

# **Mixed-Model Two-Sided Assembly Line Balancing Using a Modified Genetic Algorithm**

**Raghda B. Taha**

**Department of Engineering Design and Production Technology  
Faculty of Engineering and Material Science  
German University in Cairo, Egypt**

**Amin El-Kharbotly, Nahid Afia and Yomna Sadek**

**Department of Design and Production Engineering  
Faculty of Engineering,  
Ain Shams University, Cairo, Egypt**

## **Abstract**

Mixed model two-sided assembly lines can be found today in many industrial environments. They are now-a-days replacing the traditional mass production assembly lines with the growing trend for greater product variability and shorter life cycles. In this paper a Modified Genetic Algorithm is used to solve the Mixed model Two-sided Assembly Line Balancing Problem (MTALBP) with the objective of minimizing the number of stations as well as the number of mated-stations for a given cycle time. The modified genetic algorithm was tested by solving benchmark problems from literature. The proposed algorithm obtained the best results for all test problems when compared with the results of other algorithms. The algorithm was able to find optimum or near optimum solutions within a limited number of iterations.

## **Keywords**

Mixed-model, Two-sided, Assembly line, Genetic Algorithm

## **1. Introduction**

The two-sided assembly line is a line where tasks on the same product item can be performed in parallel at both sides of the line. It provides shorter line length, reduced throughput time, lower cost of tools and fixtures, and less material handling (Kim et al. 2009). In some industries, due to the high cost to build and maintain a single model assembly line, more than one model of the same product is assembled on the same line which is known as Mixed Model Two Sided Assembly Line. This is a common practice followed in automobile industry. In a mixed-model two-sided assembly line, the line is designed to carry out a set of product models with similar production characteristics in any model sequence and model mix. Each model has its own set of task precedence relationships, which can all be combined into a single precedence diagram. The tasks in the combined precedence diagram are performed on a set of mated-stations each of which has a pair of two directly facing stations.

Bartholdi (1993) was the first to address the Two-sided Assembly Line Balancing Problem (TALBP) with the objective of minimizing the number of stations by applying a simple assignment rule. Lee et al. (2001) developed a group assignment procedure for TALBP where assignments were carried out based on task groups rather than individual tasks in order to maximize work relatedness and work slackness. Baykasoglu and Dereli (2008) proposed an ant-colony-based heuristic algorithm for solving TALBP with zoning constraints while minimizing the number of stations and maximizing the work relatedness within a given cycle time. The results indicated that the algorithm performed well in most of the test problems. Hu et al. (2008) developed a station-oriented enumerative assignment procedure integrated with the Hoffman heuristic to develop a system for solving the TALBP. The experimental results showed that the proposed procedure was efficient. Kim et al. (2000) was the first to address the TALBP using the GA with an objective of minimizing the number of stations within a given cycle time with positional constraints. The performance of the proposed GA was compared with the results of both the mathematical

model proposed by Kim et al. (1998), and the heuristic proposed by Bartholdi (1993). Kim et al. (2009) presented a mathematical model and a genetic algorithm for the TALBP with the objective of minimizing the cycle time. They adopted the strategy of localized evolution and steady state reproduction to promote population diversity and search efficiency. The proposed mathematical model found optimal solutions for small sized problems. The experimental results showed that the proposed GA outperformed the compared algorithms in terms of solution quality and convergence speed. Ozcan and Toklu (2009) proposed a tabu search algorithm for solving a TALBP with an objective to maximize the line efficiency. The computational results when compared with the results of various algorithms (Lee et al. 2001, Baykasoglu and Dereli 2008, Hu et al. 2008 and Kim et al. 2000). The comparison indicated that the proposed algorithm performed well but consumed longer computational time. Ozcan (2010) proposed a mathematical model (Chance-constrained, Piecewise-linear programming, Mixed Integer Programming) and a simulated annealing approach for solving the TALBP with stochastic task minimizing the weighted smoothness index.

