

Mathematical Modelling of Mixed-Model Assembly Line Balancing Problem with Resources Constraints

Muhamad Magffierah Razali, Mohd Fadzil Faisae Ab.Rashid, Muhammad Razif Abdullah Make

Faculty of Mechanical Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia

E-mail: magffierah_razali@yahoo.com, ffaisae@ump.edu.my, razifabdullahmake@gmail.com

Abstract. Modern manufacturing industries nowadays encounter with the challenges to provide a product variety in their production at a cheaper cost. This situation requires for a system that flexible with cost competent such as Mixed-Model Assembly Line. This paper developed a mathematical model for Mixed-Model Assembly Line Balancing Problem (MMALBP). In addition to the existing works that consider minimize cycle time, workstation and product rate variation, this paper also consider the resources constraint in the problem modelling. Based on the finding, the modelling results achieved by using computational method were in line with the manual calculation for the evaluated objective functions. Hence, it provided an evidence to verify the developed mathematical model for MMALBP. Implications of the results and future research directions were also presented in this paper.

1. Introduction

Over the year, assembly line balancing (ALB) problem has earned a lot of attention. The purpose of ALB is to distribute different tasks to the operators for a various workstation on the line in a way where the tasks do not violate any of the precedence restrictions and some measurements of effectiveness are being optimized [1]. ALB has been evaluated widely in the relevant literature by Becker and Scholl [2, 3]. Mixed-Model ALB is categorized under general assembly line balancing which produce several models having similar characteristics on a single assembly line [4]. Usually in a mixed-model assembly line, the models being assembled have differences in the set of tasks associated with each model, the processing times, precedence relations, and amount of production. With all due respect to all of these conditions, Mixed-Model Assembly Line Balancing Problem (MMALBP) has been categorized as an NP-hard combinatorial optimization problem as well as CPU time-consuming [5].

Mixed-model assembly lines solution procedures in the relevant literature was proposed by Thomopoulos [6]. Thomopoulos classify the procedures into three categories: meta-heuristics and heuristics, hybrid, and mathematical model solutions. In general, there are MMALBP-I and MMALBP-II, which aim to minimize number of workstations for a given amount of cycle time and to minimize cycle time for a given number of workstations respectively [7]. Due to the current scenario of global market, companies changed single model lines into mixed-model lines in order to provide diversity and



meet customized customer needs on time in a perceptive manner [8]. In addition to that advantage, mixed-model lines do not require a new set-up process between model changes, provide a continuous flow of materials, reduce the inventory levels of final items, and very flexible with respect to model changes [9].

In MMALBP context, different problem model has been proposed with various objective functions. A mixed integer linear programming model has been proposed by Akpinar and Baykasoglu with the objectives to minimize the total number of workstations subjected to assignment, capacity and zoning constraints [10]. Two objectives being handled by Yagmahan that utilized an approach of multi-objective ant colony optimization which is maximizing the smoothness index between stations and minimizing the number of workstations by considering the precedence constraints [11]. Tiacci considers buffer allocation problem and assembly line balancing problem simultaneously by using genetic algorithm approach. The objective function called normalize design cost (NDC) is introduced and subjected to precedence restriction [12].

The mixed-model parallel two-sided assembly considered by Kucukkoc consist an objective function of minimize total number of utilised workstations, as well as to ensure a smooth workload (WS) among the stations from cycle to cycle and the constraints are model and task occurrence constraint, task assignment, and operation direction constraint [13]. Ant colony optimization algorithm for balancing mixed-model assembly lines (ANTBAL) is used by Vilarinho and Simaria together with an objective function to balance the workloads within each workstation and maximize weighted line efficiency by considering zoning and capacity constraints [14]. Hamzadayi aims at minimizing the number of the stations required on the line, smoothing the workload of stations between cycles as well as smoothing the workload of all stations within any cycle by take into account the restriction of parallel and zoning constraints [15].

Based on the literature review, there are some published works considering the resource constraint in their work but limited to a specific constraint. This paper will focus to propose a mathematical model for MMALBP-II to minimize general resource constraints along with the cycle time and product rate variation. MMALBP-II has been chosen because one of the objective functions in this paper is to minimize the cycle time for a given number of workstations and categorized under MMALBP-II problems.

