, pp. 344–365, 2015.

Blum, C. and Miralles, C., On solving the assembly line worker assignment and balancing problem via beam search, *Computers & Operations Research*, vol. 38, no. 1, pp. 328-339, 2011.

Borba, L. and Ritt, M., A heuristic and a branch-and-bound algorithm for the assembly line worker assignment and balancing problem, *Computers & Operations Research*, vol. 45, pp. 87-96, 2014.

Chaves, A.A., Lorena, L.A.N. and Miralles, C., Hybrid metaheuristic for the assembly line worker assignment and balancing problem, *Proceedings of the 6th International Workshop on Hybrid Metaheuristics*, pp. 1-14, Udine, Italy, October 16-17, 2009.

Chaves, A.A., Miralles, C. and Lorena, L.A.N., Clustering search approach for the assembly line worker assignment and balancing problem, *Proceedings of the 37th International Conference on Computers and Industrial Engineering*, pp. 1469–1478, Alexandria, Egypt, October 20-23, 2007.

Costa, A.M. and Miralles, C., Job rotation in assembly lines employing disabled workers, *International Journal of Production Economics*, vol. 120, no. 2, pp. 625–632, 2009.

Efe, B., Kremer G.E.O. and Kurt, M., Age and gender-based workload constraint for assembly line worker assignment and balancing problem in a textile firm, *International Journal of Industrial Engineering*, vol. 25, no. 1, pp. 1-17, 2018.

Janardhanan, M.N., Li Z., Nielsen P. and Tang Q., Artificial bee colony algorithms for two-sided assembly line worker assignment and balancing problem, *Proceedings of the 14th International Symposium on Distributed Computing and Artificial Intelligence (DCAI)*, pp. 11-18, Porto, Portugal, June 21-23, 2017.

Janardhanan, M.N., Li, Z. and Nielsen, P., Model and migrating birds optimization algorithm for two-sided assembly line worker assignment and balancing problem, *Soft Computing*, vol. 23, no. 21, pp. 11263-11276, 2019.

Karaboga, D., An idea based on honey bee swarm for numerical optimization, *Technical Report-TR06*, Erciyes University Engineering Faculty Computer Engineering Department, pp. 1-10, 2005.

Karas A. and Ozcelik F., Assembly line worker assignment and rebalancing problem: a mathematical model and an artificial bee colony algorithm, *Computers & Industrial Engineering*, vol. 56, no. 107195, pp. 1-16, 2021.

Katiraee N., Calzavara M., Finco S. and Battini D., Consideration of workforce differences in assembly line balancing and worker assignment problem. *IFAC-PapersOnLine*, vol. 54, pp. 13–18, 2021.

Liu M., Liang B. and Chu F., A risk-averse assembly line worker assignment and balancing problem with uncertain processing time, *Proceedings of the 8th International Conference on Industrial Engineering and Systems Management (IESM)*, pp. 266-271, Shanghai, China, September 25-27, 2019a.

Liu M., Liu R. and Chu F., An improved model for assembly line worker assignment and balancing problem considering stochastic worker availability, *Proceedings of the 8th International Conference on Industrial Engineering and Systems Management (IESM)*, pp. 731-736, Shanghai, China, September 25-27, 2019b.

Liu M., Liu Z., Chu F., Liu R., Zheng F. and Chu C., Risk-averse assembly line worker assignment and balancing problem with limited temporary workers and moving workers, *International Journal of Production Research*, vol. 60, no. 23, pp. 7074-7092, 2021a.

Liu R., Liu M., Chu F., Zheng F. and Chu C., Eco-friendly multi-skilled worker assignment and assembly line balancing problem, *Computers & Industrial Engineering*, vol. 151, no. 106944, 2021b.

Miralles, C., Garcia-Sabater, J.P., Andres, C. and Cardós, M., Advantages of assembly lines in sheltered work centres for disabled. A case study, *International Journal of Production Economics*, vol. 110, no. 1-2, pp. 187-197, 2007.

Miralles, C., García-Sabater, J.P., Andrés, C. and Cardós, M., Branch and bound procedures for solving the assembly line worker assignment and balancing problem: Application to sheltered work centres for disabled, *Discrete Applied Mathematics*, vol. 156, no. 3, pp. 352-367, 2008.

Moreira, M.C.O., Cordeau, J.F., Costa, A.M. and Laporte, G., Robust assembly line balancing with heterogeneous workers, *Computers & Industrial Engineering*, vol. 88, pp. 254–263, 2015.

Moreira, M.C.O. and Costa, A.M., A minimalist yet efficient tabu search algorithm for balancing assembly lines with disabled workers, 41 *Anais Do XLI Simpósio Brasileiro de Pesquisa Operacional*, pp. 660–671, Porto Seguro, Bahia, Brazil, September 1-4, 2009.

Moreira, M.C.O. and Costa, A.M., Hybrid heuristics for planning job rotation schedules in assembly lines with heterogeneous workers, *International Journal of Production Economics*, vol. 141, no. 2, pp. 552–560, 2013.

Moreira, M.C.O., Ritt, M., Costa, A.M. and Chaves, A.A., Simple heuristics for the assembly line worker assignment and balancing problem, *Journal of Heuristics*, vol. 18, no. 3, pp. 505-524, 2012.

Mutlu, O., Polat, O. and Supciller, A.A., An iterative genetic algorithm for the assembly line worker assignment and balancing problem of type-II, *Computers & Operations Research*, vol. 40, no. 1, pp. 418-426, 2013.

Oksuz, M.K., Buyukozkan, K. and Satoglu, S.I., U-shaped assembly line worker assignment and balancing problem: A mathematical model and two meta-heuristics, *Computers & Industrial Engineering*, vol. 112, pp. 246–263, 2017.

Pereira, J., The robust (minmax regret) assembly line worker assignment and balancing problem, *Computers & Operations Research*, vol. 93, pp 27-40, 2018.

Polat, O., Kalaycı, C.B., Mutlu, Ö. and Gupta, S.M., A two-phase variable neighbourhood search algorithm for assembly line worker assignment and balancing problem type-II: An industrial case study, *International Journal of Production Research*, vol. 54, no. 3, pp. 722-741, 2016.

Ramezanian, R. and Ezzatpanah, A., Modeling and solving multi-objective mixed-model assembly line balancing and worker assignment problem. *Computers & Industrial Engineering*, vol. 87, pp. 74-80, 2015.

Ritt, M., Costa, A.M. and Miralles, C., The assembly line worker assignment and balancing problem with stochastic worker availability, *International Journal of Production Research*, vol. 54, no. 3, pp. 907-922, 2016.

Vilà, M. and Pereira, J., A branch-and-bound algorithm for assembly line worker assignment and balancing problems, *Computers & Operations Research*, vol. 44, pp. 105-114, 2014.

Yilmaz, H., Modeling and solving assembly line worker assignment and balancing problem with sequence-dependent setup times, *Soft Computing*, vol. 25, pp. 12899–12914, 2021.

Yilmaz, O.F., Robust optimization for U–shaped assembly line worker assignment and balancing problem with uncertain task times, *Croatian Operational Research Review*, vol. 11, pp. 229–239, 2020.

Yang H., Lee J.H. and Kim H.J., Assembly line worker assignment and balancing problem with positional constraints, *International Conference on Advances in Production Management Systems (APMS)*, pp. 3-11, 2021.

Zacharia, P.T. and Nearchou, A.C., A population-based algorithm for the bi-objective assembly line worker assignment and balancing problem, *Engineering Applications of Artificial Intelligence*, vol. 49, pp. 1-9, 2016.

Zacharia, P.T. and Nearchou, A.C., The fuzzy assembly line worker assignment and balancing problem, *Cybernetics and Systems*, vol. 52, pp. 221–243, 2020.

Zhang, Z., Tang, Q., Han, D. and Li, Z., Enhanced migrating birds optimization algorithm for U-shaped assembly line balancing problems with workers assignment, *Neural Computing & Applications*, vol. 31, no. 11, pp. 7501-7515, 2019.

Zhang Z., Tang Q.H., Ruiz R. and Zhang L., Ergonomic risk and cycle time minimization for the U-shaped worker assignment assembly line balancing problem: a multi-objective approach. *Computers &Operations Research*, vol. 118, no. 104905, pp. 1-15, 2020.

Biographies

Aslihan Karas Celik is currently serving as a research assistant at Eskişehir Osmangazi University under the Department of Industrial Engineering. She obtained her bachelor's degree from Anadolu University and her M.Sc. from the department of Industrial Engineering at Eskişehir Osmangazi University in 2017 and 2020, respectively. She is a Ph.D. student at the same institution. She worked for Bursa Technical University for one year and has been working at Eskişehir Osmangazi University since 2020.

Dr. Feristah Ozcelik is an associate professor of Industrial Engineering at Eskisehir Osmangazi University. She received her B.Sc., M.Sc., and Ph.D. degrees from the Industrial Engineering Department of the Eskisehir Osmangazi University, in 1997, 2001, and 2007, respectively. Her main research interests are in the areas of facility layout design, cellular manufacturing systems, assembly line balancing, and metaheuristics. She is also an active member of the Turkish Chamber of Mechanical Engineers.