Indian Journal of Science and Technology, *Vol 8 (S8), 424–427, April 2015*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
DOI: 10.17485/ijst/2015/v8iS8/70479

An Ant Colony Algorithm for Assembly Line Balancing Problems with Relaxable Constraints

Jung Man Hong[1], Geun-Cheol Lee[2], Kilhwan Kim[3] and Seong-Hoon Choi[3*]

[1]Entrue Consulting, LG CNS, South Korea
[2]College of Business Administration, Konkuk University, South Korea
[3]Department of Management Engineering, Sangmyung University, South Korea; shchoi@smu.ac.kr

Abstract

In this paper, we consider the problem of balancing the assembly line, which is a flow-oriented production system suitable for mass production products such as home appliances and automobiles, etc. Since large capital investments are required for the installation of an assembly line, we should investigate various design alternatives as possible. In order to reduce the efforts of the line design engineers, this study presents a metaheuristic based on the ant colony algorithm for examining the effects when the constraints of task time and precedence are released. Experimental tests showed that the proposed algorithm obtained the solution very close to the optimal solution.

Keywords: Ant Colony Algorithm, Assembly Line Balance, Heuristic

1. Introduction

The assembly production line is one of the most well-known and used production systems, where standardized products are efficiently produced in high volume[1,2]. The performance of an assembly line is mainly affected by the process design in terms of workload balancing across the workstations which constitute the line. This famous problem is called Assembly Line Balancing Problem (ALBP), where tasks of each workstation are decided in a way that the unassigned task time is minimized.

As mentioned, the problem is famous that there has been enormous number of previous studies regarding ALBP. Recent survey papers summarized the various literatures[1,3]. A simple ALBP (S-ALBP) can be formulated mathematically as follows, which is introduced in many literatures[4-6]. The notations used in the model are presented in Table 1 also introduced by Moon and Choi[6].

$$Min\ \alpha C + \beta \sum_{k \in N} x_{dk} \tag{1}$$

Subject to

$$\sum_{j \in N} x_{ij} = 1, \forall i \in M \tag{2}$$

$$\sum_{i \in M} t_i x_{ij} \le C, \forall j \in N \tag{3}$$

$$\sum_{k \in N} kx_{ik} \le \sum_{k \in N} kx_{ij}, \forall (i,j) \in P \tag{4}$$

In the next section, we introduce a modified mathematical model in which the processing times of some task can be reduced as well as the constraints of precedence relationship between some task pairs can be relaxed. Section 3 presents a heuristic algorithm based on the Ant Colony Optimization (ACO) for the considered problem. In section 4, experimental tests were performed to investigate the performance of the proposed heuristics and the results are presented. The conclusion and comments on further research are presented in the last section.

**Author for correspondence*

Table 1. Notations

m	Number of tasks
n	Number of workstations
M	Set of tasks; $M = \{1, 2, \ldots , m\}$
N	Set of stations; $N = \{1, 2, \ldots , n\}$
P	Set of direct precedence relations; $P = \{(i, j)\mid i$ is direct precedent task of j, and $i, j \in M\}$
d	An imaginary final task with t_d=0. If x_{dk} = 1, then the optimal number of workstations is k.
x_{ij}	Decision variable, which is 0 or 1. If task i is assigned to station j, x_{ij}= 1; otherwise, 0.
α, β	Weighted value depending on problem type
C	Cycle time
t_i	Task time of task i

2. Problem Definition and Modeling

Two basic data in ALBP are processing time and the precedence relationship for each task[2,7]. Usually, they are pre-given in ALBP. In this study, however, these two pre-given data are considered as variables. That is, we can change the time of a certain task, and also precedence relationship can be relaxed under certain conditions. In this study, we analyze the effects of Line Balancing (LB) efficiency according to relaxing the above mentioned constraints.