Ozbakir and Tapkan (2011) adopted Bees Algorithm (BA) to solve TALBP with and without zoning constraints with the objective of minimizing the number of stations for a given cycle time. The results were compared with the results of several algorithms such as ant colony optimization (Baykasoglu and Dereli 2008, Simaria and Vilarinho 2009), tabu search (Özcan and Toklu 2009), enumerative algorithm (Hu et al. 2008), and group assignment procedure (Lee et al. 2001). The proposed Bees Algorithm obtained the best solutions for most of the test problems. Taha et al. (2011) also solved the TALBP with the objective of minimizing the number of stations as well as the number of mated stations. They proposed a Modified Genetic Algorithm (MGA) with modified operators and a new method for generating the initial population. Moreover, they developed new rules for assigning the “Either” tasks to the stations. The “Either” tasks are the tasks that does not require a certain operation direction. The developed algorithm proved its effectiveness compared with other algorithms ( Lee et al.2001, Baykasoglu and Dereli 2008, Hu et al.2008, Kim et al.2000, Özcan and Toklu 2009, Özbakir and Tapkan 2011 and Simaria and Vilarinho 2009) and was able to find optimum or near optimum solutions within a limited number of generations. The BA (Ozbakir and Tapkan 2011) and the MGA (Taha et al.2011) obtained the best results especially in large sized problems compared with previous algorithms used for solving the problem.

Few research works considered the MTALBP, Simaria and Vilarinho (2009) proposed a mathematical programming model for MTALBP with zoning and synchronism constraints. The objective was to minimize the number of workstations of the line and balancing the workloads between and within the workstations for the different models. Further, they also developed an ant colony optimization algorithm (2-ANTBAL) for balancing the mentioned problem. The 2-ANTBAL outperformed the procedure of (Lee et al. 2001) when tested on the available benchmark problems for single model TALBP. Ozcan and Toklu (2009) presented a mathematical model based on the model of (Kim et al. 2009) as well as a Simulated Annealing algorithm (SA) to solve the MTALBP Type-1 considering two objectives simultaneously, which are maximizing the weighted line efficiency and minimizing the weighted smoothness index. The SA was tested on the available benchmark problems for TALBP. The task times were randomly generated for each product model between zero and the maximum value of the original task times of the problem. The results were compared with the results of the mathematical model for the small sized problem and with lower bound for the large size problem. For small sized problems the SA performed well and obtained the optimum solution for all test problems. However, for large sized problems the maximum deviation from the lower bound was in the order of 40%.

In this paper, the MGA developed by the authors to solve the single model TALBP (Taha et al.2011) is adopted to solve the MTALBP with the objective of minimizing the number of stations as well as number of mated stations in order to increase the line efficiency. The performance of MGA is discussed and the results of several benchmark problems are compared with the results of previous algorithms used to solve the problem.

### **Notation**

The notations used in the mathematical formulations are given as follows in Table 1.

Table 1 Notation and abbreviations.

$i, j, h, k$	tasks number 1,2,3.....n
$n$	number of tasks
$R$	Right-side
$L$	Left-side
$E$	Either-side
$NM$	Number of Mated-stations
$NR$	Number of Right-side stations
$NL$	Number of Left-side stations
$NS$	Number of Stations where, $NS = NR + NL$
$P$	Precedence Matrix $P = [P_{i,j}]$ where, $P_{i,j} = \begin{cases} 1 & \text{if } j \text{ is a predecessor of } i \\ 0 & \text{otherwise} \end{cases}$
$t_{hm}$	processing time for task(h) for model(m)
$H$	planning horizon
$D_m$	Demand over the planning horizon for model(m)
$FT_h$	Finishing time for task $h$
$CT$	Cycle time
$TFT_{L,NM}$	Total finishing time for the left-side of the current mated-station
$TFT_{R,NM}$	Total finishing time for the right-side of the current mated-station
$TPT_{L,NM}$	Total processing time for tasks requiring the left-side of the current mated-station
$TPT_{R,NM}$	Total processing time for tasks requiring the right-side of the current mated-station
$PFT_{R,NM}$	Predecessor finishing time for the predecessors assigned to right-side of the current mated-station
$PFT_{L,NM}$	Predecessor finishing time for the predecessors assigned to left-side of the current mated-station
$WL_{L,NM}$	Workload of the left-side of the current mated-station
$WL_{R,NM}$	Workload of the right-side of the current mated-station
$SAT_{NM}(i)$	Set of assignable tasks(i) to the mated-station(NM)
$SLT_{NM}(i)$	Set of tasks(i) requiring the left-side of the mated-station(NM)
$SRT_{NM}(i)$	Set of tasks(i) requiring the right-side of the mated-station(NM)
$SET_{NM}(i)$	Set of tasks(i) that does not require a specific side of the mated-station(NM)
$TL_{NM}(i)$	Set of tasks(i) assigned to the left-side of the mated-station(NM)
$TR_{NM}(i)$	Set of tasks(i) assigned to the right-side of the mated-station(NM)
$SCT(i)$	Set of candidates tasks(i) that have no predecessors or all its predecessors have been assigned

## 2. Description of the MTALBP

In a two-sided assembly line, the products wait during the cycle time at each mated-station where there are two operators working at the opposite sides of the line simultaneously performing the different tasks on the same individual product. The tasks are performed according to certain operation sequence and may have restrictions on the operation directions. Some tasks should be performed on one of the two sides, while others could be performed at either sides of the line. Accordingly, the tasks are classified into three types: Left (L), Right (R), and Either (E) tasks. A task can only be assigned to a station if:

- The sum of the task time and the total time of the tasks performed before that task in the same station is less than or equal the cycle time.
- The sum of the task time and the finishing time of its predecessor in the opposite-side of that mated-station is less than or equal the cycle time.

