The final publication is available at IOS Press through http://dx.doi.org/10.3233/ATDE190091

# A Genetic Algorithm for Bi-Objective Assembly Line Balancing Problem

Amir NOURMOHAMMADI[1a], Masood FATHI[a], Enrique RUIZ ZÚÑIGA[a] and Amos H.C. NG[a]
[a] *Production and Automation Engineering Division, University of Skövde, Högskolevägen, Box 408, 541 28, Skövde, Sweden*

**Abstract.** Assembly line designs in manufacturing commonly face the key problem of dividing the assembly tasks among the working stations so that the efficiency of the line is optimized. This problem is known as the assembly line balancing problem which is known to be NP-hard. This study, proposes a bi-objective genetic algorithm to cope with the assembly line balancing problem where the considered objectives are the utilization of the assembly line and the workload smoothness measured as the line efficiency and the variation of workload, respectively. The performance of the proposed genetic algorithm is tested through solving a set of standard problems existing in the literature. The computational results show that the genetic algorithm is promising in providing good solutions to the assembly line balancing problem.

**Keywords.** Assembly line balancing, bi-objectives, genetic algorithm.

The final publication is available at IOS Press through http://dx.doi.org/10.3233/ATDE190091

## 1. Introduction

An assembly line is an order of production equipment, grouped into workstations that are linked together by a material-handling system. In the assembly line, the assembly process is defined as a sequence of activities performed by operators, machines, or robots in each workstation. The assembly line balancing problem (ALBP) is one of the most important decision problems arising at any shop floor when an assembly line has to be designed or reconfigured. The ALBP aims at partitioning the assembly operations among a set of workstations so that all stations on the assembly line have an equal amount of work to be carried out while a few constraints such as precedence relationships are satisfied.

ALBPs can be grouped into the simple ALBP (SALBP) and the generalized ALBP (GALBP) [1]. GALBP is the more complex version of SALBP where more real-world characteristics such as U-shaped line [2], mixed-model [3] and stochastic task times [4] are taken into consideration. SALBP has been among the well-studied problems in the line balancing context and most of the research on ALBPs have been dealing with SALBP [5]. There are two formulation types of SALBP: (1) SALBP-1, in which given a cycle time ($CT$) the number of workstations ($m$) is minimized and (2) SALBP-2, in which given the $m$, the $CT$ is minimized [6].

[1] Corresponding Author. nourmohammadi.amir@gmail.com, amir.nourmohammadi@his.se

The final publication is available at IOS Press through http://dx.doi.org/10.3233/ATDE190091

Any type of ALBPs fall within the NP-hard optimization problems [7,8]. Consequently, different heuristics and meta-heuristic approaches have been developed to address it. For instance, Fathi et al., [5] performed a comparative evaluation of heuristic rules in addressing the SALBP-1. An immune genetic algorithm (GA) was proposed by Zhang [9] for SALBP-1. Lalaoui and Afia [10] proposed a fuzzy generalized simulated annealing for SALBP-1. A hybrid approach based on ant colony algorithm and beam search was proposed by Huo et al., [11] to address SALBP-1. Furthermore, reviewing the SALBP literature shows a growing trend towards the improvement of workload balance at assembly stations while optimizing other objectives such as $CT$ or $m$. In this regard, aside from the basic objectives, other measures such as smoothness index [9] or variation of workload [12] have been optimized as secondary objectives. The readers interested in the application of soft computing methods and recent literature reviews on ALBPs are referred to [7] and [13].

Although several approximation methods are proposed in the literature to cope with SALBPs, finding a good solution within reasonable time still is of high importance [14]. Thus, in this study an efficient GA is proposed to address the bi-objective (BO) SALBP-1 in terms of line efficiency and variation of workload to find the (near) optimum solutions within a reasonable amount of time. The remaining of the paper is structured as follows. Section 2 provides a description of the problem. In Section 3, the proposed GA is presented. The computational results are presented in Section 4 while the concluding remarks are outlined in Section 5.

## 2. Assembly line balancing problem

The design of assembly line constitutes the optimal balancing of the assembly tasks among a set of stations ($m$) arranged along a material-handling system. This study aims to address the SALBP-1 while dealing with the bi-objectives (BO) namely (1) minimizing the number of stations ($m$) (which is equivalent to maximizing the efficiency of assembly line given a fixed $CT$) and (2) minimizing the variation of workload ($VW$) among stations. To this purpose, the set of assembly tasks $V = \{1,2, \dots, n\}$, each with a processing time of $t_j$, have to be uniformly balanced between stations $k = \{1, \dots, m\}$ so that (1) the time/workload of each station shown by $t(S_k) = \sum_{j\epsilon S_k} t_j$ ($S_k$ is the set of tasks assigned to station $k$) does not exceed the $CT$ and (2) the precedence relationships between tasks are satisfied. The precedence relationships can be shown by a graph $G = (V, E)$ in which $V$ is the set of nodes indicating the tasks and $E$ is the set of edges depicting the relations among tasks.

