PORT SAID ENGINEERING RESEARCH JOURNAL
Faculty of Engineering - Port Said University
Volume 26 No. 1 pp: 121-137

Balancing Two-Sided Multi-Manned Assembly Lines Using Genetic Algorithm

Nessren Zamzam[1], Nahid Afia[2], Amin El-Kharbotly[3], Yomna Sadek[4]

[1]*Design and Production Engineering Department, Faculty of Engineering, Ain Shams University, Cairo, Egypt,* nessren.zamzam@eng.asu.edu.eg
[2] *Design and Production Engineering Department, Faculty of Engineering, Ain Shams University, Cairo, Egypt,* amin_elkharbotly@eng.asu.edu.eg
3Design and Production Engineering Department, Faculty of Engineering, Ain Shams University, Cairo, Egypt, nahid_abdelhalim@eng.asu.edu.eg
[4]*Design and Production Engineering Department, Faculty of Engineering, Ain Shams University, Cairo, Egypt,* yomna.sadek@eng.asu.edu.eg

* Yomna Sadek, yomna.sadek@eng.asu.edu.eg, *DOI: 10.21608/pserj.2021.52874.1074*

Received 8-12-2020
Revised 14-3-2021
Accepted 19-4-2021

ABSTRACT

Parametric model, Optimization Two sided assembly lines are used for large sized products where assembly tasks are performed by two workers on both sides. Multi-manned lines are mainly used in the assembly of relatively large products, with a number of workers moving around the product performing assembly tasks. This movement may lead to interference and congestion of man and material. In this paper a new line configuration is investigated by combining the two types of lines in what is designated as Two-Sided Multi-Manned (TSMM) assembly line. The proposed configuration benefits from the advantages of both lines by proposing four workers; two on each side, avoiding interference of the work and reducing the assembly stations. A model based on genetic algorithm was developed to balance the proposed TSMM line under the objectives of minimizing the number of workers and the number of mated-stations. A new method for generating the initial population is proposed leading to remarkably faster convergence of the solution. A controlling parameter is introduced to enable the tradeoff between the number of workers and the number of mated stations, adding flexibility to the line design. Results reveal that the proposed model gives competitive results to genetic algorithm and particle swarm optimization in the two-sided assembly line benchmark problems. It converges to a final solution in considerably less number of iterations. The application of the TSMM line concept results in considerable reduction in the number of mated stations with space saving up to 50% for the same number of workers.

Keywords: Assembly lines, two-sided multi-manned, multi-objective, genetic algorithm

1. INTRODUCTION

Assembly line balancing problems (ALBP) were first investigated in 1955. Since that time, the research in this subject is always opting to cope with the industry applications, and get closer to realistic assumptions. Assembly lines started as a series of stations, where a single worker is assigned for each station (Figure 1 a) performing assembly tasks on the product. Due to the limited computational capabilities then, only small-sized problems were solved. With technology advancements, products got more complicated. Also, computational technologies became capable of handling bigger, and more complex mathematical models. Therefore, **two sided assembly lines** (TSAL) appeared (Figure 1.b). In this type, two workers work at the same time at opposite sides of the product. This step allowed more products to be included in the research such as cars, small trucks, helicopters and buses. Özcan & Toklu [1] stated the advantages of the two sided assembly lines. They offer shorter line length, and reduce worker movement, setup time, cost of fixtures, and material handling. The literature survey detected many efforts in solving TSAL balancing problem. The **single objective models** were the most studied problems. The objective of