In mixed-model two-sided assembly, the line is designed to produce a set of product models with similar production tasks. For each model, a precedence diagram defines the tasks precedence relationships, task time and task side. A combined precedence diagram is constructed for all models as illustrated in Figure 1. If a task exists in one model (model A) and does not exist in the other (model B), then it is to be included in the precedence diagram of model B with zero time. In a MTALBP, task  $h$  on a model  $m$  is performed in a certain time ( $t_{hm}$ ). The product models are produced over a pre-specified planning horizon,  $H$ . The demand over the planning horizon for model  $m$  is  $D_m$ . The required cycle time of the line is calculated from Eq. 1, and the overall proportion of the number of units of model  $m$  is computed from Equation 2. ( Simaria and Vilarinho 2009 and Özcan, and Toklu 2009)

$$\text{Cycle time} = \frac{H}{\sum_{m \in M} D_m} \quad (1)$$

$$\text{Proportion of units of model} = \frac{D_m}{\sum_{m \in M} D_m} \quad (2)$$

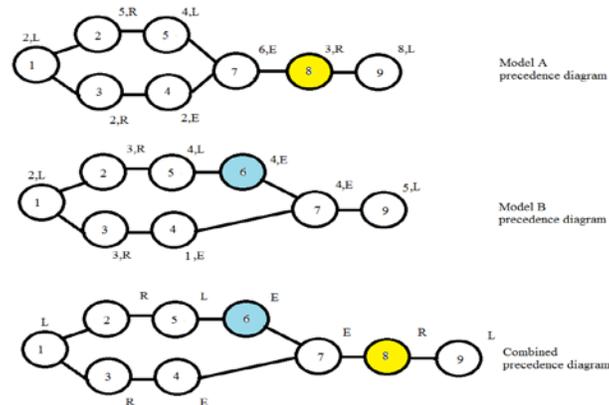


Figure 1: combining the precedence diagram of two products' models

### 3. The Modified Genetic Algorithm for the MTALBP

Genetic algorithms offer a particularly attractive optimization approach since they are generally quite effective for rapid global search of large, non-linear and poorly understood spaces (Floudas 2009 and Tasan and Tunali 2008). In the present work, a previously MGA developed for TALBP is adopted to solve the MTALBP. A station oriented assignment procedure is used to assign tasks to mated stations (Taha et al.2011). This procedure specifies the side of the either tasks based on certain rules rather than random assignments.

In the MTALBP, the operation sequence of tasks inside the stations are determined such that, a task based representation will be an efficient scheme for the GA chromosome. Each gene of the chromosome represents a task as illustrated in Figure2.

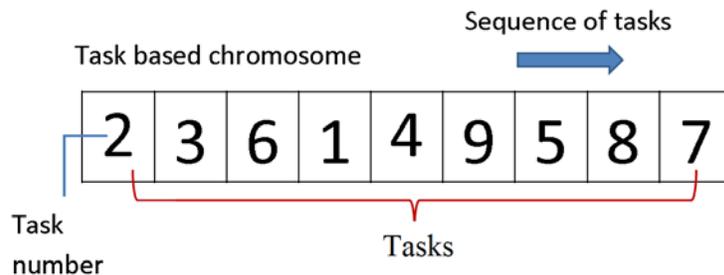


Figure 2: Chromosome representation (Taha et al.2011).

Each individual in the population represents a sequence of tasks by which they are assigned to the stations. All the generated individuals are made to represent feasible solution only for the problem. The individuals are then evaluated by assigning tasks to workstations according to a station-oriented procedure and determining the number of mated-stations as well as the number of stations. The sequence of the individual assignment must not be violated throughout the assigning procedure. The procedure starts assigning tasks to the first mated-station in both directions arbitrary to its left and right-sides directions. The other mated-stations are considered for task assignments sequentially by the same way. The tasks are assigned to the mated station under consideration according to the task-

sequence given in the chromosome. The left and right sides of the current mated station are filled with tasks as much as possible without violating the cycle time, after which the next mated station is opened and considered for assignment. The best mated stations assignment is represented by the chromosome with a sequence of tasks that minimizes the number of mated stations as well as the number of stations. In order to ensure feasible individuals, the initial population as well as the genetic operators is formulated according to task predecessors following that given in the precedence matrix. A flow chart of the GA is shown in Figure3. The following are the particulars of the applied modified genetic algorithm.