2. Problem description and formulation

In this paper, the MMALB problem is formulated with the aims to minimize total cycle time, resources and product rate variation (PRV) by considering the resources constraint. In order to explain the problem formulation, an example of assembly problem consists of six tasks for three different models as shown in Fig.1. The general assumptions of the problem are as follows:

- A number of J models will be assembled on a mixed model assembly line.
- Assembly tasks for different models are almost similar, so we can suppose a combine precedence diagram for all of the models. Now, if some models do not use some tasks in their assembly process, the relevant task time will be 0.
- Operating or processing times related to a task is the same for each different models.
- Each task type is assigned to only one station regardless of models. Hence, tasks are not assigned to different stations for different models.
- Each operator works only on a single station, only one operator carries out the assembly tasks on each station and the tasks are undividable.
- Assembly models will be assembling with the same rate and consecutively.

By referring the above mentioned assumptions, the parameters and indices of the model will be as:

Notation	Definition
S	number of workstations(fixed) $s=1,2,\dots, S$
J	number of product models to be assembled $j=1,2,\dots, J$
Ne	number of task $e=1,2,\dots, Ne$
pre_i	predecessor for task i based on precedence diagram
t_i	execution time for task i
D_T	total quantity of units or total demand
d_j	demand for product j , $j = 1, 2, \dots, a$
$X_{i,k}$	total quantity of product/produced over stages 1 to k , $k = 1, 2, \dots, Dt$
$maxR$	maximum resources $r=1,2,\dots,maxR$
C_T	cycle time
Te_j	shift task model time
Te	shift task time
te	task time
N_j	demand schedule for each model
U	production rates variation of production sequence

Decision variables

U_{ej}	1 if task, e is used on model j ; 0,otherwise
X_{es}	1 if task, e is assigned to workstation s ; 0, otherwise
Y_{rs}	1 if resource, r is used in workstation s ; 0,otherwise

2.1. Mathematical modelling

A sample problem used in this paper for a modelling purpose is adapted based on Thomopoulos [16]. The assembly data being considered are consist of six tasks for three different models as shown in Fig.1 and indirectly depict the build relationship among all the tasks. The models consist of six tasks (denoted by 1 to 6). The task is represented by circles and the connecting arrows identify the immediate predecessor tasks. It shows which tasks can begin without any predecessor tasks, and which tasks have predecessor tasks. The sequence of tasks moves from left to right.

