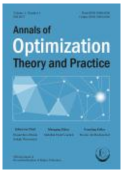
Annals of Optimization Theory and Practice

Volume 1, Number 1, 11-21
February 2018

DOI: 10.22121/aotp.2018.109416.1005



A cost based mathematical formulation for U-type assembly line balancing problem

Morteza Khorram

Department of Industrial Engineering, Shiraz University of Technology, Shiraz, Fars, Iran

✉ m.khorram@sutech.ac.ir

(Received: October 22, 2017 / Accepted: November 30, 2017)

Abstract This paper focuses on formulating a typical U-type assembly line balancing problem. A cost based objective function including equipment cost, worker time related cost, and station opening cost is introduced to be minimized in existence of a constant cycle time. Finally, efficiency of the proposed formulation of the introduced problem is studied and tested over some benchmarks.

Keywords U-type assembly line balancing problem; Mathematical modeling; Cost based objective function

---

1. Introduction

In today's market of industrial products, for a manufacture it is important to be timely from the customers' point of view. Therefore, mass production could be useful to produce the products with such speed to satisfy the customers. For this aim, assembly (production) lines could be used for being able to produce the products in a faster way. On an assembly line, there is a sequence of stations which each of them perform one or more production activities. The raw parts of the products are launched down the line and flow over the stations to be completed as final product after passing the last station. As there are several activities to be performed for obtaining a complete product, the activities are assigned to the stations of the assembly line according to two rules: (1) the precedence relationships of the activities must be respected when those are assigned to the stations too, (2) when assigning the activities to the stations, the sum of processing time of the activities of each station should not be more than an allowed time say cycle time of the line. The cycle time is an upper limit for the processing time of each station which the demand of the line is fulfilled if the line respects to it. Assembly lines based on the physical restrictions of the space may be designed in many formats e.g. straight assembly lines, U-type assembly lines, etc. Considering any of the physical forms of the assembly lines, the line can process to produce a single type of product or more than one type of the products which can name single and mixed model assembly lines

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran

respectively. The problem of how to assign the activities to optimize the criteria such as number of active stations, cycle time of the line, etc. is named assembly line balancing problem (ALBP). In a simple classification this problem is classified into two types of (1) simple assembly line balancing problem (SALBP), and (2) U-type assembly line balancing problem (UALBP).

Generally in the solutions of SALBP the stations are arranged on a straight line. The work pieces of product enter from beginning of the line (first station) and go over the stations in order and leave the line from another side (last station) as complete product. Therefore, an activity can be assigned to a station if and only if its predecessors are assigned to the same or earlier stations. Each station is assigned to at least one worker and a worker works on the work pieces which their predecessors already have been performed. On the other hand, in the problem of assembly line which is arranged based on U-type form, stations can be arranged in order to during the same cycle the activities of two work pieces at different positions on the line can be performed by a worker. Therefore, the difference of UALBP to SALBP is that a station *k* can contain not only activities which their predecessors are assigned to one of the stations 1, ..., *k*, but also activities that their predecessors will be performed until the product comes back to the station *k* for the second time. It means that a station may be extended to different parts of the line. A station which handles the same work piece in two different parts of the line is called a crossover station. Such station in a U-type line contain some activities on one leg and some on another leg of the line. In conclusion, in UALBP, an activity can be assigned to a station if either its all predecessors have been assigned until that station or its all successors have been assigned until that station. Considering the precedence graph of Figure 1 as an example, two different simple and U-type arrangement of its assembly line is shown by Figure 2. In the U-type arrangement shown in Figure 2, the first station is extended to be on both legs of the U-type line. If we assume one worker for this station, the worker first performs the activities 3 and 5 for the raw work pieces, then he/she goes to another side to perform activity 10 for the work pieces which their all activities have been completed.