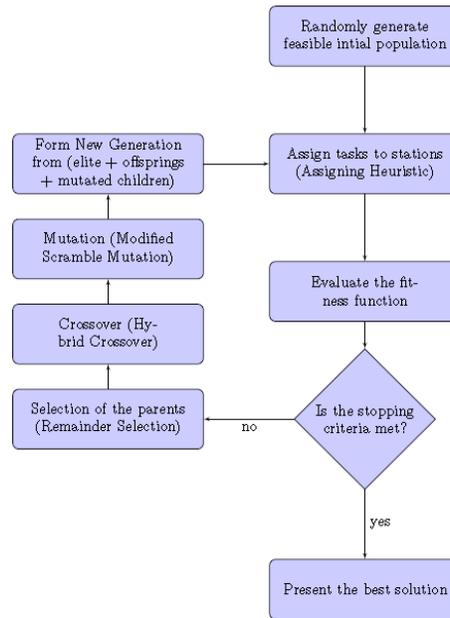


Figure 3: Flow chart of the MGA

### 3.1. Initial population

In order to ensure reasonable diversity in the in the initial population, it is generated randomly using the forward-backward method given in (Taha et al.2011). The forward method generates feasible individuals randomly according to the tasks precedence constrains. While, the backward method begins by assigning the chromosome from the last gene backward by choosing tasks randomly tasks that have no followers from the set of all candidate. Forming the initial population in this way was proven to increase the population diversity and helps in considering solutions in different scattered areas of the solution space avoiding trapping in a certain area.

### 3.2. Selection

The Remainder selection procedure is used to select parents for the crossover and mutation operations. The individuals of the population are scaled according to their fitness function. Parents are assigned deterministically from the integer part of each individual scaled value and then a roulette selection on the remaining fractional part is used.

### 3.3. Crossover

In the present MGA, a hybrid crossover operator is used to generate new individuals (Taha et al.2011). In this hybrid operator two types of crossover operators are used, namely; “Two Points Crossover” (Akpinar and Bayhan 2011) and Precedence Preservative Crossover (Bierwirth et al.1996). The performance of various crossover operators differs due to the way of inheriting genes structure from parents to offspring. The application of more than one type of crossover prevents premature convergence which is the loss of population diversity before optimal or at least satisfactory values are found. These two crossover operators are used simultaneously such that half of the new children are formulated by the first type and the other half by the second type.

### 3.4. Mutation

A modified scramble mutation is used to find mutated children (Taha et al.2011). The mutation operator is applied in the genetic algorithm in order to explore new areas in the search space and prevent pre-mature convergence. The mutated children are generated by both forward and backward methods as the initial population. The mutation method is chosen randomly for each mutated offspring.

### 3.5. Task-Assignment heuristic

Tasks are assigned to stations according to a station-oriented procedure given in the flow chart in Figure 4(Taha et al.2011). Each chromosome in the population represents a sequence of tasks that must not be violated. Tasks are selected according to their sequence and according to the preferred operation direction to fill this opened mated station as much as possible without exceeding the predetermined cycle time. Assigning tasks that do not dictate certain direction, "Either" tasks, is very exhaustive as for each sequence there are a huge number of possible sides assignments positions. To overcome this difficulty, a set of rules developed by the authors for the single-model TALBP are used to ensure that the best side for assignment has been selected for each sequence (Taha et al.2011). These rules reduce the solution space and the number of iterations needed, since that for each sequence there is only one possible assignment. These rules are mainly related to the workloads of the current mated station and to the predecessors of that task. The procedure is repeated until all tasks are assigned. In MTALBP the time of each task varies depending on the model being processed therefore the rules given in equations (3-8) are introduced in order to adopt the task assignment heuristic and side assignment rules to mixed model problems two sided assembly line problems.

