A Modified Heuristic Algorithm for the Mixed Model Assembly Line Balancing

Sungyoul Lee*

Abstract This paper proposes a modified heuristic mixed model assembly line (MMAL) balancing algorithm that provides consistent station assignments on a model by model basis as well as on a station by station. Basically, some of single model line balancing techniques are modified and incorporated to be fit into the MMAL. The proposed algorithm is based on N.T. Thomopoulos' [8] method and supplemented with several well proven single model line balancing techniques proposed in the literature until recently. Hoffman's precedence matrix [2] is used to indicate the ordering relations among tasks. Arcus' Rule IX [1] is applied to generate rapidly a fairly large number of feasible solutions. Consequently, this proposed algorithm reduces the fluctuations in operation times among the models as well as the stations and the balance delays. A numerical example shows that the proposed algorithm can provide a good feasible solution in a relatively short time and generate relatively better solutions comparing to other three existing methods.

Key Words : Mixed model assembly line balancing, Precedence matrix, Arcus' Rule IX, Modified Heuristic algorithm

1. Introduction

With the speedy and various demands of customers, today's manufacturers have now rather a MMAL than a single one. MMAL is known to be a special case of production lines where various and different models of the same product are inter-mixed to be assembled simultaneously on the same line [3]. The produced items keep changing from model to model continuously on the line. In order to reduce inventory cost, the number of models on the line is usually kept at a minimum level considering both customers' satisfaction for different varieties and the corresponding demands.

In a MMAL balancing, work elements are assigned to operators on a daily or shift basis rather than on a cycle time basis as is often done in single model lines. Here, general practice is to distribute evenly the total daily workload to stations. However, station by station assignments on individual models are not considered. This may lead to an uneven work flow along the line for a particular model.

Therefore, the objective of this study is to propose a Heuristic MMAL balancing algorithm that leads to more consistent station assignments on a model by model basis as well as a station by station.

Several single model line balancing techniques are modified and incorporated to be suitable to the MMAL case. The applied techniques and proposed balancing procedure are explained in the following sections. Finally, a case problem already used in other reference is used here again for the purpose

* Kwandong University, Department of Computer Science & Engineering

of testing the validity and comparing the performance of the proposed algorithm with others.

2. Mixed Model Assembly Line Balancing Algorithm

In order to induce a more efficient and simplified approach while in keeping the ultimate goal to balance work station times, several suitable methods among single line balancing techniques have been incorporated to the basis of N.T. Thomopoulos' MMAL balancing method [8]. Hoffman's precedence matrix [2] is used to indicate the ordering relations among tasks. Arcus' *Rule IX* [1] is applied to generate rapidly a fairly large number of feasible solutions.

2.1 Applied Methods

2.1.1 Determination of the lower and upper limits of station times

In order to induce even assignments of operation times through the all work stations, it should be avoided for a certain work station to occur too much idle time. A general balancing approach is to find even work station assignments with a minimum number of stations. It is to simulate some assigning process with various cycle times until even assignments with minimum idle times have been reached.

In a MMAL, since operation assignments are done based on the shift time (or daily production time) rather than a cycle time, a determination of feasible lower and upper limits of station times which can be applied equally through the all stations will be desirable in terms of efficiency and simplicity. Here, the upper limit (T_H) of the station times equals to the shift time itself. After obtaining a mean station time (T_a), the lower limit (T_L) can be calculated as T_a minus the difference between T_H and T_a.

$$T_a = \frac{\sum_{k=1}^{K}\sum_{j=1}^{J} N_j t_{jk}}{n} = \frac{\sum_{k=1}^{K}\widehat{t_k}}{n} \quad \cdots\cdots (1)$$

Where, n: the number of work stations.

J: the number of models

K: total number of work elements

N_j: the minimum demand ratios of the model *j*.

T_i: total times assigned on the i th work station for a shift time.

t_{jk} : $k\,th$ $element$ $time$ of the $j\,th$ model.