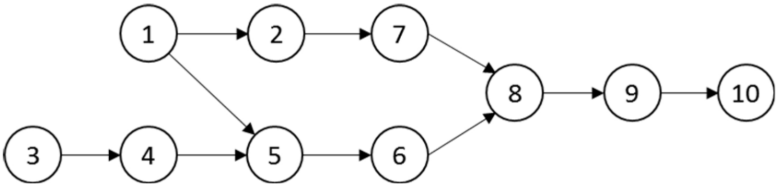


Figure 1. A precedence graph of activities as an example.

Literature of ALBPs is also of interest to be reviewed. The studies like Baybars (1986) and Becker and Scholl (2006) focused on definition, review, and classification of ALBPs. Most of typical assembly lines are detailed in such studies. As there are many limitations in ALBPs, many researches considering different assumptions have been performed. The studies of Chica *et al.* (2011) and Hamta *et al.* (2013) are interesting for the aim of cycle time minimization. On the other hand, the studies like Ponnambalam *et al.* (2000), Nourmohammadi and Zandieh (2011) and Chica *et al.* (2011) minimize the

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran

total number of stations and smoothness index in ALBPs. Meta-heuristics also was of interest to solve different ALBPs. Genetic algorithm based meta-heuristic approaches was applied by Ponnambalam *et al.* (2000), Zhang and Gen (2011), and Aydemir-Karadag and Turkbey (2013) to solve different ALBPs. In another studies simulated annealing and particle swarm optimization algorithms were used to solve ALBPs (see Baykasoglu, 2006; Hamta *et al.*, 2013).

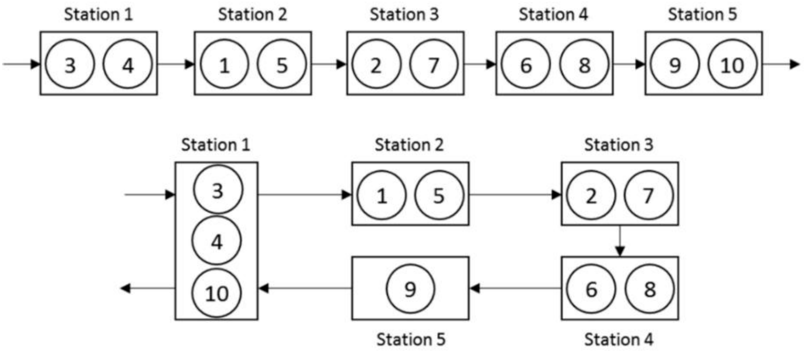


Figure 2. Feasible solutions of SALBP (above) and UALBP (below) for the precedence graph of Figure 1.

As U-type assembly line is an important type of assembly lines, this paper focuses to formulate a typical UALBP. UALBPs previously have been the topic of many studies such as Becker and Scholl (2006), Ogan and Azizoglu (2015), etc. This paper proposes a new formulation of UALBP which focuses to minimize total cost of balancing which consists of equipment (tool) cost, station opening cost and worker time-dependent cost. The mixed integer formulation of this typical cost-oriented UALBP is modeled in this study and some benchmarks are solved to prove the efficiency of the formulation.

The paper is organized by the following sections. Section 2 presents the mathematical formulation of the cost-oriented UALBP. The proposed formulation is validated by some small and large sized test problems in Section 3. The paper ends with conclusion in Section 4.

2. The cost-oriented formulation of U-type assembly line balancing problem

The cost-oriented U-type assembly line balancing problem is formulated in this section. To do such formulation, first, some assumption and then some notations should be introduced. Considering the initial concepts presented in the previous section, the following assumptions is considered for modelling the problem,

- The considered U-type line produces one type of product.
- The number of potential stations is given. So that, in the solution obtained by the model some of them may remain closed. Of course, for the closed stations no cost is charged.
- Cycle time of the line is given and fixed.

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran

- Precedence graph of the activities, activity processing time, and activity processing cost per time unit are given.
- Each station has only one worker.
- As each station may have more than one activity with different processing costs, the processing cost of the station per time unit is equal to its most expensive activity.
- Worker time-dependent cost for each station is obtained by multiplying processing cost of the station per time unit and the cycle time.