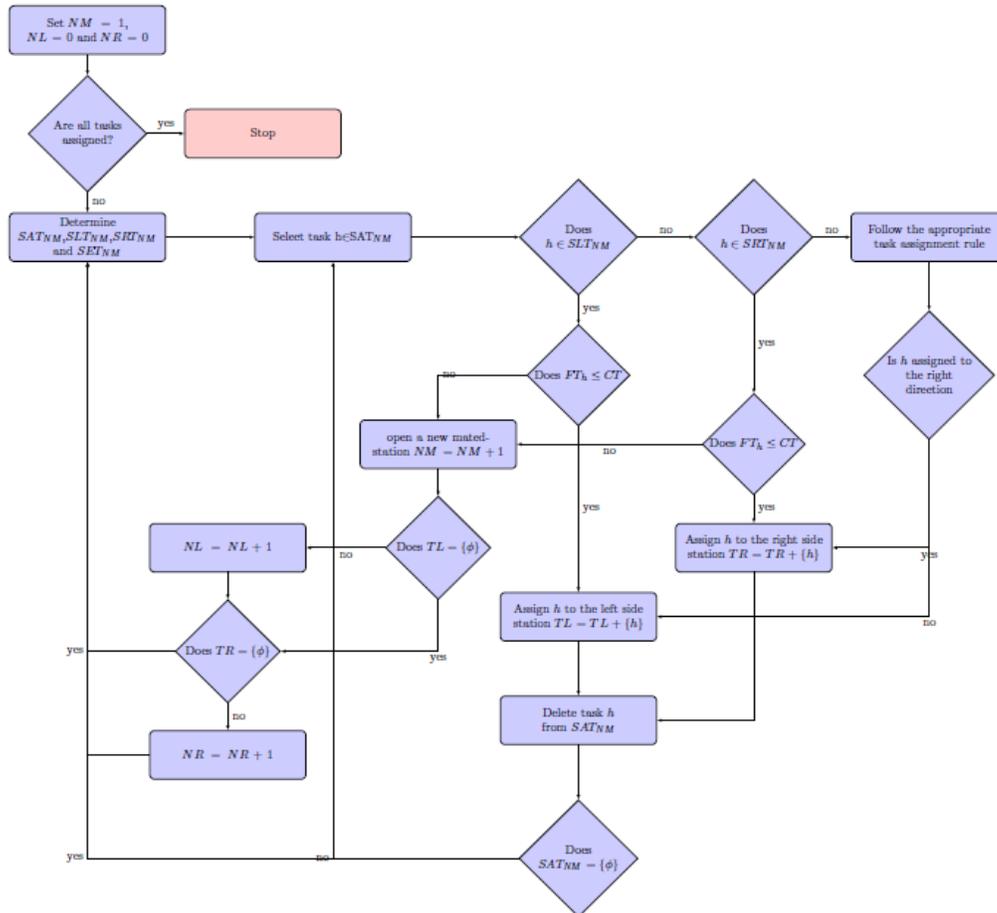


Figure 4: Task assignment heuristic flow chart

- The task finishing time of task  $h$  is calculated for each model  $m$  separately such that,

$$FT_{hm} = \text{Max}\{t_{hm} + WL_{i(R),NM}^m, t_{hm} + PFT_{R(L),NM}^m\} \forall m \in M \quad (3)$$

- The maximum task finishing time of all models is calculated from

$$FT_h = \text{Max}\{FT_{hm}\} \forall m \in M \quad (4)$$

This maximum is then compared to the cycle time to decide whether to consider opening a new station or not.

- The Total Processing time (TPT) of the successors that are candidates to the current mated station is calculated for the left side station for a certain model from;

$$TPT_{L,NM}^m = \sum_{i \in SLT_{NM}} t_{im} \quad \forall m \in M \quad (5)$$

- The maximum of all models for the left side station is calculated from;

$$TPT_{L,NM} = \text{Max}\{TPT_{L,NM}^m\} \forall m \in M \quad (6)$$

- The following equation calculates the total processing time of the tasks requiring the right side for a certain model.

$$TPT_{R,NM}^m = \sum_{i \in SRT_{NM}} t_{im} \quad \forall m \in M \quad (7)$$

- The maximum total processing time for all models is calculated from Eq. 8.

$$TPT_{R,NM} = \text{Max}\{TPT_{R,NM}^m\} \forall m \in M \quad (8)$$

### 3.6. Side assignment rules

According to the previous heuristic, tasks may have certain assigning direction, or can be performed on either sides of the line. For tasks requiring no direction, side assignment rules given in Figure 5 are used to effective assignment. The rules are applied sequentially for each of the Either tasks.