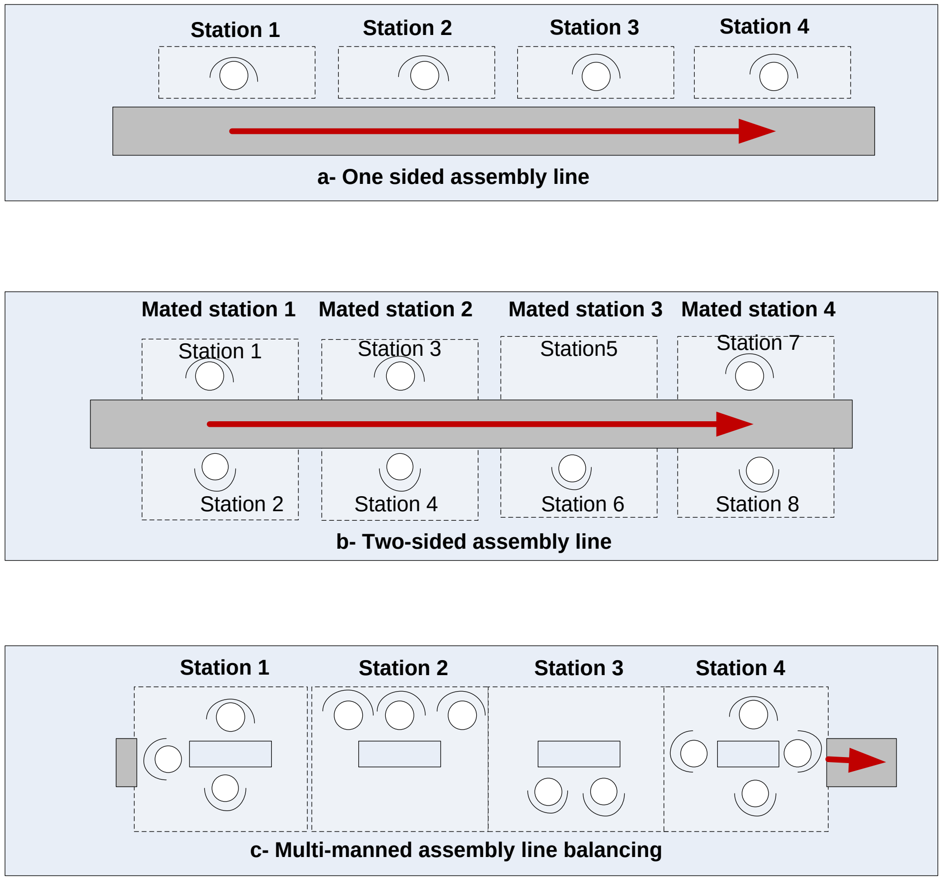

Figure 1 Assembly line configurations

minimizing the number of stations was addressed in the first two-sided assembly line model by Bartholdi [2]. Hu et al [3], Ozbakır & Tapkan [4], and Tapkan et al [5] also considered the objective of minimizing the number of stations. Tang et al [6] proposed a model with a primary objective of minimizing the number of mated stations and a secondary objective of minimizing the number of stations. Other researchers studied the multi-objective two-sided models seeking more realistic cases, and better results. Different objectives were under study. Taha et al [7] worked on minimizing the number of stations and the number of mated stations. Chutima & Chimklai [8] minimized the number of mated stations, number of stations (operators), workload smoothness, and maximized work-relatedness. Li et al [9] worked on maximizing the line efficiency, the smoothness index and the total relevant costs per product unit. Jawahar et al [10] considered minimizing number of workstations, and unbalanced time among workstations. Chutima & Olarnviwatchai [11] was interested in the car assembly industry. They considered the minimization of the number of paint color changes, the total number of ratio constraint violations and the utility work. Zhang et al [12] worked on minimizing cycle time and rebalancing cost. Gharoun et al [13] worked on minimizing cycle time and considered learning effect based on a predefined workstation and costs related to the assignment of skillful operators. **Multi-manned assembly lines** (MMAL) have more than one worker performing different tasks at the same time moving around the product, and possibly two or more workers work at the same side of the product at the same time (Figure 1.c). Multi-manned assembly lines can be applied to automotive industry. It can also be applied to smaller products like refrigerators, and washers. Assembly of such lines started at the industry by "trial and error" methods before researchers tackled it. Research efforts started in 2006 to cope with such real cases. Dimitriadis [14] stated the advantages of the multi-manned assembly lines; they achieve a better space utilization with the same number of workers, and the same total idle time of the line. Dimitriadis [14], Kellegoz & Toklu [15] , and Sepahi et al [16] used different heuristics to solve multi-manned single objective models to minimize the number of stations. They found that the multi-manned line outperforms the single line in the space utilization. Cevikcan [17] studied the same line minimizing the smoothness index. His results showed that the his model was more effective than that of Dimitriadis [14] by 20% shorter assembly line and about 20% reduction of average lead time. Kazemi & Sedighi [18] used a Genetic Algorithm (GA) to solve medium and large-sized problems under the objective of minimizing the total cost per production unit (stations and workers cost). Their initial population composed of seven priority rules beside the random generation. Their results showed that the GA performed better than other heuristics. They also showed that the addition of the priority rules in the initial population gave better results than the use of random initial population only. Fattahi et al [19] proposed two approaches to solve multi-manned problems; Mixed Integer Programming technique for small-sized problems, and Ant Colony Optimization (ACO) for medium and large-sized problems. They considered minimizing the number of workers as the first objective and minimizing number of stations as the second objective. The ACO could reach the optimal number of workers for all tested problems. Roshani et al. [20] were the first to attempt solving multi-objective multi-manned assembly lines. Objectives were to minimize smoothness index, line length and increase line efficiency. Their results were comparable to those of Fattahi et al [19] in shorter computational time. Giglio et al [21] proposed a mixed integer programming formulation for the problem of multi manned assembly line with assigning tasks to workers according to their qualification and skills. The objective was to minimize the total operating cost of the line. Their results showed that the operating cost of the system was reduced by reducing the number of stations and number of workers. However, their model could be applied to small-sized problems, which is not the normal application of the MMAL. Roshani & Giglio [22] formulated the MMAL balancing problem as a mixed-integer mathematical programming model. A primary objective minimized the cycle time for a given number of workstations and a secondary objective minimized the total number of workers. They proposed a two meta-heuristics approaches based on the simulated annealing algorithm. Results proved the reliability of the method. Chutima & Prasert [23] presented an adaptive extended coincident algorithm (AE-COIN). The multi-objective model minimized the number of workers, the number of stations, balanced the workloads between stations, and maximized work relatedness. The multiple objectives were optimized in a hierarchical manner, where the third hierarchy was optimized in a Pareto sense since they were conflicting in nature. Yilmaz &Yilmaz [24] proposed two methods to balance the MMAL with assignment restrictions. The first was a mathematical model to minimize the total number of workers for a given cycle time. The second was a Tabu search algorithm under the same assignment restrictions. Solving benchmark problems and comparing the results verified the effectiveness and efficiency of the proposed Tabu search algorithm.

Only three studies **fixed the positions** of the workers around the product in MMAL. They considered the position of the task (Right, Left or Either) instead of having the workers moving around the product. Zamzam et al [25] used a hybrid Genetic Algorithm (GA). They calculated a value for the maximum permissible number of workers in the station that prevents interference among workers, according to the size of the product, and the number of sides where assembly takes place. Their bi-objective model minimized the number of workers, and the number of stations. Ferrari et al [26] proposed a mixed integer programming model for balancing MMAL with fixed positions for car industry. To decrease the interference of workers, they divided the assembly tasks of the car into four levels of different heights. For each height there were 13 different assembly positions. The objectives were to optimize the line length, the line efficiency, and the workload smoothness. A simulated annealing algorithm with customized procedures was developed. A case study was solved to assess the efficiency of the proposed model. The results showed that the line efficiency was 89.85% with 20 workers, when the lower bound for the number of workers for the problem was 18. Yadav & Agrawal [27] developed a mathematical model to balance multi-manned parallel two-sided assembly line. The model gives a chance to assign one additional operator to each workstation per the product features. The objectives were to minimize the total idle time, and the cost related with tools. A Branch and bound algorithm was used to solve the problem. Results showed that less workstations were obtained as compared to the theoretical minimum number of workstations. That reduced the space as well as the cost of tools. Another exact solution approach was used for a small-sized case study with the option of tool sharing within the workstation or between different workstations.

Genetic algorithm. With the evolution in meta-heuristic techniques and powerful computers, it was possible to balance large and different types of assembly lines, with their NP-hard nature, while considering multi objective optimization. Many heuristics were used to solve the assembly line balancing problems. A genetic algorithm was proposed by Keun et al [28] , Ant Colony Optimization was proposed by Baykasoglu & Dereli [29] and Simaria & Vilarinho [30], enumerative algorithm was proposed by Hu et al [3] , Tabu search was proposed by Özcan & Toklu [1], and Bee colony was proposed by

Ozbakır & Tapkan [4]. Gen & Cheng [31] stated "During the last two decades, genetic algorithms have received considerable attention regarding their potential as a novel approach to multi-objective optimization problems".

Among all meta-heuristics used in balancing assembly lines, the **genetic algorithm** (GA) was the most used heuristics. Taha et al [7] generated initial population randomly and divided it into three portions. The first portion was generated in the forward direction, the second in the backward direction, while the third portion in forward and backward directions simultaneously. This enabled the generation of feasible solutions in different areas of the search space. Yang et al [32] generated initial populations randomly and the model was a multi-objective for seasonal demand products. Kazemi & Sedighi [18] added seven priority rules beside the random generation of the initial population. Akpınar & Bayhan [33] added three heuristics. All results showed that adding the heuristics to the initial population performed better than just using the random generation of population.

There are many **approaches for multi-objective models**. One approach is the minimum deviation method where all the objective functions are summed together. It is applicable when the analyst has partial information of the objectives, According to Özcan & Toklu [1]. This method gives a single optimum solution without considering any preferences among objectives. It does not allow any flexibility among objectives. Another method is the weighted sum approach where a weight is assigned to each objective then they are added in a single objective. It is very critical to estimate the values of the weights. Wrong estimations can result in undesirable solutions. Pareto approach is also another method for the multi-objective models. According to Gen & Cheng [31] "The Pareto approach assumes that no information on the preference among objectives is available and that all we know is that for each objective the greater value is preferred". It overcomes the weakness of the previous approaches. It gives all possible optimum solutions regarding either objective "non-dominated solutions". This gives flexibility to choose among solutions according to the strategy, and priorities of the decision maker. It can also be used when the objective functions are not conflicting.

From the previous literature, it can be concluded that researchers either studied two-sided assembly lines (TSAL) assuming two workers at maximum on each side of the station, or studied multi-manned assembly lines (MMAL) with more than two workers at each station moving around the product. Only one study considered the Two-sided Multi-manned (TSMM) line, but its approach was suitable for small-sized problems only (up to 12 tasks) which is not a real application for TSMM lines that are used in the assembly of large-sized products. It was also concluded from the literature that the genetic algorithm (GA) was extensively used to solve the assembly line balancing problems. An important conclusion is that generating the initial population by using priority rules (heuristics) beside the random generation yields better solutions than the random generation only**.** No research examined using priority rules only in the generation of the initial population.

Hence, the aim of this paper is to investigate the new line configuration; the two-sided multi-manned assembly line, and propose a model for the balancing of this type of line for large-sized problems. The proposed model uses GA to balance the line. The initial population is generated using few conventional assembly line priority rules only. The model is bi-objective; minimizing the number of workers and minimizing the number of mated stations.