The notations used in the model of the problem is introduced by Table 1.

Based on the above-mentioned assumptions and notations introduced by Table 1, the mathematical formulation of the UALBP is detailed in the following subsections.

2.1. Precedence constraints

The most important priority of a feasible solution of any ALBP is respecting to the precedence relationships of the activities. In the literature of UALBP there are different version of precedence constraints to respect to such relationships. In this study the following constraints is used to model the precedence constraints of UALBP of this paper.

Table 1. Notations used in formulation of the problem.

Notation	Type	Definition
i, p, s	Index	Indexes used for activity
I	Parameter	Number of activities
k	Index	Index used for station
K	Parameter	Number of potential stations
CT	Parameter	Cycle time
t_i	Parameter	Processing time of activity *i*
e_i	Parameter	Processing cost per time unit for activity *i*
PR_i	Parameter	Predecessor set of activity *i*
SC_i	Parameter	Successor set of activity *i*
L_i	Parameter	Set of equipment required by task *i*
X_{ik}	Variable	1, if activity *i* is assigned to station *k*; 0, otherwise
W_k	Variable	1, if activity potential station *k* is opened; 0, otherwise
U_i	Variable	1, if activity *i* is assigned to the front-line of U-line; 0, otherwise
Z_{lk}	Variable	1, if equipment *l* is assigned to workstation *k*; 0, otherwise
C_k	Variable	Processing cost per time unit for station *k*

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran

$$\sum_{k=1}^{K} kX_{pk} - \sum_{k=1}^{K} kX_{ik} \le K\left(1+U_p-2U_i\right) \qquad \forall(p,i) \mid p \in PR_i \qquad (1)$$

$$\sum_{k=1}^{K} kX_{sk} - \sum_{k=1}^{K} kX_{ik} \le KU_i \qquad \forall(i,s) \mid s \in SC_i \qquad (2)$$

These constraints together guarantee that an activity can be assigned to a station if either its all predecessors have been assigned until that station or its all successors have been assigned until that station.

2.2. Assignment constraints

As an assignment rule in ALBPs, each activity must be assigned to one of the potential stations. This limitation is respected by the following constraint.

$$\sum_{k=1}^{K} X_{ik} = 1 \qquad \forall i \qquad (3)$$

2.3. Cycle time constraints

Workers of any assembly line must perform their activities in a given cycle time. Cycle time of a line is obtained from demand quantity and available working hours to respond that quantity. Therefore, the sum of processing times of activities assigned to each station cannot exceed the cycle time. So, the following constraint should be applied in model of the UALBP.

$$\sum_{i=1}^{N} t_i X_{ik} \le CT \qquad \forall k \qquad (4)$$

2.4. Other station related constraints

A station is opened if contains at least one task. As the value of W_k defines whether or not station k is opened, the following constraint is proposed for this aim.

$$W_k \ge X_{ik} \qquad \forall i,k \qquad (5)$$

On the other hand, as the processing cost per time unit for each station is determined by its most expensive activity, the following inequality is proposed for calculating value of C_k.

$$e_i X_{ik} \le C_k \qquad \forall i,k \qquad (6)$$

2.5. Equipment assignment constraints

Each activity needs a special set of equipment to be performed. If more than one activity which need a special type of tool are assigned to a station, just one of that tool will be enough for performing the activities. To consider this assumption, the following set of constraints is introduced,

$$X_{ik} \le \frac{\sum_{l \in L_i} Z_{lk}}{|L_i|} \qquad \forall i,k \qquad (7)$$

In the constraint, $|L_i|$ shows the number of equipment are needed for activity i. If activity i is assigned to station k then all Z_{lk} $(l \in L_i)$ values must be 1. On the other hand, if activity i is not assigned to station k then the Z_{lk} $(l \in L_i)$ values are free to take either 1 or 0 by help of minimization type objective function.