### 3.7 Fitness function

Genetic algorithms aim at finding the most fitted chromosome over a set of generations. The fitness function provides a measure of individual's performance. The MGA employs the function given in Eq.9 as the fitness function (Taha et al.2011). This function evaluates the number of mated-stations as well as the number of stations so that if there are two different solutions with the same number of mated-stations, one of these solutions may be better balanced than the other one, since one of them may have fewer stations than the other. Therefore, the number of stations should be considered as well as the number of mated-stations.

$$FF = NM + \frac{\sum_{n=1}^{NS} CT - WL_n}{NS} \quad (9)$$

### 3.8 Stopping Criteria

Several stopping conditions can be applied for the GA. In the MGA two conditions are applied, the first condition is reaching a determined number of generations as given in Table 2, and the second condition is the lower bound as formulated by (Hu et al. 2000) and modified by (Ozcan and Toklu 2009).

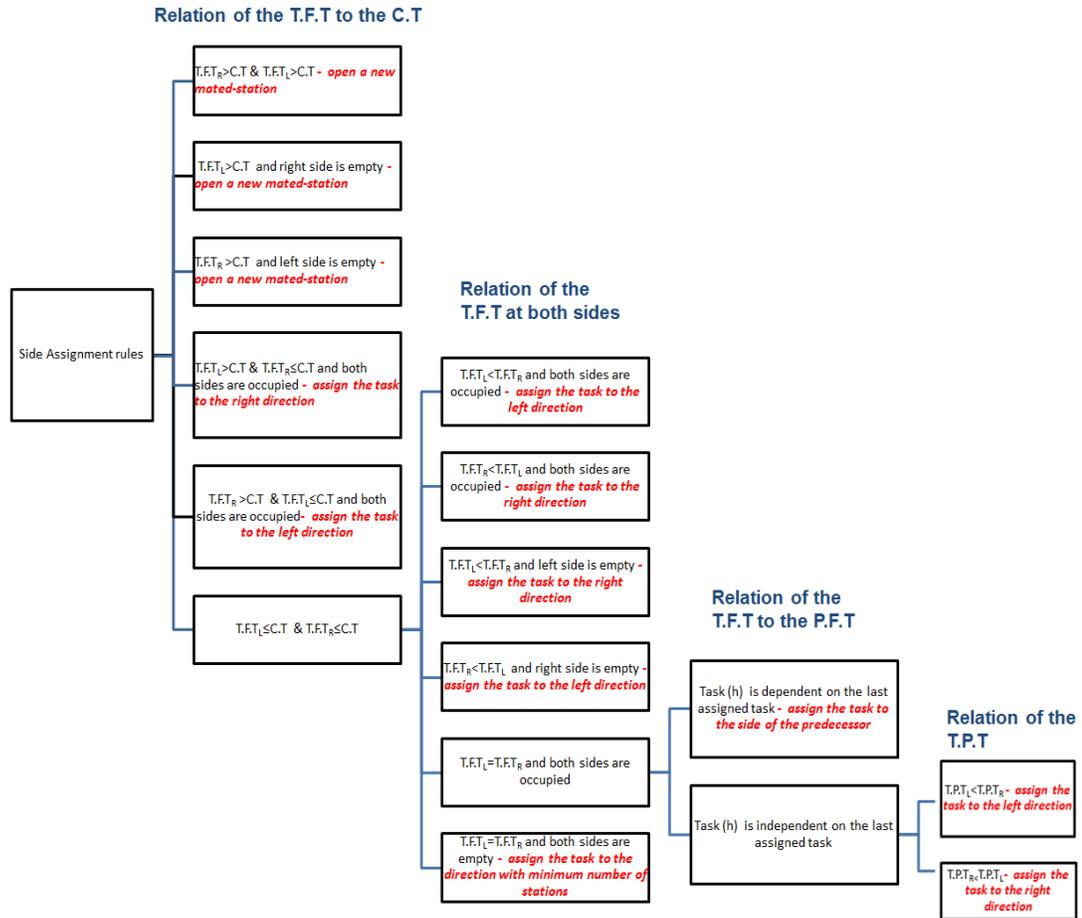


Figure 5: Side assignment rules

#### 4. Results and Discussion

The MGA was tested on the available benchmark problems for the MTALBP. The precedence constraints of P9, P12, and P24 are taken from (Kim et al.2000), while for P16, P65 and P148 are taken from (Lee et al.2001). The task times for the different models were provided by (Özcan and Toklu 2009). Each test problem was solved 20 times. The parameters of the MGA are given in Table 2. The best, the average, and the maximum number of stations (NS) are reported in Table 3 as well as the number of mated stations (NM) and the weighted line efficiency (WLE).