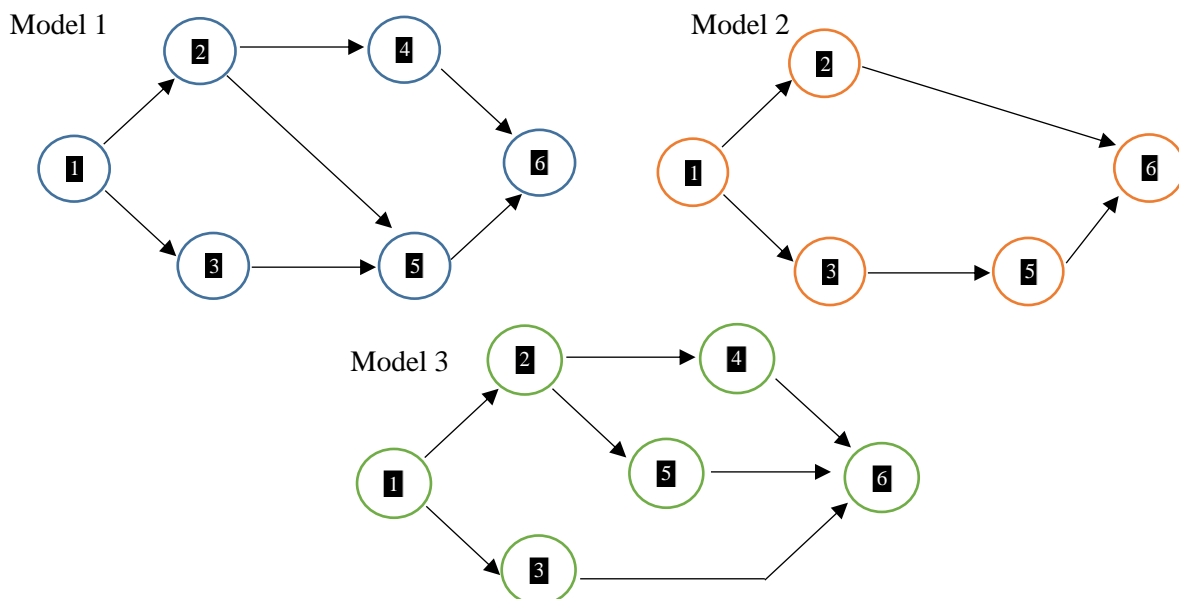


Figure 1. Precedence diagram for model 1, 2 and 3

Model 1: Task 1 on the left-hand side of the diagram and have no predecessor tasks. Task 2 and 3 cannot begin until Task 1 is completed. Also for instance, Task 5 cannot begin until Task 2 and 3 are completed. Lastly, Task 6 cannot begin until Task 4 and 5 are completed.

Model 2: Task 1 on the left-hand side of the diagram and have no predecessor tasks. Task 2 and 3 cannot begin until Task 1 is completed. Task 5 cannot begin until Task 3 is completed. Lastly, Task 6 cannot begin until Task 2 and 5 are completed.

Model 3: Task 1 on the left-hand side of the diagram and have no predecessor tasks. Task 2 and 3 cannot begin until Task 1 is completed. Task 4 and 5 cannot begin until Task 2 is completed. Lastly, Task 6 cannot begin until Task 3, 4 and 5 are completed.

The other important data are needed such as assembly time, type of resources used on the line, and model demands. Each model has their own task time as well as precedence relation but with the help of the anticipated model mix, a joint precedence graph is deduced from Figure 1.

2.2. Objectives function

The first objective function considered is to minimize cycle time. In our work, the cycle time is based on the product demand. Second objective function is minimizing total number of resource. The usage of resources such as tool, worker, and workstation is inevitable in an assembly line. However, based on the literature review less attention has been paid to minimize total number of resources even though the effect it can give to the operation cost is significant. Third objective function is product rate variation (PRV) which is common problem in the MMALB based on [17]. The problem formulation for this problem is presented as follows:

$$f1 = \min \sum_{e=1}^{N_e} \sum_{j=1}^J C_T \quad (1)$$

$$f2 = \min \sum_{s=1}^S \sum_{r=1}^{max_R} Y_{rs} \quad (2)$$

$$f3 = \min \sum_{k=1}^{D_T} \sum_{j=1}^J \left(x_{j,k} - k \times \frac{d_j}{D_T} \right)^2 \quad (3)$$

Objective function 1, $f1$ in equation (1) is aim to minimize cycle time for a given number of workstation. Equation (2) targeted to minimize resources used on assembly line and equation (3) aim to minimize product rate variation (PRV). These objective functions are subjected to these constraints:

2.3. Constraints

$$\sum_{s=1}^S X_{es} = 1, \quad e = 1, \dots, n \quad (4)$$

$$\sum_{s=1}^S X_{as} - \sum_{s=1}^S X_{bs} \leq 0, \text{ for } \forall (a, b) \in pre_i \quad (5)$$

$$\sum_{i \in wk} t_i(X_{es}) \leq C, \quad s = 1, \dots, S \quad (6)$$

Constraint (4) is to ensure that each task can only be assigned to one workstation [18]. Inequality (5) describes the precedence constraints among the tasks. It ensures that no successor of a task is assigned to an earlier station than that task. Constraint (6) ensures that the sum of task times assigned to each station does not exceed the cycle time. The maximum cycle time being considered here is stated as reference cycle time, RefCT which is expressed as:

$$Ref_{CT} = \frac{\sum \text{shift task time}, T_e}{\text{no. of workstation}, S}, \quad s = 1, \dots, S \quad (7)$$

Since this paper involve multi-objective optimization, the weighted sum approach is used as follows:

$$\sum_{i=1}^M w_i f_i(x) \quad ; \quad w_1 f_1(x) + w_2 f_2(x) + \dots + w_n f_n(x) \quad (8)$$

