

A review on simple assembly line balancing type-e problem

This content has been downloaded from IOPscience. Please scroll down to see the full text.

2015 IOP Conf. Ser.: Mater. Sci. Eng. 100 012005

(<http://iopscience.iop.org/1757-899X/100/1/012005>)

View [the table of contents for this issue](#), or go to the [journal homepage](#) for more

Download details:

IP Address: 103.53.34.15

This content was downloaded on 12/02/2016 at 03:38

Please note that [terms and conditions apply](#).

A review on simple assembly line balancing type-e problem

M Jusop¹ and M F F Ab Rashid¹

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

E-mail: masitahjusop@yahoo.com

Abstract. Simple assembly line balancing (SALB) is an attempt to assign the tasks to the various workstations along the line so that the precedence relations are satisfied and some performance measure are optimised. Advanced approach of algorithm is necessary to solve large-scale problems as SALB is a class of NP-hard. Only a few studies are focusing on simple assembly line balancing of Type-E problem (SALB-E) since it is a general and complex problem. SALB-E problem is one of SALB problem which consider the number of workstation and the cycle time simultaneously for the purpose of maximising the line efficiency. This paper review previous works that has been done in order to optimise SALB-E problem. Besides that, this paper also reviewed the Genetic Algorithm approach that has been used to optimise SALB-E. From the reviewed that has been done, it was found that none of the existing works are concern on the resource constraint in the SALB-E problem especially on machine and tool constraints. The research on SALB-E will contribute to the improvement of productivity in real industrial application.

1. Introduction

An assembly line is a manufacturing process comprises of a sequence of workstations in which a set of necessary task to assemble a product are performed. The development of assembly is system usually used in the production of goods in the industry. The idle time and the number of workstations on the production line have to be minimised whereas the line efficiency has to be maximised so as to achieve a balance line.

The decision problem of optimally partitioning the assembly task among the workstations with respect to some objective is known as Simple Assembly Line Balancing (SALB) [1]. This problem intends at grouping assembly operations which have to be performed to produce final products, and assigning the groups of operations to workstations, so as to make sure the total assembly time required at each station is nearly the same and the precedence constraints between operations are respected [2]. SALB is a type of NP-hard optimisation problems which means that when the number of assembly task is increased, the feasible solution will rise staggeringly [3-5]. Advanced approach of algorithm is necessary to solve large-scale problems.

SALB can be classified into two categories (i) Simple Assembly Line Balancing Problems (SALBP) (ii) General Assembly Line Balancing Problems (GALBP) [6, 7]. The most well-known assembly line is called simple assembly line balancing problem. Simple assembly line balancing is considered when the same product is running on the line. This type of problem is classified into four groups with respect to the objectives function [6, 8].



- Simple assembly line balancing Type-1 (SALB-1) aims to minimise the number of workstations on the line for a fixed cycle time.
- Simple assembly line balancing Type-2 (SALB-2) aims to minimise the cycle time for fixed number of workstations on the line.
- Simple assembly line balancing Type-E (SALB-E) aims to maximise the efficiency of the line simultaneously minimising the number of workstations and the cycle time.
- Simple assembly line balancing Type-F (SALB-F) aims to determine a feasible line for a combination of the number of workstations and cycle time.

Other problems which are not included in simple assembly line are considered as generalised assembly line balancing problems. Mixed-model assembly line balancing (MALBP) or mixed-model sequencing problem (MSP) and also U-line balancing problem (UALBP) are categorised as GALBP [7]. The classification of assembly line balancing problems is illustrated as in figure 1.