Table 2. MGA parameters

Parameter	Small-sized problems	Large-sized problems
Population size (Ps)	20	100
Number of generations	200	1000
Crossover rate (Rc)	0.8	0.8
Mutation rate (Rm)	0.2	0.2
Elite (e)	2	5

Table 3 lists the results obtained for MTALBP using the present MGA and the proposed assignments algorithms with that published in literature for a number of benchmark problems obtained by MIP and SA presented by (Özcan and Toklu 2009). In general, the results of the present technique are superior to the results of applying other techniques. The MGA obtained the same solution as the Simulated Annealing and Mixed Integer Programming in the problems P9, P12, P16 and P24 while it obtained the same solution as the Simulated Annealing in the problem P65. For P148 the MGA outperformed the Simulated Annealing algorithm in all instance of the problem as shown in

Figure 6. For cycle time 408 and 510 the number of stations decreased by two stations and the number of mated-station by one station while for the rest of the cycle time the improvement was within one station. It can be depicted from the statistical analysis of the results that they are consistent as in most of the cases (almost 85%), the technique gave the same number of stations in all runs. Moreover, the average of WLE for the results of large-sized test problems is 81.93% which is considered as great improvement compared with that of the SA which is 75.39%. The limited variations in the number of stations, in some of the instances, are due to the random nature of search as a prime feature of the GA technique.

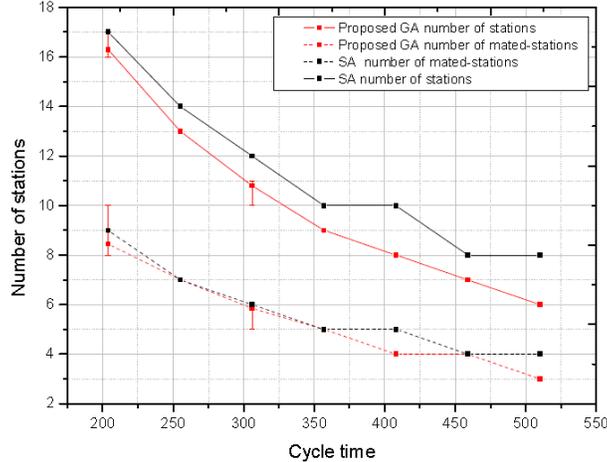


Figure 6: The number of stations and the number of mated-stations plotted against the cycle time for P148

Table 3. Results of the benchmark problems

Problem	Cycle time (unit time)	Lower bound of NS	initial line balance		MIP			simulated annealing			genetic algorithm				
			NM	NS	NS	NM	NS	NM	WLE	NSmin	NSavg	NSmax	NM	WLE	
P9 2m	4	4			4	3	4	3	78.12	4	4	4	3	78.12	
	5	3			3	2	3	2	83.33	3	3	3	2	83.33	
	6	3			3	2	3	2	69.44	3	3	3	2	69.44	
P12 2m	5	5			5	3	5	3	84	5	5	5	3	84	
	6	4			4	2	4	2	87.5	4	4	4	2	87.5	
	7	3			4	2	4	2	75	4	4	4	2	75	
P16 2m	8	3			3	2	3	2	87.5	3	3	3	2	87.5	
	15	5			6	4	6	4	75	6	6.05	7	4	75	
	16	5			6	4	6	4	70.31	6	6	6	4	70.31	
P24 2m	18	4			5	3	5	3	75	5	5	5	3	75	
	19	4			5	3	5	3	71.05	5	5	5	3	71.05	
	21	4			4	2	4	2	80.35	4	4	4	2	80.35	
	22	4			4	2	4	2	76.4	4	4	4	2	76.4	
	20	7			7	4	7	4	89.28	7	7.1	8	4	89.28	
P65 3m	24	6			6	3	6	3	78.12	6	6	6	3	78.12	
	25	6			6	3	6	3	86.8	6	6	6	3	86.8	
	30	5			5	3	5	3	83.33	5	5	5	3	83.33	
	35	4			4	2	4	2	89.28	4	4	4	2	89.28	
	40	4			4	2	4	2	78.12	4	4	4	2	78.12	
P148	326	8	6	11				9	5	84.39	9	9.7	10	5	84.64
	381	7	5	10				8	4	81.75	8	8	8	4	81.75
	435	6	4	8				7	4	81.83	7	7	7	4	81.83
	490	6	4	8				6	3	84.76	6	6	6	3	84.76
	544	5	4	8				6	3	76.34	6	6	6	3	76.34
P148	204	13	11	22				17	9	75.81	<b>16</b>	16.3	17	9	75.81