In this case, the objective functions in equation 1, 2 and 3 needs to be normalized to ensure a consistent scaling to each of objective function. This is conducted by dividing the fitness value with the maximum value for each objective function. Thus, for future work the solution methods for solving this multi-objective problems are more simple and used a direct computation [19]. Equation (9) below represent the normalized fitness functions after using weighted sum approach.

$$F(X) = w_1 f'_1(x) + w_2 f'_2(x) + w_3 f'_3(x) \quad (9)$$

3. Numerical example

The generation of a joint precedence graph based on a combination for all three models are demonstrated in Figure 2 whereas Table 1 depicts the precedence matrix for this example. All known solution procedures for MMALB problem rely on the joint precedence graph which is, thus, indispensable for solving such problems. In short, this joint precedence diagram is like a blueprint on how to assemble the unit.

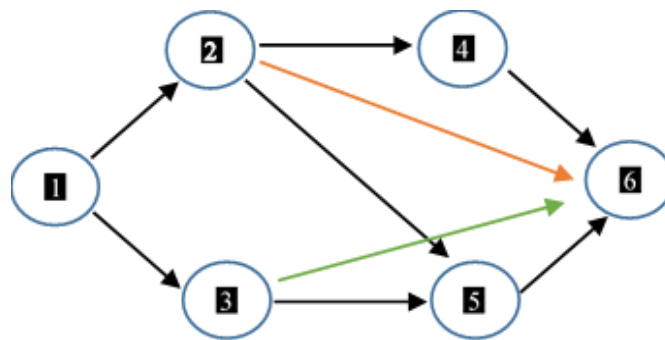


Figure 2. Joint precedence diagram

The example problem contain six tasks and three different model overall as referring to Table 1 and Table 2. Both shows the precedence table and model usage, U_{ej} respectively. Model is represented by model 1, 2 and 3 and the demand for each model is 5, 3, and 2 respectively. For the model usage, if any of the task e is used on certain model, it is marked with 1, and 0 if otherwise.

Table 1. Task, e , task time, te , predecessor element, p

e	te	p
1	25	-
2	17	1
3	8	1
4	40	2
5	11	2,3
6	33	4,5

Table 2. Task, e , task time, te , model usage, U_{ej}

e	te	U_{ej}		
		1	2	3
1	25	1	1	1
2	17	1	1	1
3	8	1	1	1
4	40	1	0	1
5	11	1	1	1
6	33	1	1	1

The modelling here also considers a resource that was used on the assembly lines. In this particular case, there will be three types of resources, R_T included tools, machines, and jigs. Table 3 summarizes the resources usage.

Table 3. Resources data tabulation

e	te	Resources, R_T				
1	25	T1	T2	J1	T3	-
2	17	T1	M2	J2	-	-
3	8	J1	-	-	-	-
4	40	T2	M1	T1	-	-
5	11	T3	J3	-	-	-
6	33	T3	M1	J2	M3	-

T1: Tool 1, **T2:** Tool 2, **T3:** Tool 3, **M1:** Machine 1, **M2:** Machine 2, **M3:** Machine 3, **J1:** Jig 1, **J2:** Jig 2, **J3:** Jig 3

Referring to mathematical model coded into MATLAB, all the selected sequence subjected to Ref_{CT} of 956.6667 minutes. Mean that, the specific tasks time that will be grouped into specific workstation must not exceed Ref_{CT} and only the last workstation can discard that rule since we will have the remaining task time to be converge at the last workstation which in this case workstation 3 (noted that maximum number of workstation is predetermined with total of three workstation).

In order to determine the performance for a particular assembly sequence, an evaluation needs to be conducted by comparing the objective functions. For assembly sequence [1 2 4 3 5 6], the procedure to

calculate objective functions are presented in the following sections. For this particular assembly sequence, the computational test from MATLAB give a result of 1300 minutes for cycle time, 15 resources, and 2.9 rate of product variation. Based on the demand of 5, 3, and 2 for model 1, 2, and 3 respectively, the mathematical model construct here can be calculated manually in order to verify the results for each objective function with the result obtained from computational model.

3.1. Cycle time calculation

For the sequence of task stated above, the task time for each task is tabulated in Table 4 below and the task distribution for each workstation is described together.