The paper consists of five sections. The following section presents the problem definition and assumptions. Section 3 details the proposed heuristics. Section 4 presents the results and discussion, followed by the conclusion and future work in Section 5.

2. PROBLEM DEFINITION AND ASSUMPTIONS

This study investigates an assembly line configuration designated as two-sided multi-manned assembly line (TSMM) and tackles its balancing problem. The proposed two-sided multi-manned assembly line is represented in Figure 2. As a two-sided assembly line, a pair of two single stations can be installed on the opposite sides of the line / product. As a multi-manned assembly line, in each single station there can be multi-workers. These workers can work simultaneously on different tasks on the same product, at the same side. This is expected to offer shorter lines (even shorter than the two-sided lines), without excess travelling distances or workers interference (as in multi-manned lines).

2.1 Model assumptions

In addition to the common assumptions of assembly line problems which were stated by Roshani et al. [20], the present model has the following assumptions:

- Each position has one or two stations mated to each other.

- Each mated station has a maximum number of four workers; a maximum of two workers on each side.
- Each mated station has a minimum number of one worker.
- Tasks are constrained by the right side, or the left side,

NW	:	Total number of workers in the line
OSAL	:	One-sided assembly line
TSAL	:	Two-sided assembly line
TSMMAL	:	Two-sided multi-manned assembly line

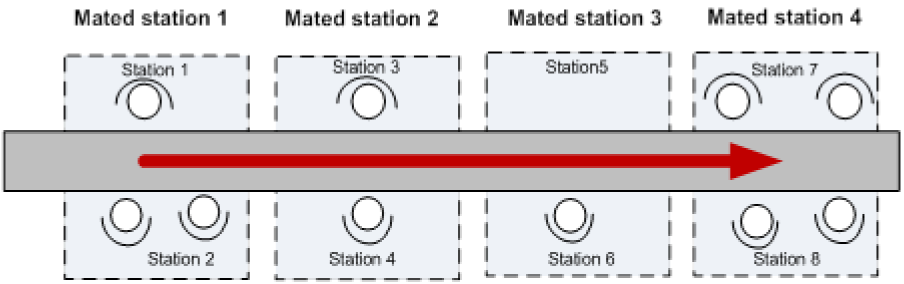

Figure 2 Two-sided multi-manned assembly line (TSMM)

or can be on either sides of the line.

- The task time is deterministic.
- The cycle time is predetermined.
- The sum of task times performed by any worker in any station should not exceed the cycle time.

More considerations

The model controls the upper bound of the permissible idle time per mated station. This is achieved by multiplying the value of the permissible idle time by a factor (β). According to the value of (β), the solution is either oriented towards save in labor (by decreasing the number of workers compromising the number of stations) or towards space saving (by decreasing the number of stations for more workers).

2.2 Nomenclature and abbreviations:

β	:	A parameter to control the upper bound of idle time (UB_{MS})
t	:	Task number , $t = \{1, \ldots T\}$
L	:	Left side of the station
MS	:	Mated station as shown in figure (2). MS = {1,…,NMS}
R	:	Right side of the station
S	:	Station as shown in figure (2). $S = \{1, \ldots, NS\}$
W	:	Worker, $W = \{1, \ldots NW\}$
AC	:	Set of available candidates
CT	:	Cycle time
LCR	:	Largest candidate rule
MMAL	:	Multi-manned assembly line
NMS	:	Total number of mated stations in the line
NS	:	Total number of stations in the line

Dt_W	:	Delay time of worker W
EF_t	:	Early finish of task t
ES_t	:	Early start of task t
F_t	:	Number of followers of task t
FTP_t	:	Finishing time of predecessors of task t
IDTw	:	Average idle time per worker
IT_{MS}	:	Idle time per mated station
LBT_t	:	Earliest possible station of task t
$NW_{MSR/L}$	:	Number of workers in mated station S on Right or Left side; $0 \le NW_{SR/L} \le 2$
NW_{MS}	:	Number of workers in mated station MS; $1 \le NW_{MS} \le 4$
P_{tj}	:	Precedence matrix , where P_{tj} = $\begin{cases} 1 & if\ task\ t\ preceeds\ task\ j \\ 0 & otherwise \end{cases}$
RPW_t	:	Ranked positional weight of task t
$RETUA_L$	:	Remaining time of unassigned tasks on left side
$RETUA_R$	:	Remaining time of unassigned tasks on right side
$RETUA_D$	:	Remaining time of unassigned tasks on either sides
Sc_R	:	Sequence of tasks according to a certain rule
Sc_{UA}	:	Set of candidate of unassigned tasks
SL_t	:	Slack of task t
Tm_t	:	Time of task t
TWL_W	:	Total workload of worker W
$TWL_{WMSR/L}$	:	Total work load of worker W in mated station MS in the right or left side
UB_{MS}	:	Upper bound of idle time per mated station
UBT_t	:	Latest possible station of task t
W_{LB}	:	Lower bound of the number of workers

W_{max}	:	Maximum number of workers per mated station
W_{min}	:	Minimum number of workers per mated station
$W_{WMSR/L}$	:	Worker number W in mated station MS in the right or left side

3. THE PROPOSED TWO-SIDED MULTI-MANNED (TSMM) HEURISTICS

The objective of the proposed heuristics is to minimize the number of workers and the number of mated stations. A bi-objective model using Genetic Algorithm (GA) is proposed for optimization. The Pareto approach is used to solve the bi-objective problem. As mentioned in the introduction, it gives flexibility in the choice among solutions.

The TSMM heuristics for assembly line balancing consists of two main stages; sequencing and assignment. They are sometimes designated in the literature as encoding and decoding. In the sequencing stage (encoding), the tasks are arranged in a feasible order after considering the precedence constraints. In the assignment stage (decoding), the tasks are assigned according to the previous sequence to one of the workers at appropriate side by applying certain assignment rules.

3.1 Sequencing (Encoding) stage and initial population

The proposed chromosome uses a task-based representation scheme. Each gene represents a task number. Hence, the chromosome resembles the sequence of the tasks.

The initial population consists of six chromosomes generated using six conventional assembly line balancing heuristics chosen from Baykasoglu [34]. Each follows a certain priority rule. The precedence relationships are preserved by the following steps:

- A set of Available Candidates {AC} is initially generated as the set of tasks with no predecessors.
- The task that satisfies the heuristic /rule is selected , removed from {AC} and added to the Sc_R. Sc_R is the sequence of tasks generated by a certain rule. It also represents the chromosome of the genetic algorithm. Ties are broken randomly.
- {AC} is updated such that the followers of the chosen task are added to {AC} without violating the precedence.
- Repeat until all tasks are added to Sc_R (the chromosome).