With the new consideration of the problem, the modified mathematical model is presented as follows. Additional notations used in the model are presented in Table 2.

$$Min\ \alpha C + \beta \sum_{k \in N} x_{dk} - \sum_{i \in R^T} \left(p_i \sum_{i \in N} s_{ij} \right) + \sum_{\forall (i,j) \in R^P} v_{ij} p_{ij}^P \tag{1}$$

Subject to

$$\sum_{j \in N} x_{ij} = 1, \forall i \in M \tag{2}$$

$$\sum_{i \in R^T} s_{ij} + \sum_{i \in (M - R^T)} t_i x_{ij} \le C,\ for\ j \in N \tag{3}$$

$$\sum_{k \in N} k x_{ik} - B_{ij} g_{ij} \le \sum_{k \in N} k x_{jk},\ for\ (i,j) \in P \tag{4}$$

$$q_i t_i x_{ij} \le s_{ij} \le t_i x_{ij}, \text{for } i \in R^T \text{ and } j \in N \tag{5}$$

Table 2. Additional notations

R^T	Set of candidate tasks of which task time can be reduced
V_{max}^T	Maximal selected number of tasks
q_i	Maximal decreasing rate of t_i
P_i^T	Penalty for one unit decrease of t_i
s_{ik}	Decision variable that presents task i assigned to station k with a new task time
v_i	Temporary binary variable
R^P	Set of candidate precedence constraints that can be removed
V_{max}^P	Maximal selected number of precedence constraints
p_{ij}^P	Penalty for removing precedence constraint (i, j)
g_{ij}	Decision variable, 0 or 1
B	A number much larger than zero
B_{ij}	$B_{ij} = B$, *if* $(i, j) \in R^P$. *Otherwise, 0*
C_0	Cycle time of the optimal solution
C_p	Cycle time of the best solution obtained by the proposed algorithm

$$t_i - \sum_{j \in N} s_{ij} \le B v_i, \forall i \in R^T \tag{6}$$

$$\sum_{i \in R^T} v_i \le V_{max}^T \tag{7}$$

$$\sum_{(i,j) \in R^P} g_{ij} \le V_{max}^P \tag{8}$$

As you can see, a penalty $-\sum_{i \in R} \left(p_i \sum_{j \in N} s_{ij} \right)$ is added to objective function (1) to prevent task time being decreased without improving the ALB efficiency. Equation (2) is the same with the original formulation.

Equations (3) and (4) are changed into equations (3') and (4'), and equations (5–8) are added. Equation (3) is changed and equations (5–7) are added to examine the increase of the LB efficiency when task time is shortened. Equation (4) is changed and equation (8) is added to evaluate the increase of the LB efficiency when the precedence relationship constraints are relaxed.

3. An Ant Colony Metaheuristic Algorithm

In this study, we use ACO, which is one of metaheuristics, imitating the aspect of ants' finding optimal path from their nest to the food source[8,9]. The main features of this heuristic are two: the distributed optimization and the stochastic search. Finding an optimal path by ants is not controlled centrally, rather by the interactions of information among them. In their way to find the path, they depend on pheromone trail left by other ants. As the number of ants increase on a certain path, the pheromone trail remains get stronger.

Ant naturally can find the shortest path from the nest to food source without sophisticated or central control. There is also no direct contact or exchange of information among them. ACO is based on the indirect communication of a group of ants. The communication is through pheromone trails. Each ant uses the pheromone trail to construct its own path to the food source. The pheromone trail provides numerical information so that the ant probabilistically selects which way to go based on this information. The pheromone is updated iteratively to reflect new selection of path.

When each ant builds a solution, the ant starts with an arbitrary selected node. And, ant selects which task should be assigned among the remaining non-assigned tasks. If the task is selected, machine is assigned for the task. Each selection of task and machine is represented by graph with nodes and links. Selection of node by ant is done by using pheromone information, τ_{ij}, and heuristic information, η_{ij}. The heuristic information is the pre-defined preference information for the selection of node *j* from node *i*. If the problem is to find the shortest path, the heuristic information will be a reciprocal value of a distance between two nodes. For the line balancing problems, various alternatives can be considered for the heuristic information. In this problem a reciprocal value of task time is considered for the heuristic information. The selection of the node can be based on the maximum calculated value of pheromone and heuristic information or random selection. The choice of which method will be used for the selection is also decided based on probabilistic selection. The probability value for the random selection q_r are given. If the next should selected based on the selection, the probability for selection is uniform. If the next node should be selected based on the pheromone and heuristic information the follow formula will be used, where λ and μ are weights for the relative importance between pheromone information and heuristic information.

$$\arg\max_{(i,j)} \left\{ \left(\tau_{ij}\right)^{\lambda} \left(\eta_{ij}\right)^{\mu} \right\} \tag{9}$$

The pheromone will be evaporate with pheromone evaporation parameter, ρ ($0<\rho<1$). The following formula is for the evaporation where δ_{ij} is an indication function that has a value if the ant has selected link (*i*, *j*).

$$\tau_{ij} = \left(1-\rho\right)\tau_{ij} + \delta_{ij} \tag{10}$$

Two kinds of pheromone updating rules are considered in this paper, local updating and global updating. Local update rule is used to update pheromone according to the trail of each ant. Global update rule is considered to update pheromone on the trail of global best solution that has been found. With the global update rule, search process will converge more faster than without global update rule.

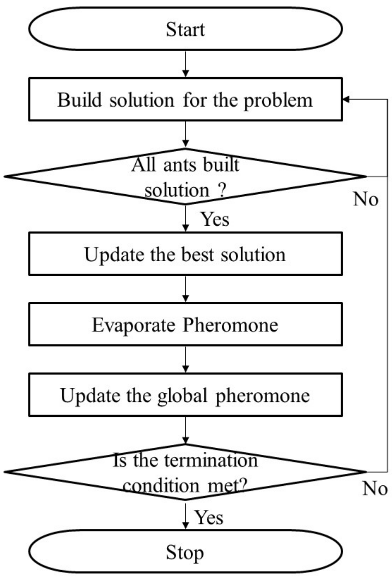

Figure 1. Overall procedure of ACO.

4. Test Results

Experimental tests were performed to investigate the performance of the proposed algorithm. Among the problems provided by the ALB research group, four target problems whose tasks numbered 11, 30, 58, and 94 were selected as the test sets, as presented in Table 3[7].

Table 3. Selection of the test set and the S-ALBP test results.

Test Set No.	Precedence Graph		Optimal Solution		Proposed Algorithm Solution		RPD (%)
	Data name	Number of Tasks	Cycle Time (Sec.)	Number of Stations	Cycle Time (Sec.)	Number of Stations	
1	MANSOOR	11	48	4	48.0	4	0.00%
2	SAWYER30	30	34	10	34.0	10	0.00%
3	WARNECKE	58	104	15	108.6	15	2.88%
4	MUKHERJE	94	201	22	207.8	22	2.99%

First, to investigate the performance of the proposed algorithm on the S-ALBPs and to seek the initial solutions for addressing the problems of the test sets, the problems were investigated under the assumption that there is no element task whose task time could be reduced, wherein the test target problems needed to minimize the cycle time when the number of the workstations was given. The test results were arranged as presented in the "Proposed Algorithm Solution" column.

To evaluate the performance of the proposed algorithm, the evaluation scale and relative percentage deviation (RDP) (see equation (11)) is in the column at the far right-hand side[6].

$$\mathrm{RDP} = \frac{C_P - C_O}{C_O} \times 100 \qquad (11)$$

The test proved, with regard to the large-sized problems, that the relatively precise solutions with less than 4% errors were sought.

5. Conclusion and Further Research

This paper proposed a heuristic using ACO for the problem of ALBP with constraints relaxed ideas. Preliminary computational experiments showed that the proposed algorithm obtained the solution very close to the optimal solution.

The considered problem can be extended considering more practical actions (i.e., relaxing constraints) to improve efficiency of the line balancing, such as, splitting a task with long processing time, sharing some tasks between two workstations, and so on.

6. References

1. Becker C, Scholl A. A survey on problems and methods in generalized assembly line balancing. Eur J Oper Res. 2006; 168:694–715.
2. Hwang H. Work study, 2nd edition. Seoul:Youngchi Publishers; 2005.
3. Becker C, Scholl A. State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. Eur J Oper Res. 2006; 168:666–93.
4. Kim YG, Kwon SH. 0-1 programming formulations for assembly line balancing of large-sized product. Korean Management Science Review, 1992; 17(1):55–65.
5. Choi SH. An integer programming model for generalized assembly line balancing problems. Productivity Review, 2011; 25(1):411–34.
6. Moon BH, Choi SH. A new paradigm for the more flexible assembly line balance study. Int J Appl Eng Res. 2014
7. ALB-Research Group; 2015. Avaliable from:http://www.assembly-line-balancing.de/
8. Dorigo M, Stutzle T, Ant colony optimization: overview and recent advances. In Handbook of Metaheuristics. Springer US; 2010.
9. Dorigo M, Caro G, Gambardella L. Ant algorithms for discrete optimization. Artificial life.1999; 5: 137–72.