$\widehat{t_k} = total$ $time$ of $k\,th$ $element$.

$$T_L = T_a - (T_H - T_a) = 2T_a - T_H \quad \cdots\cdots (2)$$

$$T_L \le T_i \le T_H \quad (i = 1, 2, \ldots, n) \quad \cdots\cdots (3)$$

2.1.2 Determination of a work element eligible to assign

If more than two work elements are eligible to be assigned to the station, an element is efficiently selected according to the sampling method based on Arcus' *Rule IX*. Here, the weighting procedure of Arcus' *Rule IX* is a product of the weights calculated by the following five Rules: i.e., *Rule IX* = *Rule I* x *Rule V* x *Rule VI* x *Rule VII* x *Rule VIII*. Each rule is explained as follows:

Rule I: Weight tasks that fit in proportion to the work element time.

Rule V: 1 / (total number of unassigned tasks - 1 - number of all the tasks that follow the task being considered)

Rule VI: total number of all following tasks + 1

Rule VII: Weight tasks that fit by the sum of the task and of all following task times.

Rule VIII: (total number of following tasks + 1) / (number of levels which those following tasks occupy + 1)

The way of using the *Rule IX* follows:

First, assign a calculated weight as above to each of eligible work elements. Secondly, assign selection probabilities to the work elements in proportion to the weights. Finally, select a work element according to the Monte Carlo method based on the generated random number.

2.2 Balancing Procedure

Terminology used here is explained as follows:

(1) Precedence matrix: a matrix that represents precedence relationship among work elements. Precedence matrix Y is given below according to Hoffman's method [2].

$$Y = \begin{cases} 1: \text{If element of row i immediately precedes the element of column j} \\ 0: \text{Otherwise} \end{cases}$$

(2) KODE ROW: a row that sums each column of the precedence matrix.

(3) Standard KODE ROW: a row that sums each column of the initial precedence matrix.

$$SKD(j) = \sum_{i=1}^{k} D(i,j) \quad (j = 1, 2, \ldots, k)$$

Where, D(i,j) : precedence relations matrix for a given data

(4) Starting KODE ROW *i*: a KODE ROW that uses to search the first work element to be assigned in work station i.

(5) Eligible work elements to assign: unassigned work elements that are eligible to be possibly assigned to the current work station. They should obey the precedence relationship constraints as well as the remaining station time limits. Here, these are the work elements that have their element times less and equal to the remaining station times among the work elements associated with the columns in which its KODE ROW is zero.

(6) Iteration limits: a desirable number of generations of feasible solutions. Normally, 1000 is appropriate.

(7) Modifier (Δi): Model balancing factor.

$$\triangle i = \sum_{j=1}^{J} |P_j - P_{ij}| \qquad (i = 1, 2, \ldots, n)$$

$$P_j = \frac{N_j}{n} \left(\sum_{k=1}^{K} t_{jk} \right) \qquad (j = 1, 2, \ldots, J)$$

Where, P_j : a mean work times per work station on all units of j th model for a shift time. P_{ij} : sum of work times on all units of j th model in i th work station for a shift time.

Among the generated feasible solutions, it is considered as optimum solution for the solution to have a minimum Δi.

{Balancing Procedure}

Step 1: Construct a combined precedence matrix by inputting given data (number of models, number of stations, shift time, demands, work element times, and precedence relations) and then calculate a standard KODE ROW.

Step 2: Calculate a mean station time (T_a) based on the given number of stations and then calculate desirable lower and upper limits of the station times. Assign the station number as 1 and iteration number as 0.

Step 3: Find an eligible work element to be assigned. Among the work elements where its KODE ROW is zero, find eligible elements that their sum of element times, T(i) is less and equal to the remaining station times, TR. Assign weights to the selected elements according to the Arcus' *Rule IX* and assign the corresponding selection probabilities to the elements based on the weights. Here, if there is no eligible element, then go to *Step 6*.

However, a starting KODE ROW of the first station always becomes a standard KODE ROW and when selecting the first element to be assigned in each station, the remaining station time is T_H.

Step 4: Select a work element from the eligible work elements list by means of Monte Carlo method and then increase the iteration number by 1. Replace the STIME as sum of T(i)s of the selected work elements and recalculate the new remaining TR by the remaining TR minus T(i) of the selected work element.

Step 5: If a STIME is less than T_L, then assign a random big number M (99999) to the diagonal element, D(i,i) of the assigned work element i. Then update a KODE ROW as a new row which is a KODE ROW minus the row on the corresponding precedence matrix of the assigned work element. If the iteration number is greater and equal to the IMAX, go to *Step 9* and otherwise, go to *Step 3*.

Step 6: Replace the STIME of the corresponding work station as a cumulative sum of work element times, $\Sigma T(i)$ up to now and replace the KODE ROW as a starting KODE ROW.

Step 7: Calculate the Δi. If it is the first one, place this as an optimum Δi and otherwise, compare this with current optimum Δi and if this is greater than the current optimum, then go to *Step 3*. Here, if a final work station is reached, then go to *Step 11*.

Step 8: Replace the optimum Δi as the current Δi and print out the current solutions. Here, if the Δi is not zero, then go to *Step 3*.

Step 9: Print the final results considering the optimum solutions obtained up to now as final solutions. Increase the station number by 1 and replace the iteration number as 1.

Step 10: Select all the remaining work elements unassigned until now and then go to *Step 6*.

Step 11: Calculate the balance delay considering the optimum solutions obtained up to now as final solutions.

Step 12: Print the results obtained up to now and then stop.

3. Case Problem and Comparison Analysis

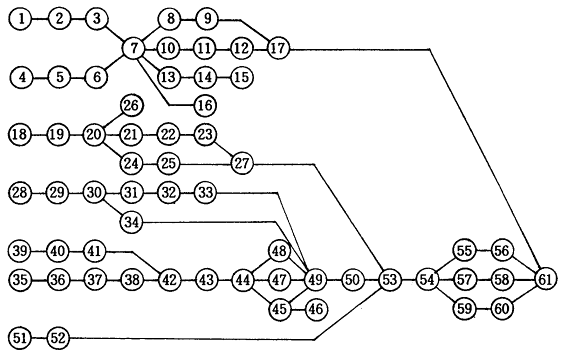
For the purpose of testing validity of the proposed system as well as comparing with other methods, a same case problem which was already used in Reference [5] has been used again here.
The problem is as follows:

There is a MMAL where three models are assembled. The total number of work elements is 61. Work element times for the three models and their combined precedence relationship diagram are shown on <Table 1> and (Figure 1) respectively. Seven or eight work stations are desirable and the minimum production ratios for the three models are 1:2:3. A cycle time for each model in each work station is given as 13.8.

The Results obtained from the proposed balancing procedure are shown on <Table 2> with the ones of the other three methods for the

<Table 1> Work Element Times for the Three Models

i	ti1	ti2	ti3	i	ti1	ti2	ti3	i	ti1	ti2	ti3
1	2.2	3.1	0.9	22	1.2	0.5	1.6	43	1.4	0.8	3.2
2	3.3	2.3	3.0	23	2.5	0.0	3.0	44	0.4	2.0	0.0
3	2.3	3.2	3.0	24	0.0	2.6	1.6	45	0.0	2.6	0.7
4	2.7	0.0	2.7	25	0.7	0.0	0.0	46	2.6	0.0	1.8
5	3.2	0.8	0.8	26	1.4	1.9	3.4	47	0.7	2.1	1.2
6	1.4	1.0	2.0	27	1.8	0.9	0.0	48	1.1	0.9	3.2
7	0.0	1.3	2.3	28	0.0	0.7	2.7	49	0.0	0.8	0.0
8	1.8	0.0	1.9	29	0.0	1.5	0.5	50	0.7	0.5	1.5
9	3.0	1.2	1.4	30	1.5	0.0	0.6	51	2.1	2.9	0.3
10	1.2	2.3	2.8	31	1.5	0.9	0.0	52	0.0	0.0	0.9
11	2.1	1.3	1.1	32	0.0	1.1	2.9	53	1.5	0.0	3.1
12	0.0	2.0	0.8	33	1.1	0.0	3.0	54	0.5	0.9	0.7
13	1.2	0.0	2.2	34	2.5	1.7	0.6	55	1.6	1.5	0.0
14	0.8	0.0	2.5	35	0.9	0.0	1.5	56	2.6	0.0	1.8
15	1.2	0.0	2.0	36	1.0	2.5	0.7	57	0.8	2.1	2.9
16	0.0	1.8	1.6	37	2.6	1.5	1.3	58	2.2	0.7	0.0
17	1.3	3.1	1.4	38	1.8	0.4	0.0	59	1.5	1.3	3.2
18	2.5	1.6	3.3	39	1.8	0.5	0.0	60	3.3	2.9	0.9
19	2.5	1.8	1.2	40	1.1	1.2	1.2	61	0.0	2.6	0.7
20	0.8	0.6	1.8	41	1.4	2.4	3.3				
21	0.4	1.6	1.2	42	0.0	0.4	0.6	total	81.5	74.3	94.5



(Figure 1) Combined Precedence Relationship Diagram for the 61 Work Elements

<Table 2> Work Assignments obtained from the 4 Different Methods.

Heuristic Methods	Work Station Number	Assigned Work Elements
LCR	1	39 40 51 28 29 30 31 4
	2	5 32 34 1 2 3 33 6 18
	3	19 20 24 25 26 7 10
	4	16 13 11 12 35 36 14
	5	15 37 38 41 42 43 44 48 21
	6	22 23 27 47 45 46 8 9 17 52 49
	7	50 53 54 59 60 57 55 58
	8	56 61
RPWT	1	18 19 39 35 20 36 40 4
	2	37 38 41 28 42 29 43 44
	3	30 21 22 24 31 1 23
	4	45 32 2 5 25 47 33 51 6
	5	48 34 3 27 49 50 7 10 13
	6	52 53 54 59 11 57 55 58
	7	60 8 12 56 9 14 17 16 61
	8	26 46 15
Kim & Kwak	1	4 5 6 28 18 19 20 1 21 26
	2	39 24 40 41 29 22
	3	23 35 51 52 36 37 30 38
	4	42 25 27 31 32 33 2
	5	34 3 43 44 47 48 45 49 50
	6	53 54 55 57 58 59 60 7 10
	7	16 11 13 14 8 9 15 12 17
	8	46 56 61
Proposed Method	1	35 1 51 2 18 3 19 4 20
	2	24 26 5 28 25 21 22
	3	23 6 39 7 10 16 8 11
	4	12 13 14 27 9 17 15 36 37 38
	5	40 41 29 42 43 30 44 31
	6	45 34 48 47 52 3 33 46 49 50
	7	53 54 55 56 57 58 59 60 61

comparison purpose. Here, it is noted that only the proposed system assigned work elements to 7 stations rather than 8 stations used in others.

<Table 3> shows variances of station times on each model and total sum of the variances times the square of production ratios as a measure of even assignments in station times on each model.

For the comparison purpose, the results from the other three methods already obtained by Kim & Kwak are shown on the same table [5]. Here, LCR stands for Largest Candidate Rule and RPWT for the Ranked Positional Weight Technique. For more details about LCR and RPWT, refer to the reference [5]. In the other three methods, it should be noted that the variances of the station times per model and the station times per cycle time were obtained without including the last station times. However, the proposed method included the last station times as well.

From the <Table 3>, it is notable that proposed system could generate much better results compare to other three methods in terms of very low variances in station times for each model. Based on the variance values of the last two rows in each method, the proposed method shows the much smaller values which mean evenly assignments on both the station by station and the model by model basis in balancing MMALs.

In other words, we can see that proposed system could provide even work assignments in station times per model as well as in total station times.

4. Conclusion

In this study, an efficient heuristic MMAL balancing algorithm was proposed. The developed system could generate a good feasible solution in a short time which leads to even station by station assignments on individual models as well. A numerical example shows that the proposed system could generate better or as good feasible solutions as other known methods until now.