The six conventional heuristics as mentioned by Baykasoglu [34] are:

- Maximum **Ranked Positional Weight** (RPW); the tasks are arranged in descending order according to the value of the RPW.

$$RPW_t = Tm_t + \sum_{j=1}^{T} P_{tj} * Tm_j \qquad (1)$$

- Largest **Candidate Rule** (LCR); the tasks are arranged in descending order according to the value of the task time.

$$LCR_t = Tm_t \qquad (2)$$

- Maximum **Number of Followers** (F), the tasks are arranged in descending order according to the value of the number of followers

$$F_t = \sum_{j=1}^{T} P_{tj} \qquad (3)$$

- Maximum **Slack** (SL), the difference between the latest and earliest possible station of assignment. The tasks are arranged in descending order according to the value of the slack.

$$SL_t = UBT_t - LBT_t \qquad (4)$$

where

$$UBT_t = NS + 1 - \left[\frac{Tm_t + \sum_{j=1}^{T} P_{tj} * Tm_j}{CT}\right]^+ \quad \forall\, t, j = 1, \ldots..T \qquad (5)$$

$$LBT_t = \left[\frac{Tm_t + \sum_{j=1}^{T} P_{tj} * Tm_j}{CT}\right]^+ \quad \forall\ t = 1, \ldots..T \qquad (6)$$

- Maximum **Processing time divided by upper bound of task** t (PT_UB), the tasks are arranged in descending order according to the value of PT_UB.

$$PT\_UB_t = \frac{Tm_t}{UBT_t} \qquad (7)$$

- Minimum **Early start** (ES) of task. It is the maximum early finish (EF) of the predecessor tasks. The tasks are arranged in ascending order according to the value of ES.

$$ES_t = max\left(EF_j * P_{jt}\right) \qquad \forall j = 1, \ldots T \qquad (8)$$

3.2 Assignment (Decoding) stage

In this stage the sequence of the tasks obtained from the initial population are assigned to workers (decoded) according to the following steps:

STEP 1: Calculations and initial values

Calculate the lower bound of number of workers (W_{LB}), the average idle time per worker (IDTw), and the upper bound of the idle time per mated station (UB_{MS}). This bound varies from one mated station to another according to the number of its workers. Set the initial values for β, MS, W_{max}, W_{min}, $W_{WMSR/L}$, and $TWL_{WMSR/L}$.

$$W_{LB} = roundup\ (\frac{\sum_{t=1}^{T} Tm_t}{CT}) \quad (9)$$

$$IDTw = \frac{CT * W_{LB}\ - \sum_{t=1}^{T} Tm_t}{NW} \quad (10)$$

$$UB_{MS} = \beta * IDTw * NW_{MS} \quad \forall MS \quad (11)$$

From eq. 11, the parameter β controls the upper bound of the permissible idle time (UB) per mated station. It helps increase or decrease this bound.

<u>Initial values:</u>

- $\boldsymbol{\beta = 2,}$
- Mated station to be assigned tasks (MS) =1,
- Maximum number of workers per mated station (W_{max}) = 4,
- Minimum number of workers per mated station (W_{min}) = 1,
- Worker to be assigned tasks ($W_{WMSR/L}$) = 0,
- Total work load of worker W in mated station MS in the right or left station ($TWL_{WMSR/L}$) =0

STEP 2: Task selection

Select the first element of Sc_R. If Sc_R = {∅} go to step 4

STEP 3: Assigning task

3a- If task t has a specific side (Right or left):

- For the workers on this side, assign the task to the worker who satisfies the following relations

$$TWL_{WMSR/L} + Tm_t \leq CT \quad and \quad FTP_t + Tm_t \leq CT \quad (12)$$

- If both workers satisfy the previous relations, assign the task to the worker who has minimum delay time (Dt_W).

$$Dt_W = FTP_t - \sum Tm_j \ where\ j \in tasks\ assigned\ to\ worker\ W \quad (13)$$

- If both workers satisfy the previous condition, assign the task to the worker who has minimum TWL_W.
- If both workers satisfy the all the previous conditions, assign the task arbitrary.
- Update the attributes of the workers, the tasks, and the stations (FTP_t, TWL_W, $TWL_{WMSR/L}$,…etc.) according to the assigned tasks to workers.
- Remove t from Sc_R then go to step 2
- If task t can't be assigned to any worker move T_t from Sc_R to Sc_{UA}, then go to step 5

3b- If task t does not have a specific side (either):

▼ If all positions are empty calculate and compare for each side (Right and Left), assign the task to the side of minimum $RETUA_D$

$$RETUA_D = \sum T_t \quad where\ t\ \in Sc_{UA} \quad (14)$$

- If $RETUA_R = RETUA_L$, assign task t to the same position of its immediate follower
- If its immediate follower is of "either side" assign the task to any side arbitrary.
- Remove t from Sc_R then go to step 2

▼ If at least one position is occupied

- For all workers, assign the task to the worker who satisfies the relations in eq.(12)
- If more than one worker satisfy the previous relation, assign the task to the worker of minimum TWL_W.
- If more than one worker satisfy the previous condition assign the task to the worker who has minimum delay time (Dt_W)
- If more than one worker satisfy the previous condition, assign the task to the worker with minimum $RETUA_D + TWL_W$
- If more than one worker satisfy all the previous conditions, assign the task arbitrary.
- Update the attributes of the workers, the tasks, and the stations (FTP_t, TWL_W, $TWL_{WMSR/L}$,…etc.) as in eq 13.
- Remove T_t from Sc_R then go to step 2
- If task t can't be assigned to any worker move T_t from Sc_R to Sc_{UA}, then go to step 5

STEP 4: Evaluating the idle time of workers

- Calculate average idle time per mated station (IT_{MS})

$$IT_{MS} = \frac{CT * NW_{MS-} \sum Tm_i}{NW_{MS}} \quad (15)$$

- If IT_{MS} > UB and $NW_{MS} > W_{min}$ cancel one worker (according to rule 4a) at a time and restart assigning the tasks as in STEP 2, until IT_{MS} < UB or $NW_{MS} = W_{min}$

According to step 4, if the average idle time per worker is longer than the permissible, a worker is unassigned from the station, and the assignment process restarts. Note that the maximum permissible number of workers per mated station is controlled by the value of UB. The value of UB is in turn controlled by β (as mentioned at the end of STEP1). Hence, the permissible number of workers per mated station is highly affected by the value of β.

Rule:

- Evaluate NW_{MSR} and NW_{MSL} and compare both values
  - If $NW_{MSL} \neq NW_{MSR}$, remove a worker from the side of maximum value
  - If $NW_{MSL} = NW_{MSR}$ remove a worker from the side of minimum $RETUA_D$
    - If $RETUA_R$ = $RETUA_L$ remove the worker of maximum IDT_W

$$IDT_W = CT - TWL_{WMSR/L} \quad (16)$$

  - If more than one worker satisfies the condition, remove worker/s arbitrary so that one worker is left.

STEP 5:
Select the first element of Sc_{UA}. If $Sc_{UA} = \{\emptyset\}$ go to step 7

STEP 6:
Open a new mated station (MS= MS +1), then go to step 3 and repeat until $Sc_{UA} = \{\emptyset\}$

STEP 7: END

The six generated chromosomes of the encoding stage are the initial population. According to the fitness function of each, selection, mutation, and crossover are applied.

3.3 The fitness function

The fitness function evaluates the performance of each chromosome in the GA. According to the fitness value, the chromosome is selected or replaced in the next generation. In the proposed model two objectives are considered: minimizing the number of workers (*NW*), and the number of mated stations (*NMS*). A bi-objective model using the Pareto approach is used to minimize both objectives. The Pareto Approach gives equally good solutions. This is the essence of Multi-objective optimization.

3.4 The Genetic algorithm parameters

The different parameters of the genetic algorithm proposed are listed in Table 1. Parameters include the selection, crossover, mutation, and other parameters. Preliminary examination has been held to find the best values for such parameters. Finally, Table 1 shows the values selected. The stopping criteria is either reaching the lower bound of the number of workers, or after 20 generations; the earliest of both.