Table 4. Task time for each sequence

Sequence of task	1	2	4	3	5	6
Model task time(minute)	625	425	520	200	275	825

- 625 }WS 1
- $425 + 520 = 945$ }WS 2
- $200 + 275 + 825 = 1300$ }WS 3

Based on the manual calculation above, the maximum cycle time for the selected sequence is 1300 minutes and we can assign the task to its own workstation subjected to *Ref_{CT}*. Task 1 is assigned to workstation 1, task 2 & 4 to workstation 2, and task 3, 5, & 6 to workstation 3 as shown in the Table 5 below.

Table 5. Workstation distribution

Sequence of task	1	2	4	3	5	6
Model task time(min)	625	425	520	200	275	825
Workstation time(min)	625	945			1300	
Workstation, <i>s</i>	1	2			3	

3.2. Resources used on the line

Based on the assembly tasks assignment in section 3.1, the second objective function can be calculated. The manual calculation for the number of resources is illustrated in Table 6. The number of resources is determined by the different resource type in a workstation. For example, in workstation 2, since tasks 2 and 4 used similar T1 resource, the total resources in this workstation is equivalent to 5.

Table 6. Resources used by each workstation

Workstation	1		2		3	
Sequence of task	1	2	4	3	5	6
Resources type	T1	T1	T1	J1	T3	T3
	T2	M2	T2	-	J3	M1
	T3	J2	M1	-	-	M2
	J1	-	-	-	-	J2
Y_{rs}	$\Sigma = 4$		$\Sigma = 5$		$\Sigma = 6$	

Therefore, by using the objective function definition for resource used on assembly line given in equation (2), the summation of resources used by each workstation are as followed:

$$\begin{aligned}
 f2 &= \min \sum_{s=1}^S \sum_{r=1}^{max_R} Y_{rs} \\
 &= 4 + 5 + 6 \\
 &= 15 \text{ resources}
 \end{aligned}$$

3.3. Product rate variation (PRV)

Product rate variation exist due to the nature of MMALBP itself which capable to assemble more than one product at a time and in this paper three different model being assemble on the same assembly line. The objectives of PRV problem is to achieve a stable production rate for each product. For instance, the target is to assemble 10 unit of particular product denoted by k , and we have a sequence of product, j .

Sequence of product, j	1	2	3	1	2	1	1	3	2	1
--------------------------	---	---	---	---	---	---	---	---	---	---

Based on the definition of objective function in (3), the value for each parameter have been calculated and summarized in Fig. 3.

X_{jk}	1	0	0	1	1	0	1	1	1	2	1	1	2	2	1	3	2	1	4	2	1	4	2	2	4	3	2	5	3	2
k	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7	8	8	8	9	9	9	10	10	10
d_i	5	3	2	5	3	2	5	3	2	5	3	2	5	3	2	5	3	2	5	3	2	5	3	2	5	3	2	5	3	2
$\left(x_{jk} - k \times \frac{d_j}{D_T}\right)^2$	0.25	0.09	0.04	0	0.16	0.16	0.25	0.01	0.16	0	0.04	0.04	0.25	0.25	0	0	0.04	0.04	0.25	0.01	0.16	0	0.16	0.16	0.25	0.09	0.04	0	0	0

Figure 3. Product rate variation summarization

Hence, the summation of $\left(x_{j,k} - k \times \frac{d_j}{D_T}\right)^2$ in the table is the PRV value that needed which resulted to 2.9 for the selected sequence of product, j .

4. Conclusion and future work

This paper proposed a mathematical model on MMALBP which consider minimize cycle time, minimize resources used on assembly line and minimize product rate variation (PRV) as the objectives. The computational model was developed using MATLAB software. In order to verify the output from the computational model, manual calculations had been carried out to compare the output.

From the output of both methods, we can conclude that the objective function in term of its mathematical expression is valid to be used since the acquired results for each objective function is the same either for computerized method or manual computation. Future work will consist of an optimization procedure for the proposed models earlier by using an artificial intelligence approach such as genetic algorithm, particle swarm optimization and simulated annealing.

Acknowledgement

The research described in this paper was performed under the research grant RDU140103 from Ministry of Higher Education Malaysia.