4m	255	11	9	18			14	7	73.64	<b>13</b>	<b>13</b>	<b>13</b>	7	79.163
	306	9	8	16			12	6	71.6	<b>10</b>	10.8	11	<b>5</b>	85.76
	357	8	6	12			10	5	73.64	<b>9</b>	<b>9</b>	<b>9</b>	5	81.676
	408	7	6	12			10	5	64.44	<b>8</b>	<b>8</b>	<b>8</b>	<b>4</b>	80.4
	459	6	5	10			8	4	71.6	<b>7</b>	<b>7</b>	<b>7</b>	4	81.676
	510	6	5	10			8	4	64.44	<b>6</b>	<b>6</b>	<b>6</b>	<b>3</b>	85.76

## 6. Conclusion

Mixed-model Two-sided Assembly Line Balancing Problem is one of the hard problems studied in literature. Exact and mathematical methods fail to find optimal solutions for large-size problems. The MGA adopted in this paper was able to obtain the best known solution for all of the test problems. It also outperformed other algorithms in solving the large-size problems. The results also showed that the side assignment rules were efficient especially in large-size problems. Moreover, the method used in generating the initial population was able to generate feasible solution in different areas of the search space. The applied hybrid crossover as well as the modified scramble mutation was able to preserve the feasibility of all solutions through all the generations. The techniques applied in the MGA were able to find optimum or near optimum solutions within a limited number of iterations.

## References

- Akpnar, S., and Mirac Bayhan, G., A hybrid genetic algorithm for mixed model assembly line balancing problem with parallel workstations and zoning constraints, *Engineering Applications of Artificial Intelligence* 24 (2011) 449–457.
- Bartholdi, J. J., Balancing two-sided assembly lines: A case study, *International Journal of Production Research* 31 (1993) 2447–2461.
- Baykasoglu, A., and Dereli, T., Two-sided assembly line balancing using an ant colony based heuristic, *International Journal of Advanced Manufacturing Technology* 36 (2008) 582–588.
- Bierwirth, C., Mattfeld, D., and Kopfer, H., On permutation representations for scheduling problems, in: H.-M.Voigt, W. Ebeling, I. Rechenberg, H.-P. Schwefel (Eds.), *Parallel Problem Solving from Nature U PSN IV*, volume 1141 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 1996, pp. 310–318.
- Floudas, P. P., and C.A., *Encyclopedia of Optimization*, Springer, 2009.
- Hu, X., Wu, E., and Jin Y., A station-oriented enumerative algorithm for two-sided assembly line balancing, *European Journal of Operational Research* 186 (2008) 435–440.
- Kim, Y. K., Kim, Y., and Kim, Y. J., Two-sided assembly line balancing: A genetic algorithm approach, *Production Planning and Control* 11 (2000) 44–53.
- Kim, Y. K., Song, W. S. , and Kim, J. H. , A mathematical model and a genetic algorithm for two-sided assembly line balancing, *Computers and Operations Research* 36 (2009) 853–865.
- Kim , Y. K., Kim, Y., and Lee, T. O., Two-sided assembly line balancing models, *Technical Report* (1998).
- Lee, T. O., Kim, Y., and Kim, Y. K., Two-sided assembly line balancing to maximize work relatedness and slackness, *Computers and Industrial Engineering* 40 (2001) 273–292.
- Özbakir, L., and Tapkan, P., Bee colony intelligence in zone constrained two-sided assembly line balancing problem, *Expert Systems with Applications* 38 (2011) 11947–11957.
- Özcan, U., and Toklu, B., A tabu search algorithm for two-sided assembly line balancing, *International Journal of Advanced Manufacturing Technology* 43 (2009) 822–829.
- Özcan, U., and Toklu, B., Balancing of mixed-model two-sided assembly lines, *Computers and Industrial Engineering* 57 (2009) 217–227.
- Özcan, U., Balancing stochastic two-sided assembly lines: A chance constrained, piecewise-linear, mixed integer program and a simulated annealing algorithm, *European Journal of Operational Research* 205 (2010) 81–97.
- Simaria, A. S., and Vilarinho, P. M., 2-antbal: An ant colony optimisation algorithm for balancing two-sided assembly lines, *Computers and Industrial Engineering* 56 (2009) 489–506.
- Taha, R. B., El-Kharbotly, A. K., Sadek, Y. M., and Afia, N. H., A Genetic Algorithm for solving two-sided assembly line balancing problems, *Ain Shams Engineering Journal*, Available online 2 December 2011, ISSN 2090-4479, 10.1016/j.asej.2011.10.003.
- Tasan, S. O., and Tunali, S., A review of the current applications of genetic algorithms in assembly line balancing, *Journal of Intelligent Manufacturing* 19 (2008) 49–69.