Table 1. GA parameters

Parameter	Value / Type
Population size	6
Crossover rate	0.8
Mutation rate	0.2
Elite	2
Number of generation	20
Selection technique	Stochastic remainder
Crossover technique	Two point crossover technique
Mutation technique	Scramble mutation

3.5 The general form of the proposed model

The proposed model is a general model that can be used to balance different assembly line configurations. It can used for the one-sided (OSAL), two-sided (TSAL), multi-manned (MMAL), and the two-sided multi-manned (TSMMAL) assembly lines. Table 2 shows the parameters used or neglected for each line configuration. The minimum number of workers in all cases is one.

Table 2 Selecting parameters to decide types of assembly line

Line type	Side of tasks	Max. number of workers on left side	Max. number of workers on right side
OSAL	Left only	one	zero
TSAL	L, R, Either	one	one
MMAL	Either	Four	
TSMMAL	L, R, Either	Two	Two

4. RESULTS AND DISCUSSION

The aim of these experiments is to study the following:

- Evaluate the performance of the proposed TSMM heuristics in solving the assembly line balancing problems. This is achieved by solving benchmark problems and comparing the results to other algorithms in the literature.
- Investigate the advantages of the two-sided multi-manned line configuration regarding the space saving, and the ability to reach the lower bound of the number of workers. This is achieved by solving the benchmark problems and comparing the results of the two-sided multi-manned line to the results of the two-sided line.
- Prove the capability of the proposed TSMM heuristics in being a general one that can deal with different line configurations. This is achieved by using the model to solve benchmark problems for different line configurations.
- Assess the performance of the initial population proposed. This is achieved by comparing the convergence and consistency of the TSMM heuristics to other available data from the literature with different initial population.

4.1 Evaluating The Performance Of The Proposed Tsmm Heuristics

The evaluation of the performance of the proposed model is done by comparing its results to the available results of Yadav & Agrawal [27] who considered the two-sided multi-manned line. Their two small-sized problems are not enough for comparison. Therefore, more results will be obtained by solving two-sided assembly line balancing benchmark problems by the proposed TSMM heuristics as a two-sided line. Then, comparing the results to other two-sided assembly line balancing models in the literature. This evaluates the proposed encoding and decoding stages, as well as the proposed initial population.

Table 3 presents the comparison with Yadav & Agrawal [27]. They solved two small sized problems; 12 tasks and 13 tasks. It is worth mentioning that their model can have up to three workers on one side of the mated station, and it has tool sharing constraints. These main two differences in the model explain the difference in the results.

Table 3 Number of workers (NW) and Number of mated stations (NMS) for TSMMAL problems

	CT	Yadav [27]		TSMM heuristics	
		NW	NMS	NW	NMS
P12 Yadav [27]	6	10	3	8	3
P13 Yadav [27]	6	9	3	9	4

For the P12 problem, the proposed TSMM heuristics could reach a lower number of workers (NW), and the same number of mated stations (NMS). Perhaps it is because of the tool sharing constraints that their model could not reach the 8-workers solution. For P13 we could reach the same number of workers, and one more mated station. Their model resulted in three workers on one side of the third mated station. Our model has up to two workers on one side of the station. This is the reason of having one more mated station than Yadav & Agrawal [27].

Table 4 presents the seven benchmark problems. Four small-sized problems namely; P9, P12, P16, and P24 were presented by Kim et al [35]. Three large-sized problems namely; P65, P148, and P205 were presented by Lee et al [36]. Each of the previous problems has several instances for different cycle times. As all GA models, the problem has to be solved more than once as results may not coincide. In our case, each instance is solved five times. Although it seems to be too little for judgment, however, all five results always coincide. This has to do with the consistency of the solution discussed in section 4.3.

Table 4 compares the number of stations (NS) resulting from the proposed TSMM Genetic Algorithm and from four other heuristics in the literature; Ant colony-based algorithm by Kellegoz & Toklu [15], Tabu Search algorithm by Özcan & Toklu [1], Enumerative algorithm by Jawahar [10], Bee colony intelligence by Yuan et al [37], and Genetic algorithm by Taha [7]. In the table, highlighted values indicate the highest NS value among other heuristics. From figure (1.b) each station has one worker. Hence, the number of workers is the same as the number of stations. So, the lower bound for the number of stations is the same as the lower bound for the number of workers (W_{LB}). From Table 4, it can be seen that no model is superior to the others in all the instances. There is always one model that performs better in some instances and worse in other instances. It can also be seen that the proposed TSMM heuristics proved to be competitive, as it could reach the lower bound for the number of stations in most cases. This set of experiments proves that the proposed TSMM heuristics is competitive and reliable to be used in solving assembly line balancing problems.

Table 4 Number of stations (NS) of two-sided benchmark problems resulting from different heuristics

	CT	Lower Bound	Kellegoz [15]	Özcan [1]	Jawahar [10]	Yuan [37]	Taha [7]	TSMM
P9	4	5	5	5	5	5	5	5
	5	4	4	4	4	4	4	4
	6	3	3	3	-	3	3	3
P12	5	5	6	6		6	6	6
	6	5	5	5	5	5	5	5
	7	4	4	4	4	4	4	4
	8	4	-	4	4	4	4	4
P16	15	6	-	-	-	6	6	6
	16	6	-	6	6	6	6	6
	18	5	-	-	-	6	6	6
	19	5	-	5	6	5	5	5
	20	5	-	-	-	5	5	5
P24	18	8	-	8	8	8	8	8
	20	7	8	8	8	8	8	8
	24	6	-	6	7	6	6	6
	25	6	6	6	6	6	6	6
	30	5	5	5	-	5	5	5
	35	4	4	4	-	4	4	4
	40	4	4	4	-	4	4	4
P65	326	16	17	17	-	17	17	17
	381	14	15	15	-	14	14	14
	435	12	13	13	-	13	13	13
	490	11	12	11	-	11	11	11
	544	10	10	10	-	10	10	10
P148	255	21	21	21	-	21	21	21
	306	17	18	18	-	18	18	18
	357	15	15	15	-	15	15	15
	408	13	14	13	-	13	13	13
	459	12	12	12	-	12	12	12
	510	11	11	11	-	11	11	11
P205	1133	21	24	24	-	22	22	22
	1322	18	22	21	-	20	20	**19**
	1510	16	18	18	-	17	17	17
	1699	14	18	17	-	16	15	15
	1888	13	15	16	-	14	14	14
	2077	12	14	14	-	12	12	12
	2266	11	12	13	-	12	11	12
	2454	10	11	12	-	10	10	11
	2643	9	11	11	-	10	10	10
	2832	9	10	10	-	10	9	9

4.2 Advantages of the Two-sided multi-manned assembly line.

It is expected that the two-sided multi-manned assembly line requires smaller space compared to that of the two-sided line. This can be tested by comparing the results obtained from using the proposed model as two-sided multi-manned to those of the two-sided line. Notice that the results of the two-sided line are given in Table 4, section 4.1. comparison will be carried twice; based on the number of mated stations (NMS), and based on the number of workers (NW).

4.2.1 Comparison Based On The Number Of Mated Stations (Nms)

The model is tested for a set of TSAL benchmark problems. Each instance was solved five times. The results of the proposed model were compared to the results of Taha et al. [7] using Genetic algorithm (GA), and Tang et al [6] using hybrid Particle swarm algorithm (PSO). They were able to find the best results when compared to others. However, none of them was able to outperform the other in all the instances.