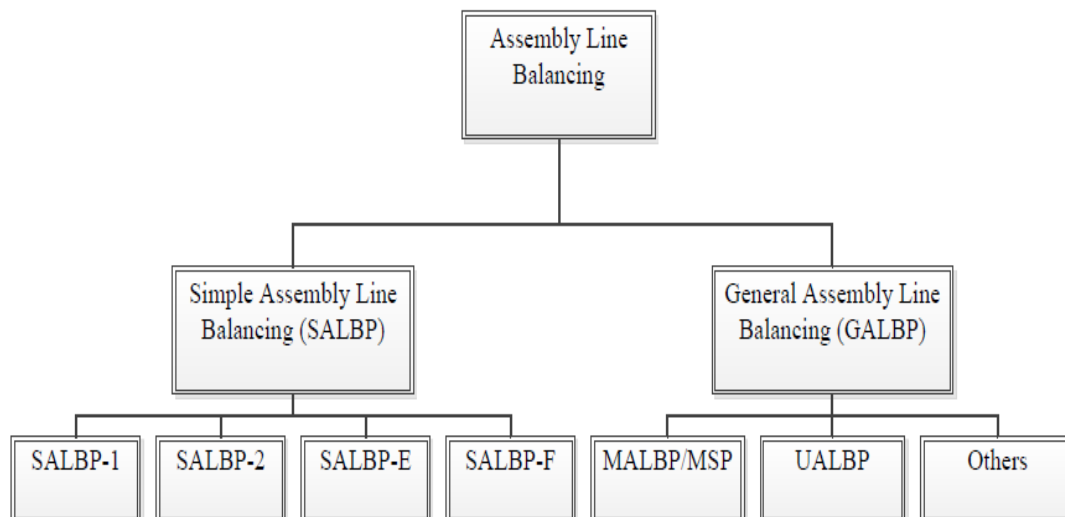


Figure 1. Classification of assembly line balancing problems.

Most of previous researches are focusing on SALB-1 [5, 6, 9, 10] and SALB-2 [2, 11-13]. Only a small number of previous research study on SALB-E as it is more complicated compare with SALB-1 and SALB-2. Study on SALB-E need to consider multi-objective functions instead of single objective in both SALB-1 and SALB-2. In real manufacturing scenario, it is better if we consider both parameters; minimised the number of workstations and minimised the cycle time for the purpose to maximise the assembly efficiency.

This paper reviews the previous study on simple assembly line balancing Type-E. The rest of the paper consists of problem modelling and objective function, SALB-E optimisation algorithm, and genetic algorithm for SALB-E. Finally, conclusion and suggestion for future research are addressed.

2. Problem modelling and objective function

Simple Assembly Line Balancing of Type-E Problem (SALBP-E) has been reviewed by Gurevsky et al. under dissimilarities of task processing times [14]. The research on stability of feasible and optimal solutions for SALBP-E is presented in this paper. Two heuristic procedures are proposed and evaluated on certain targets in order to find a concession between the two goal functions. Polynomial time algorithm has been proposed so as to compute the stability radius of feasible balances.

The paper presented by Suwannarongsri & Puangdownreong proposed a combination of partial random permutation (PRP) method and an adaptive tabu search (ATS) in an attempt to specify the optimum solutions for the assembly line balancing problem [15]. The researcher has considered the

simple assembly line balancing in the work with four objective functions (i) minimise the number of workstations, (ii) minimise the idle time, (iii) minimise the workload variance and (iv) maximise the line efficiency. The equation (1) is used to represent the line efficiency.

$$E = \sum_{i=1}^m T_i / (mc) \quad (1)$$

where E : Line efficiency

m : Number of workstations

c : Cycle time

T_i : processing time of the i^{th} workstation

A test against three benchmark single-model SALB problems such as Buxey, Sawyer, and Warnecke on actual SALB problem has been conducted by the researcher to assure the efficiency of the proposed multiple-objective method. The results shows that the proposed method is efficient for multiple-objective compare to the single-objective.

Previous study by Scholl & Becker stated that there is no direct method to solve the SALBP-E [6]. That type of model can be solved by a search method; the combination of the number of stations m and the cycle time c which is feasible for the efficient line is chosen among the others or, the value of required line capacity as in equation (2) should be minimal.