2.6. Cost-based objective function

As defined before the objective function of the model consists of three types of cost-based function which are modeled as follow,

- Minimizing the total equipment cost which is obtained by multiplying each tool by its purchasing cost for all tools which is calculated by term $\sum_{l=1}^{L}\sum_{k=1}^{K} EC_l Z_{lk}$.
- Minimizing the total opening cost of all stations as $CI\sum_{k=1}^{K} W_k$.
- Minimizing the total processing costs of all stations by term $CT\sum_{k=1}^{K} C_k$.

2.7. Overall formulation

Considering the above-mentioned constraints and objective functions, and also in existence of sign constraints, the cost-based formulation of UALBP as summarized as follow.

$$\min \left(\sum_{l=1}^{L}\sum_{k=1}^{K} EC_l Z_{lk} \right) + \left(CI \sum_{k=1}^{K} W_k \right) + \left(CT \sum_{k=1}^{K} C_k \right) \qquad (8)$$

subject to

$$\sum_{k=1}^{K} kX_{pk} - \sum_{k=1}^{K} kX_{ik} \le K\left(1 + U_p - 2U_i\right) \qquad \forall (p,i) \mid p \in PR_i \qquad (9)$$

$$\sum_{k=1}^{K} kX_{sk} - \sum_{k=1}^{K} kX_{ik} \le KU_i \qquad \forall (i,s) \mid s \in SC_i \qquad (10)$$

$$\sum_{k=1}^{K} X_{ik} = 1 \qquad \forall i \qquad (11)$$

$$\sum_{i=1}^{N} t_i X_{ik} \le CT \qquad \forall k \qquad (12)$$

$$W_k \ge X_{ik} \qquad \forall i,k \qquad (13)$$

$$e_i X_{ik} \le C_k \qquad \forall i,k \qquad (14)$$

$$X_{ik} \le \frac{\sum_{l \in L_i} Z_{lk}}{|L_i|} \qquad \forall i,k \qquad (15)$$

$$X_{ik}, Z_{lk}, Y_i, W_k, U_i \in \{0,1\} \qquad \forall i,k,l \qquad (16)$$

$$C_k \ge 0 \qquad \forall k \qquad (17)$$

3. Computational experiments

The cost-based UALBP which is formulated by the model (8)-(17), is experimentally studied over some benchmarks in this section. For this aim the model is coded in GAMS 24.1.2 to run on a computer with an Intel Core i7-4770, 3.4 GHz processor and 8.00 GB RAM. To study the performance of the model, five benchmark problem of literature of assembly line balancing is selected to be solved by the model. In the data of the classic version of the benchmarks the only given data are precedence diagram and activity processing times. Therefore, we need to modify the data and add the values of missing parameters of the model. The data of the benchmarks are reported by tables 2, 3, and 4.

The formulation (8)-(17) is solved for the benchmarks of Table 2. And the results are reported in Table 5. The CPU times reflect the complexity of the proposed model.

Table 2. Some parameters' value of the benchmarks.

Parameter	Benchmark 1	Benchmark 2	Benchmark 3	Benchmark 4	Benchmark 5
No. of activity	8	11	21	25	30
No. of station	8	11	21	25	30
No. of equipment	4	6	8	10	12
Station opening cost	10000	10000	10000	10000	10000
Cycle time	20	10	15	10	25
Equipment purchasing cost (left to right, top to down)	1500, 2000 3500, 2000	1500, 2000 3500, 2000 1000, 5000	1500, 2000 3500, 2000 1000, 5000 3500, 2000	1500, 2000 3500, 2000 1000, 5000 3500, 2000 1000, 5000	1500, 2000 3500, 2000 1000, 5000 3500, 2000 1000, 5000 3500, 2000

Table 3. Some of the data of benchmarks 1 and 2.