<Table 3> Station Times and Variances per Model of the 4 Methods

Heuristic Methods	Work Station Number	Model			T_k
		1	2	3	
LCR	1	13.8	11.1	13.2	75.6
	2	12	9.7	13.7	72.5
	3	12	9.2	13.7	71.5
	4	9.1	4.3	13.8	59.1
	5	6.1	13.7	13.7	74.9
	6	13.5	13.8	11.6	75.9
	7	12.1	9.9	12.3	68.8
	8	2.6	2.6	2.5	15.3
	σ_j^2	5.936	8.880	0.637	29.870
	$\sum R_j^2\sigma_j^2$	5.936+8.880x4+0.637x9=47.189			
RPWT	1	12	7.5	13.5	67.5
	2	6.1	11.2	13.4	68.7
	3	13.2	13	13.3	79.1
	4	13.1	13.7	13.2	80.1
	5	8.4	9.1	13.8	68
	6	8.1	7.6	13.7	64.4
	7	13.8	8.9	10.3	62.5
	8	6.8	3.3	3.3	23.3
	σ_j^2	8.073	5.431	1.279	40.622
	$\sum R_j^2\sigma_j^2$	8.073+5.431x4+1.279x9=41.308			
Kim & Kwak	1	12.6	9.9	13.2	72
	2	12.1	10.3	12.1	69
	3	11.7	11.3	13.1	73.6
	4	10.1	8.2	13.6	67.3
	5	5.5	12.3	13.6	70.9
	6	12.9	11	13.5	75.4
	7	11.4	8.7	11.1	62.1
	8	5.2	2.6	4.3	23.3
	σ_j^2	5.574	1.800	0.764	16.802
	$\sum R_j^2\sigma_j^2$	5.574+1.800x4+0.764x9=19.650			
Proposed Method	1	11.6	10.10	13.5	72.3
	2	11.6	10.7	13.7	74.1
	3	11.7	10.5	13.7	73.8
	4	12.7	10.5	13.4	73.9
	5	11.5	10.3	13.5	72.6
	6	8.4	10.2	13.4	69
	7	14	12	13.3	77.9
	σ_j^2	2.460	0.356	0.020	6.051
	$\sum R_j^2\sigma_j^2$	2.460+0.356x4+0.020x9=4.064			

References

[1] Elwood, S.B., "COMSOAL: A Computer Method of Sequencing Operations for Assembly Lines", *Readings in Production and Operations Management*, New York, John Wiley and Sons, Inc., pp. 336-349, 1966.

[2] Hoffmann, T.R., "Assembly Line Balancing with a Precedence Matrix", *Management Science*, Vol. 9, No. 4, pp. 551-562, 1963.

[3] Kim, Siwon and Jeong, Bongju, "Product sequencing problem in Mixed-Model Assembly Line to minimize unfinished works", *Computers and Industrial Engineering*, Vol. 53, pp. 206-214, 2007.

[4] Kim, Y.G. and Hyun, C.J., "Genetic Algorithms for Mixed Model Assembly Line Sequencing", *Journal of KIIE*, Vol. 20, No. 3, pp. 15-34, 1994.

[5] Kim, Y.G. and Kwak, J.S., "Mixed Model Assembly Line Balancing with the Related Task Consideration", *Journal of KORMS*, Vol. 18, No. 2, pp. 1-22, 1993.

[6] Kim, Y.G., Lee, S.Y., and Kim, Y.J., "A Genetic Algorithm for Improving the Workload Smoothness in Mixed Model Assembly Lines", *Journal of KIIE*, Vol. 23, No. 3, pp. 515-532, 1997.

[7] Boysen, Nils, Fliedner, Malte, and Scholl, Armin, "Sequencing mixed-model assembly lines: Survey, classification and model critique", *European Journal of Operational Research*, Vol. 192, pp. 349-373, 2009.

[8] Thomopoulos, N.T., "Mixed Model Line Balancing with Smoothed Station Assignments", *Management Science*, Vol. 16, No. 9, pp. 593-603, 1970.

이 성 렬 (Sungyoul Lee)

• 종신회원
• 인하대학교 학사 (기계공학)
• 택사스주립대학 - 알링톤 석사 (산업공학)
• 노스다코타주립대학 박사 (산업공학)
• 현재 : 관동대학교 공과대학 컴퓨터학과 교수
• 연구분야 : 데이터마이닝, 메타휴리스틱 응용 등

논 문 접 수 일 : 2010년 07월 30일
1차수정완료일 : 2010년 09월 07일
게 재 확 정 일 : 2010년 09월 10일