## 3. The proposed GA

Considering the NP-hard nature of the BO-SALBP-1, a meta-heuristic algorithm is proposed in this section. The main steps of the proposed GA are as follows:

- Initialization: read inputs ($CT$, precedence relationships and task times) and set GA parameters, generation of initial population
- Representation, encoding and decoding of each solution
- Fitness function evaluation

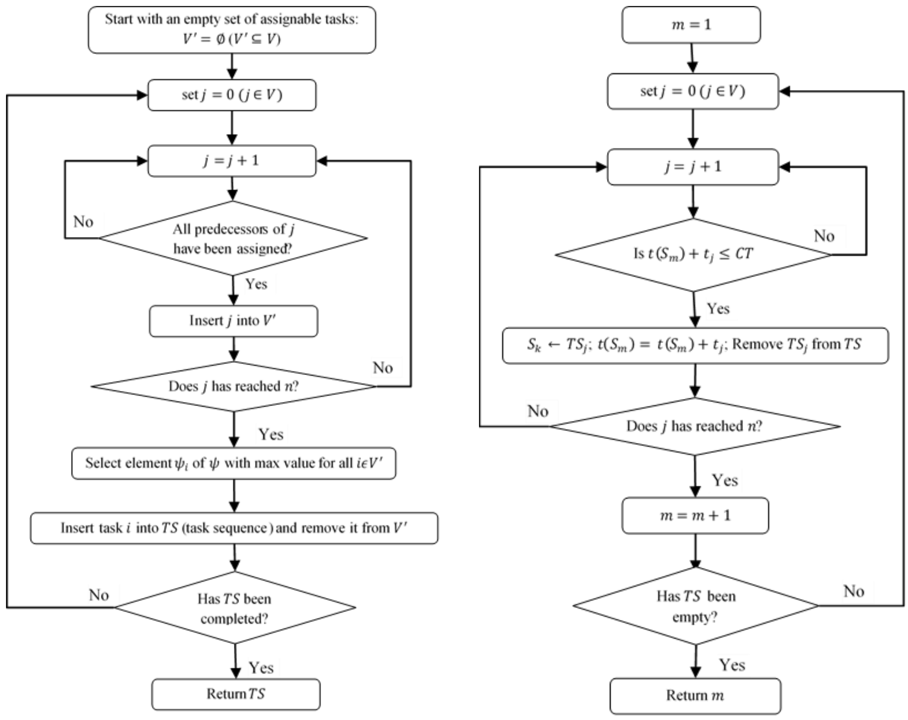
The final publication is available at IOS Press through http://dx.doi.org/10.3233/ATDE190091

- Crossover and mutation operators

In the following sub-sections, the descriptions of the GA elements are discussed in detail.

### *3.1. Representation, encoding and decoding methods*

Each individual solution of GA is represented by a permutation vector of integers between $[1, n]$ ($n$ is the number of tasks) known as task priority vector (i.e., $\psi$), where $\psi_i$ indicates the relative priority of the $i$th task. According to this representation, assuming that two tasks (e.g., $j$ and $j'$) are candid to be assigned to a station, the task with a higher relative priority will be chosen.
In order to ensure the generation of feasible solutions, an encoding method should be used to map each vector of task priorities ($\psi$) to an actual SALBP-1 solution. It is assumed a $n$-task SALBP-1 with predetermined relations between tasks shown by a graph defined as $G = (V, E)$. The encoding method aims to build a topological sort of $G$ upon a specific $n$-dimensional vector of task priorities $\psi$ [15]. Hence, topological sort is a sequence of tasks based on their priorities in an order known as task sequence ($TS$) to satisfy the precedence constraints. This method is described in **Figure 1**(a), where $V'$ is a subset of $V$ (i.e., the set of tasks) and $\psi$ is a task priority vector. Additionally, a decoding method is needed to assign the tasks in the resulted $TS$ into the stations [12]. The applied decoding method is shown in **Figure 1**(b), where $m$ is the established number of stations. The rest of notations have been defined in Section 2.




(a) (b)

**Figure 1.** (a) Encoding and (b) decoding methods.

### *3.2. The evaluation of fitness function*

To evaluate the fitness of GA solutions for the considered BO-SALBP-1, two objectives have to be calculated. (1) Maximizing the line efficiency ($LE$) given a $CT$, which is calculated by Eq. (1).

$$Max \;\; LE = \frac{t_{sum}}{m \times CT} \tag{1}$$

where $t_{sum}$ is the total sum of task times and $m$ is the number of stations obtained for the current solution. (2) Minimizing the variation of workload ($VW$) which is calculated by Eq. (2). $VW$ is a measure to determine the level of workload equalization at stations and ranges between [0,1]. The lower $VW$ shows a smoother distribution of workload at stations.

$$Min \quad VW = \sqrt{\frac{\sum_{i=1}^{m}(U_i - A)^2}{m}} \tag{2}$$

where $A = \sum_{i=1}^{m} U_i \,/m$ shows the average workloads at stations and $U_i = t(S_i)/\max_{i=1}^{m} t(S_i)$ is the workload at $i$th station.

The two above-mentioned objectives (Eq.s (1) and (2)) are merged in one fitness function as presented in Eq. (3).

$$Max \quad Fitnesss\ Function = LE + (1\text{-}VW) \tag{3}$$

### *3.3. Crossover and mutation operators*

Considering the applied representation, a typical crossover known as the weight mapping crossover operator is applied [15]. This operator can be viewed as two-point crossover of the parents' vectors by remapping and ordering of their genes. This crossover operator is shown in **Figure 2**(a). Furthermore, regarding the mutation operator two genes are chosen at random and swapped as presented in **Figure 2**(b).

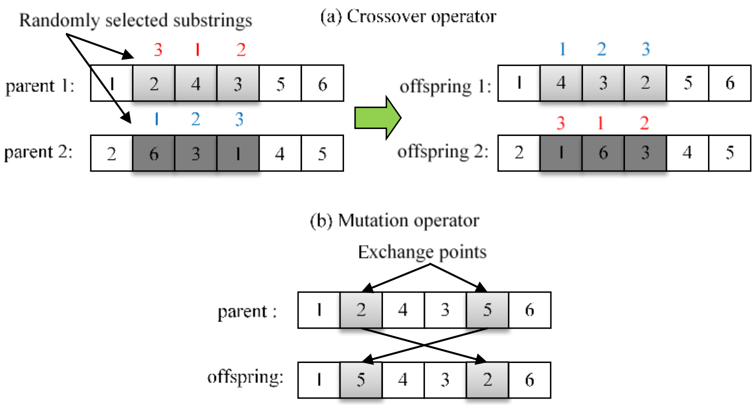



**Figure 2.** GA (a) crossover and (b) mutation operators.

The final publication is available at IOS Press through http://dx.doi.org/10.3233/ATDE190091

## 4. Computational results

To show the efficiency of the proposed GA in dealing with the BO-SALBP-1, it is applied on well-known SALBP-1 test problems that can be found at https://assembly-line-balancing.de. To evaluate the influence of the proposed GA on the quality of solutions, the problems are solved with and without considering the objective functions of the SALBP-1. To this purpose, the relating GA was applied on a Core i7 2.4 GHz PC to address the considered SALBP-1. The stopping condition of GA was chosen as when the best-found solution has been repeated for 100 generations without any improvement. Moreover, the GA parameters including the population size, the crossover ($Pr_{cr}$), mutation ($Pr_{mu}$) and reproduction ($Pr_{rep}$) rates were set to 100, 0.8, 0.15 and 0.05, respectively, based on some pilot studies.

**Table 1** shows the results of applying the proposed approach on the considered test problems by comparing (1) the feasible BO-SALBP-1 solutions with (2) the optimized BO-SALBP-1 solutions. To cope with the stochastic nature of the algorithm, the results are obtained after running the relating GA on the considered test problems for 10 times and the best-found solutions are reported in terms of $m$ and $VW$, representing the obtained number of stations and the variation of workload, respectively.

**Table 1.** The computational results of applying the proposed GA on different test problems

| Size | problem | *CT* | Feasible BO-SALBP-1 | | Optimized BO-SALBP-1 | |
|---|---|---|---|---|---|---|
| | | | *m* | *VW* | *m* | *VW* |
| Small | Jackson | 7 | 9 | 0.2469 | 8 | 0.1557 |
| | | 9 | 7 | 0.1957 | 6 | 0.1228 |
| | Mitchell | 14 | 10 | 0.2984 | 8 | 0.0428 |
| | | 15 | 9 | 0.3143 | 8 | 0.0909 |
| | Buxey | 36 | 11 | 0.1806 | 10 | 0.0388 |
| | | 41 | 9 | 0.1854 | 8 | 0.0122 |
| | Sawyer | 41 | 9 | 0.1981 | 8 | 0.0122 |
| | | 48 | 8 | 0.2592 | 7 | 0.0242 |
| Medium | Gunther | 44 | 15 | 0.1933 | 12 | 0.0777 |
| | | 49 | 11 | 0.1123 | 11 | 0.2029 |
| | Kilbridge | 62 | 10 | 0.0576 | 9 | 0.0076 |
| | | 69 | 9 | 0.1085 | 8 | 0 |
| Large | Arcus1 | 3786 | 22 | 0.1368 | 21 | 0.0289 |
| | | 4454 | 21 | 0.1913 | 18 | 0.0277 |
| | Tonge | 160 | 26 | 0.1961 | 23 | 0.0214 |
| | | 168 | 24 | 0.1743 | 22 | 0.0204 |

As one can observe, the best-found solutions obtained through the optimized BO-SALBP-1 scenario outperform the best-found solutions obtained by the feasible BO-SALBP-1 scenario in terms of $m$ and $VW$ over all test problems. This is due to the minimization of $m$ (or equivalently the maximization of $LE\%$) and the $VW$ by applying the proposed GA on the considered test problems.

## 5. Conclusion

Managers seek for the best solution methods to cope with the assembly line balancing problem (ALBP). This study attempts to optimize the configuration of assembly lines considering the simple assembly line balancing problem (SALBP) while dealing with two objectives namely line efficiency and variation of workload. To this purpose, a GA was proposed to cope with the considered SALBP-1. The obtained results on known test problems showed that the proposed GA is capable of providing promising solutions in terms of the line efficiency and the variation of workload.

The proposed GA in this study can be further developed to cope with other types of assembly line configurations such as U-shaped and two-sided lines. Moreover, it can also be used to address multi-model and mixed-model assembly lines. Additionally, the proposed GA can be hybridized with other meta-heuristics or local search algorithms.

## Acknowledgements

The authors would like to thank the industrial and support by the European Union's Horizon 2020 research and innovation program under grant No. 723711 through the MANUWORK project.

## References

[1] M. Fathi, A. Nourmohammadi, A. H. C. Ng and A. Syberfeldt, An optimization model for balancing assembly lines with stochastic task times and zoning constraints, *IEEE Access* **7**(1) (2019), 32537–32550.
[2] Z. Zhang, Q. Tang and L. Zhang, Mathematical model and grey wolf optimization for low-carbon and low-noise U-shaped robotic assembly line balancing problem, *Journal of Cleaner Production* **215** (2019), 744–756.
[3] P. Samouei and J. Ashayeri, Developing optimization & robust models for a mixed-model assembly line balancing problem with semi-automated operations, *Applied Mathematical Modeling* **72** (2019), 259–275.
[4] A. Nourmohammadi, H. Eskandari and M. Fathi, Design of stochastic assembly lines considering line balancing and part feeding with supermarkets, *Engineering Optimization* **51**(1) (2019), 63–83.
[5] M. Fathi, D. B. M. M. Fontes, M. Urenda Moris and M. Ghobakhloo, Assembly line balancing problem: A comparative evaluation of heuristics and a computational assessment of objectives, *Journal of Modeling in Management* **13**(2) (2018), 455–474.
[6] J. Pereira and E. Álvarez-Miranda, An exact approach for the robust assembly line balancing problem, *Omega* **78** (2018), 85–98.
[7] O. Battaïa and A. Dolgui, A taxonomy of line balancing problems and their solution approaches, *International Journal of Production Economics* **142**(2) (2013), 259–277.
[8] H. Zhang, C. Zhang, Y. Peng, D. Wang, G. Tian, X. Liu, Y. Peng, Balancing problem of stochastic large-scale U-Type assembly lines using a modified evolutionary algorithm, *IEEE Access* **6** (2018), 78414–78424.
[9] H. Zhang, An immune genetic algorithm for simple assembly line balancing problem of type 1, *Assembly Automation* **39**(1) (2019), 113-123.
[10] M. Lalaoui and A. El Afia, A Fuzzy generalized simulated annealing for a simple assembly line balancing problem, *IFAC-PapersOnLine* **51**(32) (2018), 600–605.
[11] J. Huo, Z. Wang, F. T. S. Chan, C. K. M. Lee and J. O. Strandhagen, Assembly Line Balancing Based on Beam Ant Colony Optimization, *Mathematical Problems in Engineering* **2018** (2018), 1–17.
[12] A. Nourmohammadi, M. Zandieh and R. Tavakkoli-moghaddam, An imperialist competitive algorithm for multi-objective U-type assembly line design, *Journal of Computational Science* **4**(5) (2013), 393–400.
[13] M. Fathi and M. Ghobakhloo, A technical comment on 'a review on assembly sequence planning and assembly line balancing optimisation using soft computing approaches,' *International Journal of Advanced Manufacturing Technology* **71**(9–12) (2014), 2033–2042.
[14] M. Fathi, M. J. Álvarez and V. Rodríguez, A new heuristic-based bi-objective simulated annealing method for U-shaped assembly line balancing, *European Journal of Industrial Engineering* **10**(2) (2016), 145-169.
[15] R. Hwang and H. Katayama, Uniform workload assignments for assembly line by GA-based amelioration approach, *International Journal of Production Research* **48**(7) (2010), 1857–1871.