Activity	Benchmark 1				Benchmark 2			
(i)	t_i	e_i	PR_i	L_i	t_i	e_i	PR_i	L_i
1	11	5	-	1, 3	6	4	-	1, 3, 5
2	17	5	1	2, 4	2	5	1	2, 4, 6
3	9	2	2	1, 2, 4	5	5	1	1, 2, 4
4	5	4	2	2, 3, 4	7	5	1	2, 3, 4
5	8	1	3	3, 4	1	3	1	3, 4, 5, 6
6	12	3	3, 4	1	2	5	2	1, 6
7	10	1	5	2	3	4	3, 4, 5	2, 5
8	3	4	6	2	6	1	6	2
9					5	2	7	3, 4
10					5	4	8	1, 2, 5
11					4	2	10	3, 4, 6

4. Concluding remarks

A typical assembly line balancing problem was studied in this article. A cost based objective function was suited to a U-type assembly line balancing problem to

Table 4. Some of the data of benchmarks 3, 4, and 5.

Act. (*i*)	Benchmark 3				Benchmark 4				Benchmark 5			
	t_i	e_i	PR_i	L_i	t_i	e_i	PR_i	L_i	t_i	e_i	PR_i	L_i
1	4	4	-	1, 3, 5	4	1	-	1, 3, 5, 9	8	1	-	1, 3, 5, 9, 12
2	3	1	1	2, 4, 6, 7, 8	3	4	-	2, 4, 6, 7, 8, 10	7	4	-	2, 4, 6, 7, 8, 10, 11
3	9	1	1	1, 2, 4, 7, 8	9	4	2	1, 2, 4, 7, 8	19	4	-	1, 2, 4, 7, 8, 11
4	5	1	3	2, 3, 4, 7, 8	5	1	3	2, 3, 4, 7, 8	10	1	1	2, 3, 4, 7, 8, 11, 12
5	9	2	4	3, 4, 5, 6, 8	9	5	4	3, 4, 5, 6, 8, 9, 10	2	5	1	3, 4, 5, 6, 8, 9, 10, 12
6	4	5	5	1, 6	4	3	5	1, 6, 10	6	3	5	1, 6, 10
7	8	4	5	2, 5, 7	8	4	6	2, 5, 7, 9	14	4	6	2, 5, 7, 9, 11
8	7	4	6, 7	2, 7	7	5	4	2, 7	10	5	7	2, 7, 11
9	5	3	8	3, 4, 8	5	5	8	3, 4, 8	1	5	8	3, 4, 8, 12
10	1	1	9	1, 2, 5, 7	1	5	6, 9	1, 2, 5, 7, 9	4	5	-	1, 2, 5, 7, 9, 11
11	3	4	9	3, 4, 6, 8	3	5	7, 8	3, 4, 6, 8, 10	14	5	2	3, 4, 6, 8, 10, 12
12	1	3	9	1, 3, 5	1	1	7	1, 3, 5, 9	15	1	2	1, 3, 5, 9, 12
13	5	2	9	2, 4, 6, 7, 8	5	2	9, 11	2, 4, 6, 7, 8, 10	5	2	12	2, 4, 6, 7, 8, 10, 11
14	3	4	7	1, 2, 4, 7, 8	3	5	13	1, 2, 4, 7, 8	12	5	13	1, 2, 4, 7, 8, 11