References

- [1] A. K. Elia, "Minimizing Number of Stations and Cycle Time for Mixed Model Assembly Line," vol. 3, no. 8, pp. 1359–1364, 2014.
- [2] C. Becker and A. Scholl, "A survey on problems and methods in generalized assembly line balancing," *Eur. J. Oper. Res.*, vol. 168, no. 3, pp. 694–715, 2006.
- [3] A. Scholl and C. Becker, "State-of-the-art exact and heuristic solution procedures for simple assembly line balancing," *Eur. J. Oper. Res.*, vol. 168, no. 3, pp. 666–693, Feb. 2006.
- [4] N. Boysen, M. Fließner, and A. Scholl, "A classification of assembly line balancing problems," *Eur. J. Oper. Res.*, vol. 183, no. 2, pp. 674–693, Dec. 2007.
- [5] O. Battaglia and A. Dolgui, "Reduction approaches for a generalized line balancing problem," *Comput. Oper. Res.*, vol. 39, no. 10, pp. 2337–2345, 2012.
- [6] N. T. Thomopoulos, "Mixed Model Line Balancing with Smoothed Station Assignments," *Manage. Sci.*, vol. 16, no. 9, pp. 593–603, 1970.
- [7] H. Gökçen and E. Erel, "Binary Integer Formulation for Mixed-Model Assembly Line Balancing Problem," *Comput. Ind. Eng.*, vol. 34, no. 2, pp. 451–461, Apr. 1998.
- [8] Z. D. Zhang and H. Sharifi, "Towards theory building in agile manufacturing strategy - A taxonomical approach," *IEEE Trans. Eng. Manag.*, vol. 54, no. 2, pp. 351–370, 2007.
- [9] I. Kucukkoc and D. Z. Zhang, "Simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines," *Int. J. Prod. Res.*, vol. 52, no. 12, pp. 3665–3687, 2014.
- [10] Ş. Akpinar and A. Baykasoğlu, "Modeling and solving mixed-model assembly line balancing problem with setups. Part I: A mixed integer linear programming model," *J. Manuf. Syst.*, vol. 33, no. 1, pp. 177–187, Jan. 2014.
- [11] B. Yagmahan, "Mixed-model assembly line balancing using a multi-objective ant colony optimization approach," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 12453–12461, Sep. 2011.
- [12] L. Tiacchi, "Simultaneous balancing and buffer allocation decisions for the design of mixed-model assembly lines with parallel workstations and stochastic task times," *Int. J. Prod. Econ.*, vol. 162, pp. 201–215, Apr. 2015.
- [13] I. Kucukkoc and D. Z. Zhang, "Mathematical model and agent based solution approach for the simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines," *Int. J. Prod. Econ.*, vol. 158, pp. 314–333, 2014.
- [14] P. M. Vilarinho and A. S. Simaria, "ANTBAL: An ant colony optimization algorithm for balancing mixed-model assembly lines with parallel workstations," *Int. J. Prod. Res.*, vol. 44, no. 2, pp. 291–303, 2006.
- [15] A. Hamzadayi and G. Yildiz, "A simulated annealing algorithm based approach for balancing and sequencing of mixed-model U-lines," *Comput. Ind. Eng.*, vol. 66, no. 4, pp. 1070–1084,

- Dec. 2013.
- [16] N. T. Thomopoulos, *Assembly Line Planning and Control*. Springer International Publishing, 2013.
 - [17] N. Boysen, M. Fliedner, and A. Scholl, “The product rate variation problem and its relevance in real world mixed-model assembly lines,” *Eur. J. Oper. Res.*, vol. 197, no. 2, pp. 818–824, Sep. 2009.
 - [18] Ş. Akpınar and A. Baykasoğlu, “Modeling and solving mixed-model assembly line balancing problem with setups. Part II: A multiple colony hybrid bees algorithm,” *J. Manuf. Syst.*, vol. 33, no. 4, pp. 445–461, Oct. 2014.
 - [19] P. Chutima and P. Chimklai, “Multi-objective two-sided mixed-model assembly line balancing using particle swarm optimisation with negative knowledge,” *Comput. Ind. Eng.*, vol. 62, no. 1, pp. 39–55, Feb. 2012.