Table 5 presents the results of the three models regarding the number of workers and the number of mated stations. The first three columns list the benchmark problem size, the instance cycle time, and the lower bound for the number of workers(W_{LB}). The following columns give the results of the number of workers (NW) and the number of mated stations (NMS) of the three heuristics. Finally, the column of "SAVINGS" shows the saving in the number of workers, and the percentage of saving in the space by applying the TSMM heuristics. Saving in the space is represented by the saving in the number of mated stations. Although the saving in the number of workers is detailed in the next section, however, it is mentioned here in order to show that the space saving was not compromised by the number of workers.

It is clear that the space savings in the small-sized problems are relatively small. These problems have few numbers of stations that cannot show much improvement. On the other hand, for large-sized problems, we could reach space saving up to 50% than Tang et al [6] and Taha et al. [7] for the same number of workers (or less). More space saving can be achieved for larger number of workers. This proves the advantages of multi-manned two-sided assembly lines that can yield space saving greater than the regular two-sided lines.

Due to the pareto approach used in the proposed bi-objective model, each of P16CT18 and P16CT21 resulted in two alternative solutions. One of them matches the results of Taha [7] and Tang [6]. The other solution saves one worker on the account of increasing one mated stations. This offers more flexibility to the decision maker either to save in the space or to save in the labor.

4.2.2 Comparison based on the number of workers (NW)

The results of the 44 instances of Table 5 were compared again to both Taha et al. [7] and Tang et al [6] regarding the number of workers. Table 6 summarizes the results. Eight instances are not listed in the table because the three models could not reach the lower bound of number of workers. These instances are P12 CT5, P24 CT 20, P65 CT326, P205 CT1133, P205 CT1322, P205 CT1510, P205 CT1699, and P205 CT2643. Table 6 shows only the instances where a heuristic outperformed another. Comparison aspect is the number of workers (NW) reached by the three heuristics. The proposed model outperformed both Taha et al. [7] and Tang et al [6] in three instances; (P16 CT18, P16 CT21, and P65 CT 435) where it reached the lower bound for the number of workers. In four instances (P148 CT204, P148 CT306, P205 CT1322, and P205 CT1888) the proposed TSMM model outperformed Taha et al. [7]. In one instance (P205 CT2266) the TSMM model outperformed Tang et al [6] and reached the lower bound as Taha et al. [7]. In one instance only (P205 CT2454) Tang et al [6] and Taha et al. [7].

Table 5 Benchmark problems NMS comparison results

Prob.	CT	W_{LB}	Taha [7]		Tang [6]		TSMM GA		SAVINGS			
									compared to Taha [7]		compared to Tang [6]	
			NW	NMS	NW	NMS	NW	NMS	NW	Space %	NW	Space %
P9	4	5	5	3	5	3	5	3	0	0	0	0
	5	4	4	3	4	2	4	2	0	33	0	0
	6	3	3	2	3	2	3	2	0	0	0	0
p12	4	7	7	4	7	4	7	4	0	0	0	0
	5	5	6	3	6	3	6	2	0	33	0	33
	6	5	5	3	5	3	5	2	0	33	0	33
	7	4	4	2	4	2	4	2	0	0	0	0
	8	4	4	2	4	2	4	2	0	0	0	0
P16	15	6	6	4	6	4	6	3	0	25	0	25
	16	6	6	4	6	3	6	3	0	25	0	0
	18	5	6	3	6	3	6	3	0	0	0	0
							5	4	1	-33	1	-33
	19	5	5	3	5	3	5	3	0	0	0	0
	20	5	5	3	5	3	5	3	0	0	0	0
	21	4	5	3	5	3	5	3	0	0	0	0
						3	4	4	1	-33	1	-33
	22	4	4	2	4	2	4	2	0	0	0	0
P24	18	8	8	4	8	8	8	3	0	25	0	-
	20	7	8	4	8	8	8	3	0	25	0	-
	24	6	6	3	6	6	6	3	0	0	0	-
	25	6	6	3	6	6	6	2	0	33	0	-
	30	5	5	3	5	5	5	2	0	33	0	-
	35	4	4	3	4	4	4	2	0	33	0	-
	40	4	4	2	4	4	4	2	0	0	0	-
P65	326	16	17	9	17	9	17	7	0	22	0	22
	381	14	14	8	14	7	14	7	0	13	0	0
	435	12	13	7	13	7	12	5	1	29	1	29
	490	11	11	6	11	6	11	5	0	17	0	17
	544	10	10	5	10	5	10	4	0	20	0	20
P148	204	26	27	14	26	13	26	9	1	36	0	31
	255	21	21	11	21	11	21	6	0	45	0	45
	306	17	18	9	17	9	17	7	1	22	0	22
	357	15	15	8	15	8	15	4	0	50	0	50
	408	13	13	7	13	7	13	4	0	43	0	43
	459	12	12	6	12	6	12	3	0	50	0	50
	510	11	11	6	11	6	11	3	0	50	0	50
P205	1133	21	22	13	22	11	22	10	0	23	0	9
	1322	18	20	10	19	10	19	8	1	20	0	20
	1510	16	17	9	17	9	17	7	0	22	0	22
	1699	14	15	8	15	8	15	7	0	13	0	13
	1888	13	14	7	13	7	13	6	1	14	0	14
	2077	12	12	6	12	6	12	4	0	33	0	33
	2266	11	11	6	12	6	11	4	0	33	1	33
	2454	10	10	5	10	5	11	4	-1	20	-1	20
	2643	9	10	5	10	5	10	4	0	20	0	20
	2832	9	9	5	9	5	9	3	0	40	0	40

Table 6 Number of workers (NW) for benchmark problems resulting from three heuristics.

Prob.	CT	W_{LB}	TSMM		Taha [7]		Tang [6]		Comparison		
			NW	NMS	NW	NMS	NW	NMS	TSMM	Taha [7]	Tang [6]
P16	**18**	**5**	**5**	**4**	**6**	**3**	**6**	**3**	√ *		
	21	**4**	**4**	**4**	**5**	**3**	**5**	**3**	√ *		
P65	435	12	**12**	**5**	13	7	13	7	√ *		
P148	204	26	**26**	**9**	27	14	26	13	√*		√*
	306	17	**17**	**7**	18	9	17	9	√*		√*
P205	1322	18	**19**	**8**	20	10	19	10	√		√
	1888	13	13	6	14	7	13	7	√*		√*
	2266	11	**11**	**4**	11	6	12	6	√*	√*	
	2454	10	11	4	10	5	10	5		√*	√*

√ Reached the least number of workers (NW)
***** Reached the lower bound for the number of workers **(**W_{LB})

outperformed the proposed TSMM heuristics. It is clear that the results of TSMM heuristics are closer to the lower bound of the number of workers.

From Table 5 and Table 6, it is clear that solving the two-sided benchmark problems as a multi-manned assembly line has shown considerable improvement in the number of mated stations. This is because it gave shorter assembly line with space saving up to 50% and the results of the present model are close to the lower bound of the number of workers. This proves the advantages of the two-sided multi-manned line over the two-sided line.

4.3 Effect of the proposed initial population on the convergence and consistency

Convergence of the solution in the genetic algorithm occurs when all individuals in the generation are identical. Consistency occurs wen all runs of a problem result in the same solution. The convergence and consistency of a Genetic Algorithm are good indicators of the performance of the algorithm. Two large-sized benchmark instances are compared to Taha et al. [7]. All results were obtained by solving each instance five times.

Table 7 summarizes the results. Table 7 compares the number of workers (NW) obtained by each algorithm. It also compares the number of generations that were run to reach NW. From the table, for both benchmark problems, the proposed TSMM algorithm reached a better solution faster than Taha et al. [7]. The smaller NW indicates a better solution. The fewer number of generations indicates the faster solution. TSMM converged in the third generation to 17 workers, while the random initial population of GA could not reach the same value till the 100th generation, which is one of the stopping criteria for Taha et al. [7]. Although data is not available to compare the computational time; yet, the number of generations for convergence can be a good indicator that the proposed model consumes less time.

Table 7 Convergence comparison for P205 CT1510, and CT1312

		Taha [7]	TSMM
P205 CT1510	NW	19	17
	Generation that reached min. NW	35	3
	No. of generations till stopping	100	20
P205 CT1312	NW	21	19
	Generation that reached min. NW	15	4
	No. of generations till stopping	100	20

To test consistency, the proposed TSMM heuristics was used to solve each instance five times. It resulted in the same solution for each of the five runs. Moreover, the mean fitness value and the best fitness value coincided after few generations. This proves the high consistency. Figure 3 shows a comparison between Taha et al [7] and the TSMM heuristics. In the figure, the black dots (dark dots) indicate the best fitness value, and the blue dots (light dots) indicate the mean fitness value. From Figure 3(a), the mean fitness value and the best fitness value do not coincide. Besides, convergence of the mean fitness value started in the 50th generation at a value of 19 (NW = 19). It is worth mentioning that their best solution was (NW=19) till the 100th generation. From Table 5, we know that Taha et al could reach (NW=17). It was not reached till the 100th generation. In Figure 3.b. of TSMM, both types of dots coincide at fitness value of 17 starting the third generation. This means that the GA reached (NW = 17) at the 3rd generation. Convergence of the results also started in the 3rd generation.

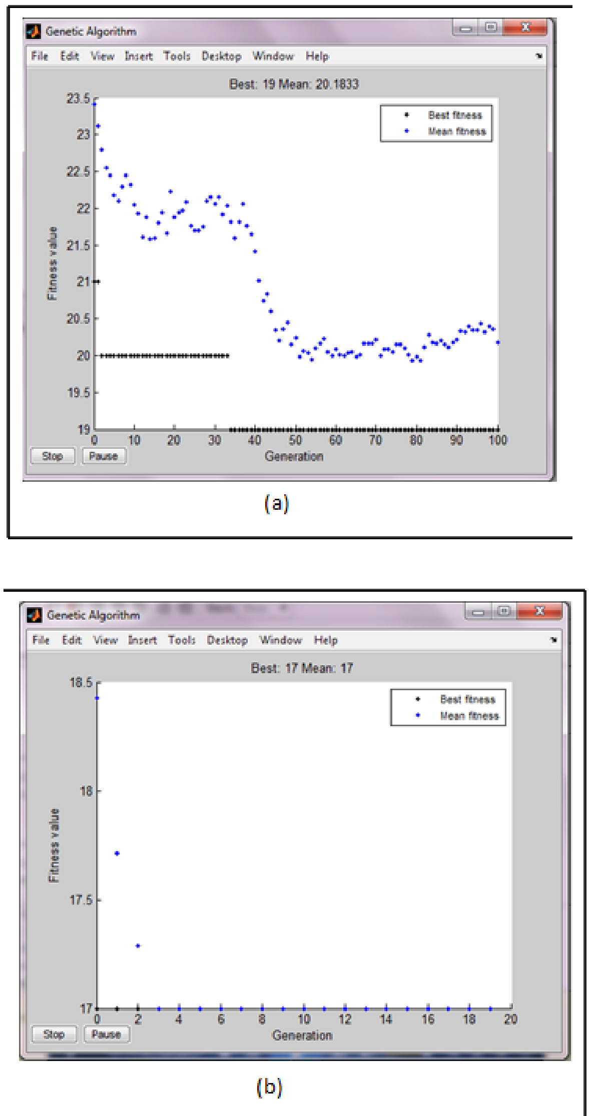

Figure 3 P205 CT1510 Convergence and consistency of

(a) Taha (b) TSMM

4.4 Using TSMM heuristics for balancing different assembly line configurations

The proposed TSMM heuristics exist in a general-form used to solve different types of assembly lines by adjusting the number of workers, the location of the worker in the station and the positioning of the tasks. A medium-sized benchmark problem (P65 of five instances) is solved for four different line types; one sided (OSAL), two sided (TSAL), multi-manned (MMAL), and two-sided multi-manned line (TSMM). The number of workers is compared for the four lines. The results are shown in Table 8.

P65		**CT**	**326**	**381**	**435**	**490**	**544**
		W_{LB}	16	14	12	11	10
Assembly line configuration	**OSAL**	**NW**	17	14	12	11	10
		NS	17	14	12	11	10
	TSAL	**NW**	17	14	13	11	10
		NMS	9	7	7	6	5
	MMAL	**NW**	17	14	12	11	10
		NS	6	5	5	4	4
	TSMMAL	**NW**	17	14	12	11	10
		NMS	7	7	5	5	4

From Table 8, it is found that the proposed model could reach the lower bound of the number of workers in most of the instances for the four line configurations. It can be seen that the one-sided assembly line always yields the largest number of stations (occupying the largest space), while the multi-manned line yields the smallest number of stations (occupying the smallest space). The MMAL sometimes yields a smaller NMS than the TSMMAL as MMAL is not constrained with the side of tasks. Hence, it has more flexibility with the assignment of the tasks among workers. It is worth mentioning that the MMAL has the disadvantage of interference of man and material around the product.

5. CONCLUSION AND FUTURE WORK

This work proposes a genetic algorithm to balance the two-sided multi-manned assembly line (TSMMAL). The TSMMAL is a two-sided assembly line that uses multi-manned on both sides. This line configuration benefits from the two-sided line fixing the workers in their sides which reduces the man and material handling, and reduces the congestion during work. It benefits from the multi-manned lines allowing more than multiple workers in each station which reduces the number of stations and saves space.

The research has two main directions. The first direction proves the competitiveness of the proposed TSMM heuristics in balancing assembly lines. The second direction is to study the applicability and benefits of the two-sided multi-manned assembly line (TSMMAL) over the two-sided line.

For the first direction of research, the TSMM heuristics is proven to be competitive with the best results of the heuristics in the literature of balancing benchmark two-sided assembly lines. It can reach the lower bound of the number of workers in most of the assembly line problems. Moreover, the usage of limited priority rules for the initial population is efficient in solving the assembly line balancing problem. Besides, the solution converges to the optimal value in much fewer generations than the random initial population. It reduces considerably the time for searching for the elites. This results in considerably shorter computational time.

For the second direction of research, the results emphasize the advantage of the two-sided multi-manned assembly line (TSMMAL) over the two-sided line. It can save up to 50 % of the line length without increasing the number of workers. This leads to better area utilization, lower cost of fixtures and less waste in workers' movement.

Additionally, the proposed TSMM heuristics can be adopted easily to solve the assembly line balancing problem of different line configurations; one-sided, two-sided, multi-manned, and two-sided multi-manned.

For future work; the TSMM heuristics can be extended to include features such as synchronous and non-synchronous tasks, positional constraint, and zoning constraint. Further study should be made to determine the effectiveness of using limited number of chromosomes (elite chromosomes) in other problems as scheduling. It is also recommended to study extensively the effect of the parameter β on the results of the problem, and its recommended values.

Credit Authorship Contribution Statement:

Nessren Zamzam: conceptualization, formal analysis, investigation, resources, original draft.

Amin El-Kharbotly: methodology, validation, supervision, formal analysis, review

Nahid Afia: validation, supervision

Yomna Sadek: methodology, validation, supervision, formal analysis, writing, review and editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

6. REFERENCES

[1] Özcan U. & Toklu B. (2009). A tabu search algorithm for two-sided assembly line balancing, International Journal of Advanced Manufacturing Technology, 43; 822-829

[2] Bartholdi, J.J. (1993) Balancing two-sided assembly lines – a case study. International Journal of Production Research, 31; 2447–2461.

[3] Hu, X.-F., Wu, E. -F., & Ye, J. (2008). A station-oriented enumerative algorithm for two-sided assembly line balancing, European Journal of Operational Research, 186

[4] Ozbakır L. & Tapkan P. (2011). Bee colony intelligence in zone constrained two-sided assembly line balancing problem. Expert Systems with Applications, 38; 11947–11957

[5] Tapkan P, Ozbakir L. Baykasoglu A. (2012) Modeling and solving constrained two-sided assembly line balancing problem via bee algorithms. Applied Soft Computing, 12; 3343–3355

[6] Tang, Q., Li, Z., Zhang, L., Cao X., Zhang S. (2014) Balancing Two-sided Assembly Line Using Particle Swarm Optimization algorithm. International Journal of Advancements in Computing Technology (IJACT), 6; 136-144

[7] Taha B., El-Kharbotly A., Sadek Y., Afia N.A. (2011). Genetic Algorithm for solving two-sided assembly line balancing problems. Ain Shams Engineering Journal, 2; 227–240

[8] Chutima P., & Chimklai P. (2012). Multi-objective two-sided mixed-model assembly line balancing using particle swarm optimization with negative knowledge. Computers & Industrial Engineering, 62; 39–55

[9] Li, D., Zhang, C., Shao, X.(2014) A multi-objective TLBO algorithm for balancing two-sided assembly line with multiple constraints Journal of Intelligent Manufacturing ;doi:10.1007/s10845-014-0919-2.

[10] Jawahar N., Ponnambalam S., Sivakumar K., Thangadurai V. (2014). Heuristics for Multi-objective Optimization of Two-Sided Assembly Line Systems, the Scientific World Journal, Article ID458959, 16 pages

[11] Chutima, P., & Olarnviwatchai, S. (2016). A multi-objective car sequencing problem on two-sided assembly lines. Journal of Intelligent Manufacturing, 29(7), 1617-1636

[12] Zhang, Y., Hu, X., Wu, C. (2018). Heuristic Algorithm for Type II Two-sided Assembly Line

Rebalancing Problem with Multi-objective. In MATEC Web of Conferences (Vol. 175, p. 03063). EDP Sciences
[13] Gharoun, H., Hamid, M., Iranmanesh, S. H., Yazdanparast, R. (2020). Using an intelligent algorithm for performance improvement of two-sided assembly line balancing problem considering learning effect and allocation of multi-skilled operators. Journal of Industrial and Systems Engineering, 12(4), 57-75.
[14] Dimitriadis SG. (2006). Assembly line balancing and group working: a heuristic procedure for workers' groups operating on the same product and work- station. Computers & Operations Research, 33; 2757–2774.
[15] Kellegoz T. & Toklu B. (2012) An efficient branch and bound algorithm for assembly line balancing problems with parallel multi-manned workstations. Computers & Operations Research, 39; 3344–3360
[16] Sepahi A. & Naini S. (2014).Multi-manned assembly line balancing problem with variable task times. European Journal of Academic Essays 68-75
[17] Cevikcan E, Durmusoglua MB, Unal ME. (2009) A team-oriented design methodology for mixed model assembly systems, Computers & Industrial Engineering, 56:576–599.
[18] Kazemi A. & Sedighi A. (2013). A Cost-oriented model for multi-manned assembly line balancing problem, Journal of Optimization in Industrial Engineering, 13; 13-25
[19] Fattahi P, Roshani A, Roshani A. (2011) A mathematical model and ant colony algorithm for multi-manned assembly line balancing problem. International Journal of Advanced Manufacturing Technology, 53:363–378
[20] Roshani A.,Roshani A.,Roshani A.; Salehi M., Esfandyari A. (2013) A simulated annealing algorithm for multi-manned assembly line balancing problem, Journal of Manufacturing Systems, 32; 238– 247
[21] Giglio, D., Paolucci, M., Roshani, A., & Tonelli, F. (2017). Multi-manned assembly line balancing problem with skilled workers: a new mathematical formulation. IFAC-PapersOnLine, 50(1),
[22] Roshani, A. & Giglio, D. (2017) Simulated annealing algorithms for the multi-manned assembly line balancing problem: minimising cycle time. International Journal of Production Research, 55(10), 2731-2751
[23] Chutima, P. & Prasert, K. (2018). Solving a multi-manned assembly line balancing problem in a Pareto sense. International Journal of Process Management & Benchmarking, 8(4), 490-515
[24] Yilmaz, H., &Yilmaz, M. (2020). A mathematical model and tabu search algorithm for multi-manned assembly line balancing problems with assignment restrictions. Engineering Optimization 52(5), 856-874.
[25] Zamzam, N., Sadek, Y., Afia, N., El-Kharbotly, A., (2015) Multi-Manned Assembly Line Balancing using Genetic Algorithm. International Journal of Engineering Research & Technology, 4(12), 56-61
[25] Ferrari, E., Faccio, M., Gamberi, M., Margelli, S., & Pilati, F. (2019). Multi-manned assembly line synchronization with compatible mounting positions, equipment sharing and workers cooperation. IFAC-PapersOnLine, 52(13), 1502-1507.
[27] Yadav, A. & Agrawal, S. (2019) A multi-manned parallel two-sided assembly line balancing with tool sharing approach-a company case study solved by exact solution approach. ISSN(P): 2249-6890; ISSN(E): 2249-8001, Vol. 9, Issue 2, 51-60
[28] Keun K, Yeongho K, Ju K. (2000) Two-sided assembly line balancing: a genetic algorithm Approach, Prod Plan Control, 11; 44–53.
[29] Baykasoglu, A. & Dereli, T. (2008) Two-sided assembly line balancing using ant-colony- based heuristic. International Journal of Advanced Manufacturing Technology, 36, 582–588
[30] Simaria AS, & Vilarinho PM. (2009). 2-ANTBAL: An ant colony optimization algorithm for balancing two-sided assembly lines. Computers and Industrial Engineering, 56; 489–506
[31] Gen, M. & Cheng, R. (2000). Genetic algorithms and engineering optimization (Vol. 7). John Wiley & Sons.
[32] Yang, C., Gao J., Sun L. (2013). A multi-objective genetic algorithm for mixed-model assembly line rebalancing, Computers& Industrial Engineering, 65; 109–116
[33] Akpınar S. & Bayhan G. (2011). A hybrid genetic algorithm for mixed model assembly line balancing problem with parallel workstations and zoning constraints Engineering Applications of Artificial Intelligence, 24; 449–457
[34] Baykasoglu A. (2006). Multi-rule multi-objective simulated annealing algorithm for straight and U type assembly line balancing problems, J Intell Manuf, 17; 217–232.
[35] Kim YK, Kim Y, Kim YJ. (2000) Two-sided assembly line balancing: a genetic algorithm approach. Prod Plan Control, 11:44–53.
[36] Lee T., Kim Y, Kim YK. (2001) Two-sided assembly line balancing to maximize work relatedness and slackness, Computer industrial engineering, 40:273–92
[37] Yuan, B., Zhang, C., Shao, X. (2015). A late acceptance hill-climbing algorithm for balancing two-sided assembly lines with multiple constraints. Journal of Intelligent Manufacturing, 26(1), 159-168