$$T = m.c \quad (2)$$

where T is line capacity

The review published by Wei & Chao are focused on SALBP-E in order to optimise the line balancing efficiency as well as minimising the idle time [16]. This objective can be achieved by minimising the number of stations and the cycle time. SALBP-1 and SALBP-2 models are combined by the researcher in order to develop the SALBP-E model. In SALBP-1, the number of stations is minimised with fixed cycle time. This model is re-defined to SALBP-1-i with the intention of determining the minimum number of stations. The goal of modified model SALBP-2 is to ensure the minimisation of cycle time ct with a fixed number of workstations m . The efficiency of the line is formulated as equation (3):

$$E = \frac{t_{sum}}{m.ct} \quad (3)$$

where t_{sum} is the total time of all tasks

In order to maximise the line efficiency, the optimal number of workstation must be obtained by a given ct_{max} . The value of ct_{max} must be less than or equal to the total task times and at the same time it also should be greater than or equivalent to the largest task time in data. Only one workstation will be required whenever the value of ct_{max} is exceed or the same as total task times. No solution will obtained as the value for ct_{max} is less than or equivalent to the largest task time in data. The respecting conditions are used for ct_{max} .

$$\max t_i \leq ct_{max} \leq \sum t_i$$

If $ct_{max} \geq \sum t_i$ then $m = 1, E = \frac{T_{total}}{1.T_{total}} = 1$ thus, Balance loss = 0

If $ct_{max} \geq \max t_i$, no solution

After the value of ct_{max} has been set, the optimal number of workstations m can be attained by using the spreadsheet. The value of m lies between m_{min} and m_{max} and it has been calculated as equation (4) and equation (5):

$$m_{min} = \left\lceil \sum_{i=1}^n \frac{t_i}{ct_{max}} \right\rceil \quad (4)$$

$$m_{max} = \left\lceil \frac{\sum t_i}{\max t_i} \right\rceil \quad (5)$$

where $m_{min} \leq m \leq m_{max}$

In another work, Zacharia & Nearchou minimised the number of workstations m and cycle time c using fuzzy task processing times so-called as f-SALBP-E [17]. The objective functions of the problem are to maximise the efficiency of the line, simultaneously minimising the number of workstations m and the cycle time c . The fuzzy efficiency \check{e} of the line is linearly dependent with summation of fuzzy processing times of all the task t_{sum} . It is also can be attained by minimising the product of number of workstations and fuzzy cycle time of the line. The line efficiency function is represented by equation (6):

$$\check{e} = \frac{t_{sum}}{m \cdot \check{c}} \quad (6)$$

where t_{sum} : total sum of the fuzzy processing time of all the tasks
 \check{c} : fuzzy cycle time of the line

The uncertainty and variability of task processing time and cycle time are presented by triangular fuzzy numbers (TFNs). A heuristic method based on Genetic Algorithm (GA) has been developed to solve the f-SALBP-E as it is a type of NP-hard optimisation problems. A two-phase GA is used for the purpose to solve the problem. In this approach, the optimal solution found from the first run is used to generate the early population of the binary run. There is no resource constraint being stated in the study. By considering the fuzzy processing time for the single assembly line balancing problem, a formulated mathematical model is performed and thus minimised the number of workstations and the fuzzy cycle time on the line.

A new genetic algorithm has been presented by Al-Hawari et al. to solve multi-objective simple assembly line balancing problem [18]. Minimisation of number of workstations, minimisation of workload variation, and maximisation of line efficiency are considered as the objective functions in the study. A Multi-Assignment Genetic Algorithm (MA-GA) has been proposed by the researcher with the combination of forward, backward, and bidirectional methods. The researcher concluded that the proposed algorithm has shown a better performance in solving multi-objective simple assembly line balancing for a larger size of problem. Equation (7) represents the line efficiency, E which is supposed to be maximised.

$$\max E = \frac{\sum_{i=1}^n t_i}{m \cdot c_a} \quad (7)$$

The efficiency of the line can be maximised by minimising both variables; the actual number of workstations m and the actual cycle time of the assembly line = $\max_{1 \leq k \leq m} \{t(S_k)\}$ whereas the sum of handling time of task i is fixed. The minimum number of actual workstations m can be obtained using the mathematical formulation as stated in equation (8):

$$\min m = \sum_{k=1}^M \max_{1 \leq i \leq n} \{x_{ik}\} \quad (8)$$

$$x_{ik} = \begin{cases} 1, & \text{if task } i \text{ is assigned to station } k \\ 0 & \text{otherwise} \end{cases}$$

$$t(S_k) = \text{the total time assigned to workstation } k$$

Suwannarongsri et al. has proposed a combination of tabu search (TS) and genetic algorithm (GA) to identify the solution for simple assembly line balancing problem [13]. The goals of the problem are to (i) minimise the number of workstations, (ii) minimise the workload variance, (iii) minimise the idle time and (iv) maximise the efficiency of the line. The maximum line efficiency can be calculated by using equation (9):

$$\max L_{eff} = \max \frac{\sum_{i=1}^n T_i}{(n \times ct_r)} \times 100 \quad (9)$$

where n : number of workstations

T_i : processing time of i^{th} workstation

ct_r : actual cycle time

L_{eff} : line efficiency

3. SALB-E optimisation algorithm

A two-part genetic algorithm (GA) is established to solve f -SALBP-E [17]. The first part of GA started with generating initial population, followed by performing the best solutions until it reached termination conditions. The optimal solution achieved from the first attempt is used as the source for the early population in the binary part for the purpose to find a better performance. The algorithm rises in a good feasible solution which is approximately to the exact solution in an acceptable time period.

The algorithm proposed by Al-Hawari et al. uses the combination of forward, backward, and bidirectional methods of task assignment [18]. These methods are used to assign each of tasks in a chromosome to workstations. Priority-based encoding, crossover, mutation, sequence encoding, decoding (assignment), evaluation, and selection are the primary procedures in MA-GA. As mentioned previously, the researcher simplified that the proposed MA-GA can solve problem for a larger size. It provides many feasible solutions of task assignments by combining the three methods simultaneously instead of combine using the only forward method. MA-GA will also increase the probability of identifying the optimal solution.

Suwannarongsri et al. used TSGA-based method which is the combination of TS and GA method to find the solutions for simple assembly line balancing problem. The researchers have performed a test of all type of SALBP problems from a literature against the proposed method. The result showed that the proposed TSGA-based method is capable in producing better solutions compared with conventional method [13].

Most of previous researcher used genetic algorithms (GAs) as an optimisation technique especially in SALB problem [2, 11, 12, 19, 20]. However, only a small number of researchers are focusing on simple assembly line balancing of Type-E problem [13, 14, 17]. As a consequence, the implementation of GA method has not been widely publicised in SALB-E itself.

4. Genetic algorithm for SALB-E

Genetic Algorithms (GAs) are mainly used by researcher for optimising large and complex problem specifically in SALB problem [2, 21-24]. GAs used a direct random search as an optimisation method

for complex problem with the aim of finding optimum solutions [21]. The application of genetic algorithm is quite popular compare with the simulated annealing and ant colony optimisation [25].

In [17], the design of GA comprises of chromosome's encoding, a decoding mechanism, an evaluation mechanism, generation of early population, and generation of offspring. The solution for f -SALBP-E is characterised by chromosome's encoding, which consists of tasks priorities (first part of the chromosome) and number of workstations on the line (binary part of chromosome). The tasks are then assigned to workstations by using a suitable decoding scheme. In evaluation mechanism, an individual chromosome with higher fitness value tends to have higher probability to be selected. The feasible tasks provide a better solution for the problem as it has low values of total fuzzy idle time.

The early random population undergoes selection, crossover, and mutation process to produce new generation. The optimum solution obtained from the first part is used as the source for the early population in second part for the aim of finding a better solution. A roulette wheel method is used in selection process. Chromosomes with higher fitness value will be selected to produce new population. Crossover operator is developed to produce new chromosomes from two parents' chromosomes by changing the tasks order. In GA, mutation mechanism worked by flipping or swapping an only chromosome to produce a single new chromosome.

Previous paper presented by Al-Hawari et al. used three assignment methods (i) forward (ii) backward and (iii) bidirectional in Multi-Assignment Genetic Algorithm (MA-GA) [18]. A forward assignment method is the mainly used for solving SALBP. By using this technique, the works are allocated sequentially to workstations by taking into consideration the cycle time constraint. In backward assignment, a flipping method is used. The task sequence chromosome is flipped to be assigned using forward assignment method whereas, the bidirectional assignment method used both forward and backward directions. From the acquired result, bidirectional assignment attained the best solution.

Three genetic operators that have been used in GA are (i) crossover (ii) mutation and (iii) selection. The researcher used weight mapping crossover operator (WMX), swap mutation operator, and roulette wheel selection (RWS). The crossover operates two chromosomes (parent) to produce a new chromosome. One-point WMX is used in the proposed MA-GA and one crossover cut has been pointed at anyplace along the length of the parent, producing two offspring that have their genes. In the research, the swap mutation operator is used in order to keep the genetic diversity. In selection step, the roulette wheel selection method has been applied to produce a new population. The chromosomes with higher fitness value get more chances to be selected. To avoid the loss of the best chromosome(s), an elitism approach is adopted while using the RWS.

Suwannarongsri et al. used TS method to determine the number of tasks assign in each workstation whereas GA is employed to assign the sequence of tasks for each workstation by considering the precedence constraints [13]. The searching process of the GA is comparable to the nature development of biological beings. The flowchart of GA is summarised as in figure 2.

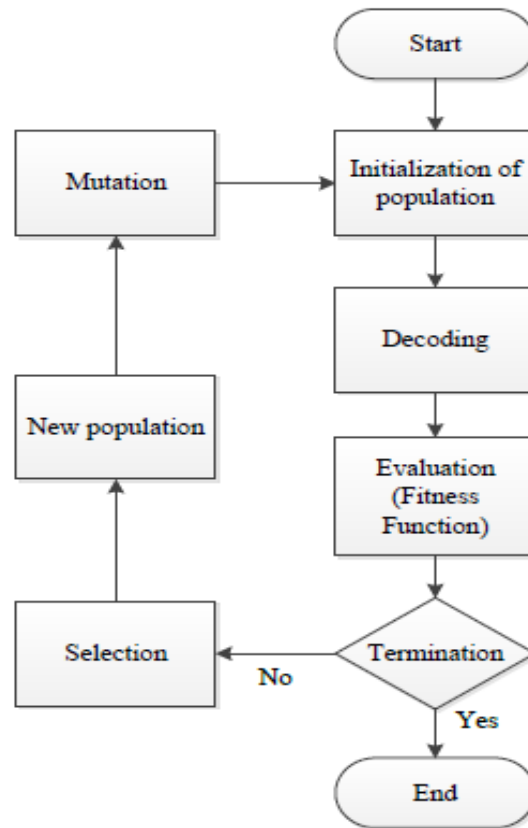


Figure 2. Flowchart of genetic algorithm.

5. Conclusion

This paper reviewed the optimisation algorithm and techniques used by the previous researcher on SALB-E. From literature review that have been conducted, it can be concluded that the application of genetic algorithm (GA) as an optimisation technique are on the rise due to its ability to solve a large-scale optimisation problem as well as searching near optimal solution.

Only a few studies are focusing on SALB-E as it is a general and complex problem. Up till now, none of them are concern on the resource constraint in the problem especially machine and tool constraint. Future research direction could be to consider recourse constraint in the optimisation of SALB-E itself.

Acknowledgement

This study was supported by a research grant from the Ministry of Higher Education under RDU140103. The authors would like to thank Universiti Malaysia Pahang for the facilities provided.

References

- [1] Rashid MFF, Hutabarat W, Tiwari A. A review on assembly sequence planning and assembly line balancing optimisation using soft computing approaches. *The International Journal of Advanced Manufacturing Technology*. 2012;59:335-349.
- [2] Gu L, Hennequin S, Sava A, Xie X. Assembly line balancing problems solved by estimation of distribution. *Automation Science and Engineering, 2007 CASE 2007 IEEE International Conference on*: IEEE; 2007. p. 123-127.

- [3] Kriengkarakot N, Pianthong N. The Assembly Line Balancing Problem. *KKU Engineering Journal*. 2007;34:133-140.
- [4] Nearchou AC. Multi-objective balancing of assembly lines by population heuristics. *International Journal of Production Research*. 2008;46:2275-2297.
- [5] Emeke Great O, Offiong A. Productivity Improvement In Breweries Through Line Balancing Using Heuristic Method. *International Journal of Engineering Science & Technology*. 2013;5.
- [6] Scholl A, Becker C. State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research*. 2006;168:666-693.
- [7] Boysen N, Flidner M, Scholl A. A classification of assembly line balancing problems. *European Journal of Operational Research*. 2007;183:674-693.
- [8] Becker C, Scholl A. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research*. 2006;168:694-715.
- [9] Chong KE, Omar MK, Bakar NA. Solving assembly line balancing problem using genetic algorithm with heuristics-treated initial population. 2008.
- [10] Ponnambalam S, Aravindan P, Naidu GM. A multi-objective genetic algorithm for solving assembly line balancing problem. *The International Journal of Advanced Manufacturing Technology*. 2000;16:341-352.
- [11] Zhang W, Gen M, Lin L. A multiobjective genetic algorithm for assembly line balancing problem with worker allocation. *Systems, Man and Cybernetics, 2008 SMC 2008 IEEE International Conference on: IEEE; 2008*. p. 3026-3033.
- [12] RuiJun Z, DingFang C, Yong W, ZhongHua Y, Xinxin W. Study on line balancing problem based on improved genetic algorithms. *Wireless Communications, Networking and Mobile Computing, 2007 WiCom 2007 International Conference on: IEEE; 2007*. p. 2033-2036.
- [13] Suwannarongsri S, Limnararat S, Puangdownreong D. A new hybrid intelligent method for assembly line balancing. *Industrial Engineering and Engineering Management, 2007 IEEE International Conference on: IEEE; 2007*. p. 1115-1119.
- [14] Gurevsky E, Battaia O, Dolgui A. Balancing of simple assembly lines under variations of task processing times. *Annals of Operations Research*. 2012;201:265-286.
- [15] Suwannarongsri S, Puangdownreong D. Multi-objective assembly line balancing via adaptive tabu search method with partial random permutation technique. *Industrial Engineering and Engineering Management, 2008 IEEM 2008 IEEE International Conference on: IEEE; 2008*. p. 312-316.
- [16] Wei N-C, Chao I-M. A solution procedure for type E simple assembly line balancing problem. *Computers & Industrial Engineering*. 2011;61:824-830.
- [17] Zacharia PT, Nearchou AC. A meta-heuristic algorithm for the fuzzy assembly line balancing type-E problem. *Computers & Operations Research*. 2013;40:3033-3044.
- [18] Al-Hawari T, Ali M, Al-Araidah O, Mumani A. Development of a genetic algorithm for multi-objective assembly line balancing using multiple assignment approach. *The International Journal of Advanced Manufacturing Technology*. 2014:1-14.
- [19] Gonçalves JF, De Almeida JR. A hybrid genetic algorithm for assembly line balancing. *Journal of Heuristics*. 2002;8:629-642.
- [20] Sabuncuoglu I, Erel E, Tanyer M. Assembly line balancing using genetic algorithms. *Journal of intelligent manufacturing*. 2000;11:295-310.

- [21] Ranjan R, Pawar P. Assembly Line Balancing Using Real Coded Genetic Algorithm. *International Journal of Scientific Research in Computer Science and Engineering*. 2014;2:1-5.
- [22] Matondang MZ, Jambak MI. Soft computing in optimizing assembly lines balancing. *Journal of Computer Science*. 2010;6:141.
- [23] Tasan SO, Tunali S. A review of the current applications of genetic algorithms in assembly line balancing. *Journal of intelligent manufacturing*. 2007;19:49-69.
- [24] Mohd Razali N, Geraghty J. Biologically inspired genetic algorithm to minimize idle time of the assembly line balancing. *Nature and Biologically Inspired Computing (NaBIC)*, 2011 Third World Congress on: IEEE; 2011. p. 105-110.
- [25] Battaia O, Dolgui A. A taxonomy of line balancing problems and their solution approaches. *International Journal of Production Economics*. 2013;142:259-277.