Table 4. continued

Act. (i)	Benchmark 3				Benchmark 4				Benchmark 5			
	t_i	e_i	PR_i	L_i	t_i	e_i	PR_i	L_i	t_i	e_i	PR_i	L_i
15	5	2	10, 11, 12	2, 3, 4, 7, 8	5	5	12	2, 3, 4, 7, 8	9	5	14	2, 3, 4, 7, 8, 11, 12
16	3	1	15	3, 4, 5, 6, 8	3	3	14	3, 4, 5, 6, 8, 9, 10	10	3	3	3, 4, 5, 6, 8, 9, 10, 12
17	13	1	13, 16	1, 6	13	1	15	1, 6, 10	2	1	3	1, 6, 10
18	5	4	13, 15	2, 5, 7	5	2	16, 17	2, 5, 7, 9	10	2	17	2, 5, 7, 9, 11
19	2	3	14, 18	2, 7	2	1	14	2, 7	18	1	18	2, 7, 11
20	3	3	17	3, 4, 8	3	5	14	3, 4, 8	16	5	14, 16	3, 4, 8, 12
21	7	4	2, 4	1, 2, 5, 7	7	3	20	1, 2, 5, 7, 9	21	3	20	1, 2, 5, 7, 9, 11
22					5	5	15, 19, 21	2, 5, 7, 9	14	5	15, 21	2, 5, 7, 9, 11
23					3	3	17	2, 7	16	3	22	2, 7, 11
24					8	2	21	3, 4, 8	7	2	10, 20	3, 4, 8, 12
25					4	3	18, 20, 23	1, 2, 5, 7, 9	17	3	24	1, 2, 5, 7, 9, 11
26									9	13	9, 25	3, 4, 5, 6, 8, 9, 10, 12
27									25	5	23, 26	1, 6, 10
28									7	2	27	2, 5, 7, 9, 11
29									14	3	27	2, 7, 11

Table 4. continued

Act. (i)	Benchmark 3				Benchmark 4				Benchmark 5			
	t_i	e_i	PR_i	L_i	t_i	e_i	PR_i	L_i	t_i	e_i	PR_i	L_i
30									2	7	29	3, 4, 8, 12

Table 5. Objective function values and CPU times for the benchmarks obtained by model (8)-(17).

Benchmark	Optimal objective function value	CPU time (seconds)
Benchmark 1	69800	3
Benchmark 2	93220	8
Benchmark 3	176920	1650
Benchmark 4	223540	3265
Benchmark 5	364975	5130

decrease equipment cost, station opening cost, and worker time related cost simultaneously. Finally the efficiency of the proposed UALBP formulation was tested over some benchmark problems.

References

1. Aydemir-Karadag, A., & Turkbey, O. (2013). Multi-objective optimization of stochastic disassembly line balancing with station paralleling. *Computers and Industrial Engineering*, 65, 413–425.
2. Baybars, I. (1986). A survey of exact algorithms for the simple assembly line balancing problem. *Management Science*, 32, 909–932.
3. Baykasoglu, A. (2006). Multi-rule multi-objective simulated annealing algorithm for straight and U type assembly line balancing problems. *Journal of Intelligent Manufacturing*, 17, 217–232.
4. Becker, C., & Scholl, A. (2006). A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research*, 168, 694–715.
5. Chica, M., Cordón, Ó., & Damas, S. (2011). An advanced multi objective genetic algorithm design for the time and space assembly line balancing problem. *Computers and Industrial Engineering*, 61, 103–117.
6. Hamta, N., Fatemi Ghomi, S. M. T., Jolai, F., & Akbarpour Shirazi, M. (2013). A hybrid PSO algorithm for a multi-objective assembly line balancing problem with flexible operation times, sequence-dependent setup times and learning effect. *International Journal of Production Economics*, 141, 99–111.
7. Nourmohammadi, A., & Zandieh, M. (2011). Assembly line balancing by a new multi-objective differential evolution algorithm based on TOPSIS. *International Journal of Production Research*, 49, 2833–2855.
8. Ogan, D., & Azizoglu, M. (2015). A branch and bound method for the line balancing problem in U-shaped assembly lines with equipment requirements. *Journal of Manufacturing Systems*, 36, 46–54.

9. Ponnambalam, S. G., Aravindan, P., & Mogileeswar Naidu, G. (2000). A multiobjective genetic algorithm for solving assembly line balancing problem. *International Journal of Advanced Manufacturing Technology*, 16(5), 341–352.
10. Zhang, W., & Gen, M. (2011). An efficient multiobjective genetic algorithm for mixed-model assembly line balancing problem considering demand ratio-based cycle time. *Intelligent Manufacturing*, 22, 367–378.

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran