Designing of an Assembly Line based on Reliability Approach

Dilip Roy [a] and Debdip Khan [b]

[a] Centre for Management Studies, The University of Burdwan, India-713104
Email: dr.diliproy@gmail.com

[b] Department of Business Administration, Burdwan Raj College, India-713101
Email: debdip.khan@gmail.com

Abstract. The problem of Assembly Line Balancing is to assign a set of tasks to an ordered sequence of workstations without violating the precedence constraints. The efficiency of the line will increase when tasks are more evenly distributed. In general, the efficiency measure(s) should be optimized subject to cycle time restriction and precedence constraints. Under the deterministic setup, efficiency of the system can be measured in various ways. Research works, reported so far, mainly deal with balancing loss as an inverse measure of efficiency. As a result, in earlier works balancing loss has been minimized subject to precedence constraints.In case the work elements are best described in terms of stochastic time, the entire problem has to be addressed with a different measure of efficiency. Expected variance of the idle times of workstations can be viewed as an inverse measure of stability of the system. A more appropriate and direct measure could be the reliability of the system such that each workstation adheres to assigned cycle time with high chance.

The present work defines the reliability of the assembly line in terms of cycle time and distribution of the tasks times and offers an optimization formulation for the problem under precedence constraints. For demonstration purpose, one well known example in the literature has been addressed under stochastic setup.

Notation

a(i,j)   binary measure taking value 1 for assignment of task i to workstation j

B   balancing loss

C   cycle time

C_{min} minimum cycle time for a given K

C_t trial cycle time

E(.)   statistical expectation operator

K   number of jobs

L_j variable idle time of jth work station (necessarily nonnegative)

N   number of workstations

N_{min} minimum number of workstation for a given cycle time

$N(\mu, \sigma^2)$ normal distribution with mean μ and variance σ^2

R_{AL} reliability of the assembly line

R_j reliability of the jth workstation

S_t slackness for trial cycle time C_t, i.e., $S_t = C - C_t$

t_i independent and random task time or assembly time of ith job

W_j j^{th} workstation

μ_i expected task time of ith job

σ_i^2 variance task time of ith job

Corresponding Author. Email: debdip.khan@gmail.com

1. Introduction

Assembly Line method has become a popular and important technique in production involving high volume. In view of its low cost production it has gradually replaced the traditional production methods.

Given a set of independent, identifiable and indivisible tasks of various durations, a set of precedence constraints among the tasks, one has to assign each task to exactly one workstation in such a way that no precedence constraint is violated and the assignment is complete and optimal in some sense (Becker and Scholl [1]). By precedence constraints we mean the technical restriction which demands the completion of some tasks before undertaking another task. Thus, Balancing of Assembly line, which is a very high rate of output ensuring difficult optimization problem, is restricted by precedence constraints, and cycle time constraints.

Attempts to solve the line balancing problems started during 1950s. Initially, the main focus was to design and configure workstations and assign tasks to workstations in a heuristic manner. Bowman[2] first considered the linear programming approach to arrive at an optimum solution to the line balancing problem. Thereafter, several researchers used different optimization techniques for solving the problem of line balancing. Hoffman [3], Mansoor and Yadin [4] and Geoffrion [5] used mathematical programming approach to present a clear formulation of the problem and arrive at the solution. Baxey [6] emphasized on the configuration of multiple workstations. Later, Integer programming procedure was used by Graves and Lamer [7] for designing an assembly system. Sarin and Erel [8] developed a cost model for the single product assembly line balancing problem for minimizing the total labour cost. Berger et al [9] adopted Branch-and-bound algorithms for the multi-product assembly line balancing problem. The problem of balancing assembly lines with stochastic task processing times using simulated annealing was addressed by Suresh and Sahu [10]. Pinnoi and Wilhelm [11] used the branch and cut approach for system design. In 2002, Nicosia et al [12] introduced the concept of cost and studied the problem of assigning operations to an ordered sequence of non-identical workstations, which also took into consideration the precedence relationships and cycle time restrictions. Erel et al [13] presented a beam search-based method for

the stochastic assembly line balancing problem in U-lines. Zhao et al [14] dealt with sequence-to-customer goal with stochastic demands for a mixed-model assembly line for minimizing the number of stations. In 2006, a branch-and-bound based solution was proposed by Bukchin and Rabinowitch [15]. Gu et al [16] tried to solve assembly line balancing problem by estimation of distribution. Agarwal and Tiwari [17] proposed a collaborative ant colony algorithm to stochastic mixed-model U-shaped disassembly line balancing and sequencing problem. Gamberini et al [18] presented a multiple single-pass heuristic algorithm solving the problem of stochastic assembly line rebalancing. Roy and Khan [19] addressed the optimization of an integrated line balancing process with workstation inventory management. Roy and Khan [20] also tried to balance assembly line by minimizing balancing loss and system loss.

2. Problem Description

In most of the above mentioned cases, the only important consideration for assuring efficiency in line balancing was to minimize the cost of assignment through balancing loss. These methods are very useful for transfer lines where lines are fully automated and line elements are preferably performed by machines or robots in a nearly deterministic manner. But in case of Assembly lines, human beings are involved and they have the problem of variable operation times for the same task. So, assembly line balancing problem is not only the problem of line design with nearly equal distribution of tasks among the stations or the adaptation of tasks to the speed of the workers but also to provide some amount of slackness in each workstation to take care of the stability of the system. It may be pointed out that the success of an organization depends not only on quality and reliability of the final product, but also on the reliability of the production set up. Otherwise, system failure may result in irregular supply of the item which will reduce the customer base and hence the profit of the organization by increasing the cost of production or loss of customers or both. So, there should be both reliable products with reliable production set up for smooth and stable functioning of the production activities.

The objective of the current work is to present a mathematical formulation for designing a stable assembly line where both chance of system failure and number of workstations will be

minimum. Equivalently, expected balancing loss has been minimized under the stochastic domain to generate an initial set of feasible solutions and then the reliability of the assembly line has been maximized. Thus, we propose a two-stage optimization method and use stochastic simulation approach to solve the final problem.

3. Proposed Methodology

Under the deterministic setup, the uneven allocation of works to different workstations results in loss of efficiency. The efficiency of an assembly line is therefore measured in terms of balancing loss, $B = (NC - \sum t_i)/NC$. Under the stochastic setup since t_i's are random variables this balancing loss itself becomes a random variable. So, one may like to minimize the expected value of the same, i.e.,

$$E(B) = \frac{(NC - \sum \mu_i)}{NC}$$

But this measure alone is not sufficient to ensure efficiency of the production system. For example, for a perfectly balanced situation with E(B) = 0, the chance of failure of an assembly line under symmetric distribution of each workstation time is $\left(\frac{1}{2}\right)^N$ which tends to zero as the number of workstations becomes large. So, there must be some other consideration for ensuring high chance of meeting the cycle time requirements in each workstation. Drawing analogy with the concept of product reliability in terms of meeting the mission requirement, we may define the reliability of a workstation in terms of idle time meeting the non-negativity restriction. Thus, reliability of jth workstation, R_j can be defined as

$$R_j = Pr. \left[L_j \geq 0 \right].$$

Then the assembly line can be viewed as an arrangement of N workstations in series in the sense if one workstation fails to meet the cycle time requirement the entire assembly line faces operational failure. This observation translated in terms of reliability indicates $R_{AL} = \prod_{j=1}^{N} R_j$, where R_{AL} is the reliability of the assembly line. We propose to consider the reliability of the assembly line along with the expected balancing loss as dual measure of system efficiency. Thus, the efficiency of the total system will be maximum when both the expected balancing loss

will be minimum and the system reliability will be maximum. Therefore, the objective of our proposed method can be equivalently expressed as the minimization of the number of workstations (N) and maximization of the system reliability, R_{AL}, subject to precedence constraints.

4. Mathematical Formulation

Let us consider the binary variable $a(i, j)$ such that

$$a(i,j) = \begin{cases} 1, & \text{if } i \in W_j, \text{i th task is assigned to Wj} \\ 0, & \text{if } i \notin W_j, \text{i th task is not assigned to Wj} \end{cases}$$

and is true for i = 1, 2, ..., K, j = 1, 2, ..., N. The following condition must hold for each i = 1, 2, ..., K, under the restriction that the ith task can be assigned to only one workstation:

$$\sum_{j=1}^{N} a(i,j) = 1 \tag{1}$$

Further, according to precedence constraints if task i' is to be assigned before assigning task i, that is $i' < i$, then

$$a(i,j) \leq \sum_{r=1}^{j} a(i',r) \quad \forall \, i' < i \quad \forall \, j \tag{2}$$

Since the task times are random variables, the condition for completion of tasks in a workstation within the assigned cycle time can be described in terms of reliability measure,

$$R_j = Pr.\left[L_j \geq 0 \right] = Pr.\left[\frac{L_j - E(L_j)}{\sqrt{var(L_j)}} \geq -\frac{E(L_j)}{\sqrt{var(L_j)}} \right]$$
$$= 1 - \Phi\left(-\frac{E(L_j)}{\sqrt{var(L_j)}} \right),$$

under normality of the each elemental times. Thus,

$$R_j = \Phi\left(\frac{E(L_j)}{\sqrt{var(L_j)}} \right)$$

The reliability of the assembly line, R_{AL}, can be expressed as

$$R_{AL} = \prod_{j=1}^{N} \Phi\left(\frac{E(L_j)}{\sqrt{var(L_j)}} \right)$$

following the properties of the series system and the fact that workstations are arranged in series.

Thus, the optimization framework of the line balancing problem can be expressed in terms of the following objectives:

Minimize   N

Maximize $R_{AL} = \prod_{j=1}^{N} R_j$

subject to the following constraints, :

(i) $\sum_{j=1}^{N} a(i,j) = 1$ $\forall \quad i$

(ii) $a(i,j) \leq \sum_{r=1}^{j} a(i',r)$ $\forall \quad i' < i$

(iii) $a(i,j) = 0,1$ $\forall \quad i, j$

We prefer to address the above optimization problem in two stages. First, we undertake the task of minimization of E(B) and thereby generate, in the first instant, feasible solutions with minimum value of E(B). Then we obtain the final solution of the problem by imposing the second objective of maximization of R$_{AL}$. Even for generating the set of feasible solutions, we consider a sequential approach of assigning trial cycle time that results in slack time. This slack time is to be assigned to each workstation meeting the optimality condition arising out of the first objective of the above formulation. In this approach the trial cycle time starts from some lowest value and gets increased step by step so as to reach the maximum limit C. Determination of the lowest value for trial cycle time depends on the following consideration.

Given a choice of C, it may be noted that the theoretical minimum number of workstations, N$_{min}$, must satisfy the following constraints:

$$\sum_{i=1}^{K} T_i / C \leq N_{min} \leq \sum_{i=1}^{K} T_i / C + 1 \, ,$$

from where we arrive at C$_{min}$, the minimum value of C, as

$$C_{min} = \left[\sum_{i=1}^{K} T_i / N_{min} + 1 \right]$$

Thus, given a cycle time, C, one may conceptually consider a trial cycle time, C$_t$, satisfying the condition C$_{min} \leq$ C$_t \leq$ C, to arrive at the set of feasible workstation configurations and maintain the same cycle time C by uniformly adding to each workstation a slackness S$_t$ to C$_t$, where S$_t$ = C - C$_t$. This will help to increase the system stability.

5. The Algorithm

Our proposed two-stage procedure with sequential generation of feasible solution and selection of final solution can be best described by the following algorithm.

1. Calculate the theoretical minimum number of workstations, N$_{min}$, following the formula $\sum_{i=1}^{K} \frac{\mu_i}{C} \leq N_{min} \leq \sum_{i=1}^{K} \frac{\mu_i}{C} + 1$. Calculate the minimum cycle time, C$_{min}$, using the relationship, $C_{min} = \left[\sum_{i=1}^{K} \frac{\mu_i}{N_{min}} + 1 \right]$

2. Set the trial cycle time C$_t$ at C$_{min}$.

3. Prepare the list of all unvisited tasks – call it List *U*.

4. Prepare List *R* from the tasks of List *U* with no immediate predecessor or whose immediate predecessors have been visited. The tasks of *R* are ready for selection.

5. Prepare List *A* from the task of List *R* having assembly time less than that of cycle time and is allowable for inclusion.

6. Randomly select a task from List *A* and reset the cycle time as {C$_t$ – assembly time}.

7. If cycle time is less than the assembly time, then open a new workstation. Reinitialize cycle time to its original value and repeat the above steps until all nodes are visited.

8. After getting the complete distribution of tasks to workstations, calculate R$_{AL}$, the reliability of the assembly line.

9. After each run, the new reliability value R_{AL} is compared with the previous R_{AL} value. If the new R_{AL} value is greater than the previous value, the new solution is stored as the basis for next comparison.

10. Increase the cycle time by one unit until it crosses *C* value. If *C* value is crossed, go to step 12.

11. Repeat step 2 to 10.

12. Check whether all the work elements have been assigned to specified number of work stations. If not, increase the value of N$_{min}$ by 1 and go to step 2.

13. Print the best solution in terms of overall maximum reliability.

6. Worked Out Example

Figure 1 represents an assembly line balancing problem. This is a famous problem studied in Ray Wild [21]. We have adopted it for the purpose of explaining how the proposed model works. The numerical figure within a circle represents the task number.

In Table 1, the above mentioned problem is summarized in terms of work elements, immediate predecessor(s), expected task durations and their variances. We assume independent normality for each task duration. Using the above table, we can easily get the minimum number of workstation, N_{min} as 5. So, minimum trial cycle time, C_{min}, comes out as

$$C_{min} = \left[\sum_{i=1}^{K} \frac{\mu_i}{N_{min}} + 1 \right], \text{ i.e. } C_{min} = 29 \text{ time units.}$$
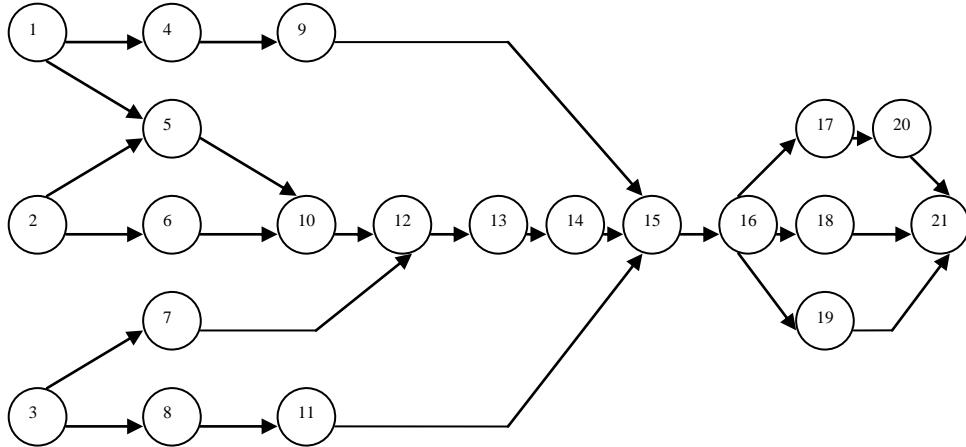
Figure 1: Precedence diagram of work elements.

Table 1: Precedence relation and task times of work elements.

Work Element (i)	Immediate Predecessor	Expected Activity Time (μ_i)	Variance of activity time (σ_i^2)
1	-	6	0.09
2	-	5	0.0625
3	-	8	0.16
4	1	9	0.2025
5	1, 2	5	0.0625
6	2	4	0.04
7	3	5	0.0625
8	3	6	0.09
9	4	10	0.25
10	5, 6	5	0.0625
11	8	6	0.09
12	10, 7	2	0.01
13	12	5	0.0625
14	13	4	0.04
15	9, 11, 14	12	0.36
16	15	10	0.25
17	16	5	0.0625
18	16	15	0.5625
19	16	10	0.25
20	17	5	0.0625
21	18, 19, 20	6	0.09

Table 2: Trial Configurations

Trial Cycle Time	Sl	Work Station 1	Work Station 2	Work Station 3	Work Station 4	Work Station 5	R_{AL}
31	1	1,3,2,5,8	6,4,10,11,7,12	13,9,14,15	16,19,17, 20	18,21	0.873450476
	2	1,3,8,2,7	6,11,5,10,4,12	13,9,14,15	16,17,19, 20	18,21	0.873450476
	3	2,3,7,8,1	11,6,4,5, 10,12	9,13,14,15	16,18,17	20,19, 21	0.853130899
	4	2,3,1,7,8,	5,11,6,10,12,13,14	4,9,15	16,18,17	20,19, 21	0.852897841
	5	2,3,7,8,11	1,4,6,5,10,12	9,13,14,15	16,19,17, 20	18,21	0.873450476
	6	3,1,7,8,11	2,6,5,10, 12,4	9,13,14,15	16,19,17, 20	18,21	0.871655455
	7	3,7,8,1,2	5,6,10,12,13,14,11	4,9,15	16,19,17, 20	18,21	0.873211867
	8	2,3,1,5,8	11,6,10,4,7,12	13,9,14,15	16,18,17	19,20 21	0.853130899
	9	3,7,8,11,2	6,1,5,4,10,12	13,14,9,15	16,18,17	20,19, 21	0.853130899
	10	1,2,3,8,11	4,6,7,5,10,12	13,14,9,15	16,17,18	20,19, 21	0.851377636
32	1	1,2,6,5,10	3,7,12,8,4	11,9,13,14	15,16,17, 20	19,18, 21	0.7750863
	2	2,6,1,5,10	4,3,8,7,12	13,11,14,9	15,16,19	17,18, 20,21	0.771341063
	3	2,6,1,5,10	3,8,11,4	7,12,9,13,14	15,16,19	18,17, 20,21	0.778812826
	4	1,2,6,5,10	4,3,8,11	9,7,12,13,14	15,16,17, 20	19,18, 21	0.782594343
	5	2,1,5,6,10	3,4,7,12, 13	14,8,11,9	15,16,17, 20	18,19, 21	0.783706858
	6	2,6,3,7,1	5,10,12,13,8,14	4,11,9	15,16,17, 20	18,19, 21	0.786286827
	7	2,1,6,3,5	7,10,12,13,14,8	11,4,9	15,16,19	18,17, 20,21	0.782487468
33	1	2,3,1,8,7	6,11,4,5, 10,12	9,13,14,15	16,18,17	19,20, 21	0.853130899
	2	3,2,1,8,11	6,5,7,4,10,12	13,14,9,15	16,19,17, 20	18,21	0.871655455
	3	3,7,8,2,11	6,1,5,10, 12,4	9,13,14,15	16,18,17	20,19, 21	0.853130899
	4	3,1,7,8,2	4,11,5,6, 10,12	9,13,14,15	16,17,19, 20	18,21	0.873450476
	5	1,2,5,3,8	6,4,7,11, 10,12	13,14,9,15	16,19,17, 20	18,21	0.873450476
	6	3,8,1,2,11	7,4,6,5,10,12	9,13,14,15	16,19,17, 20	18,21	0.871655455
	7	2,6,3,8,7	1,5,4,9	11,10,12, 13,14	15,16,17, 20	18,19, 21	0.769756335
	8	3,7,2,6,8	1,4,9,5	11,10,12, 13,14	15,16,19	18,17, 20,21	0.766036852
	9	3,8,1,4	9,2,5,6,10	11,7,12,13,14	15,16,19	18,17, 20,21	0.776466963
	10	1,3,2,6,8	5,10,4,9	11,7,12,13,14	15,16,17, 20	19,18, 21	0.779904332
	11	2,6,1,3,8	4,5,7,9	11,10,12, 13,14	15,16,19	18,17, 20,21	0.776135813
	12	1,2,4,6,5	3,7,8,9	11,10,12,13,14	15,16,17, 20	19,18, 21	0.78007511
	13	1,2,6,4,5	10,9,3,8	11,7,12,13,14	15,16,19	18,17, 20,21	0.776305766
	14	2,1,4,5,6	10,9,3,8	7,11,12,13,14	15,16,17, 20	18,19, 21	0.78007511
	15	2,6,3,8,1	4,5,7,9	10,11,12, 13,14	15,16,17, 20	19,18, 21	0.779904332
	16	1,3,4,8	7,2,5,9,6	10,12,11, 13,14	15,16,17, 20	19,18, 21	0.780237089
	17	2,1,4,6,5	3,7,10,12,13,14	9,8,11	15,16,19	18,17, 20,21	0.77973899
34	1	3,7,8,1,11	2,6,5,4,10, 12	9,13,14,15	16,19,17, 20	18,21	0.871655455
	2	3,7,8,11,1	2,5,6,10,4, 12	13,9,14,15	16,18,17	19,20, 21	0.851377636
	3	3,2,8,1,11	7,5,6,4,10, 12	13,9,14,15	16,18,17	19,20, 21	0.851377636
	4	2,3,8,11,1	6,5,7,10, 12,13,14	4,9,15	16,17, 20, 19	18,21	0.86378678
	5	3,2,8,11,1	7,5,4,6,10, 12	13,9,14,15	16,19,17, 20	18,21	0.871655455
	6	3,1,8,11,2	6,7,5,10, 12,13,14	4,9,15	16,17,18	20,19, 21	0.843692015
	7	3,8,11,1,7	2,5,6,10, 12,13,14	4,9,15	16,19,17, 20	18,21	0.86378678
	8	3,7,8,1,11	2,6,5,10, 12,13,14	4,9,15	16,18,17	20,19, 21	0.843692015
35	1	2,1,6,4,3	7,9,8,11,5	10,12,13, 14,15	16,19,17, 20	18,21	0.788077309
	2	2,1,6,3,4	8,11,5,9, 10	7,12,13, 14,15	16,17,19, 20	18,21	0.788077309

Table 3: Final Optimum Configuration

C	Work Station 1	Work Station 2	Work Station 3	Work Station 4	Work Station 5	R_{AL}
31	2, 3, 7, 8, 11	1, 4, 6, 5, 10, 12	9, 13, 14,15	16, 19, 17, 20	18, 21	0.873450476

Now, we consider in the final solution the Cycle time C as 35 time units. So, the trial cycle time starts from 29 time units and goes upto 35 time units. For the given problem, we get no feasible solution for the trial cycle times as 29 and 30 time units. For the rest of the cycle times we get feasible solutions. These trial configurations are presented in Table 2.

The final solution based on optimization criterion is presented in Table 3 for trial cycle time as 31 time units.

In the optimum configuration it is found that the optimum value of R_{AL} is 0.873450476 having the configuration of 5 workstations with work elements 2, 3, 7, 8, 11 assigned to workstation 1, work elements 1, 4, 5, 6, 10, 12 assigned to workstation 2, work elements 9, 13, 14, 15 assigned to workstation 3, work elements 16, 19, 17, 20 assigned to workstation 4 and work elements 18, 21 assigned to workstation 5.

7. Conclusion

Since system failure is due to variations in human behavior, a stochastic setup has been considered for describing the situation. Further, reliability of a production system is as important as the product reliability and we have for the first time considered reliability optimization problem for an assembly line. For that purpose, a mathematical programming approach has been followed. A two-stage approach has been installed to arrive at the final solution to the dual objective problem of minimization of expected balancing loss and maximization of system reliability. For the purpose of reliability evaluation we have assumed normal distribution for task times. However, the proposed approach is a generic one and capable of solving different large or small, simple or complex assembly problems under different distributional assumptions.

References

[1] Becker, C and Scholl, A., A survey on problems and methods in generalized assembly line balancing, *European Journal of Operational Research,* 168, 694-715 (2006).

[2] Bowman, E. H., Assembly Line Balancing by Linear Programming, *Operations Research*, 385-389 (1960).

[3] Hoffmann, T. R., Assembly line balancing with precedence matrix, *Management Science*, 9, 551-562 (1963).

[4] Mansoor, E. M. and Yadin, M., *On the problem of assembly line balancing, Development in Operations Research,* edited by B. Avi-ltzhak, Gordon and Breach, New York (1971).

[5] Geoffrion, A. M., The purpose of mathematical programming is insight, not numbers, *Interface*, 7(1), 81-92 (1976).

[6] Baxey, G. M., Assembly line balancing with multiple stations, *Management Science*, 20, 1010-21, (1974).

[7] Graves, S. C. and Lamer B. W., An integer programming procedure for assembly system design problems, *Operations Research,* 31, 522-545 (1983).

[8] Sarin, S. C. and Erel, E., Development of cost model for the single-model stochastic assembly line balancing problem, *International Journal of Production Research*, 28(7), 1305-16 (1990).

[9] Berger, I., Branch-and-bound algorithms for the multi-product assembly line balancing problem, *European Journal of Operations Research,* 168, 694-715 (1992).

[10] Suresh, G. and Sahu, S. Stochastic assembly line balancing using simulated annealing, *International Journal of Production Research*, 32 (8), 1801-1810 (1994).

[11] Pinnoi, A. and Wilhelm, W. E., Assembly system design: A branch and cut approach, *Management Science,* 44, 103-118 (1998).

[12] Nicosia G., Pacciarelli, D. and Pacifici, A., Optimally balancing assembly lines with different workstations, *Discrete Applied Mathematics*, 118, 99-113 (2002).

[13] Erel, E., Sabuncuoglu, I. and Sekerci, Stochastic assembly line balancing using beam search, *International Journal of Production Research*, 43(7), 1411-1426 (2005).

[14] Zhao, X., Yeung, J. H. Y. and Xie, J., Sequence-to-customer goal with stochastic demands for a mixed-model assembly line, *International Journal of Production Research*, 44 (24), 5279-5305 (2006).

[15] Bukchin, Y. and Rabinowitch, I., A branch-and-bound based solution approach for the mixed-model assembly line-balancing problem for minimizing stations and task duplication costs, *European Journal of Operational Research*, 174, 492-508 (2006).

[16] Gu, L., Hennequin, S., Sava, A. and Xie, X., Assembly line balancing problem solved by estimation of distribution, *Automation science and Engineering, IEEE International Conference*, (2007).

[17] Agarwal, S. and Tiwari, M. K., A collaborative ant colony algorithm to stochastic mixed-model U-shaped disassembly line balancing and sequencing problem, *International Journal of Production Research,* 46(6), 1405-1429 (2008).

[18] Gamberini, R., Gebennini, E., Grassi, A. and Regattieri, A., A multiple single-pass heuristic algorithm solving the stochastic assembly line rebalancing problem, *International Journal of Production Research,* 47(8), 2141-2164 (2009).

[19] Roy, D. and Khan, D., Integrated model for line balancing with workstation inventory management, *International Journal of Industrial Engineering Computation,* 1, 139-146, (2010).

[20] Roy, D. and Khan, D., Assembly Line Balancing to minimize Balancing Loss and System Loss, *Journal of Industrial Engineering International*, 6 (11), 1-5 (2010).

[21] Wild, R., *Operation Management*, Thomson, London (2004).

Dilip Roy (dr.diliproy@gmail.com), Professor, Centre for Management Studies, The University of Burdwan, Burdwan 713104, WB, India. He did his doctoral work from the Indian Institute of Management, Calcutta with specialization in Operation Management. He also received a Ph.D. degree in Statistics from the University of Calcutta, and did post doctoral work at the Dalhousie University of Halifax, Canada. He is guiding Ph.D. scholars in both Statistics, and Management, and has contributed significantly in characterization, classification, and model building. He has published more than 150 research papers in reputed journals. Outlets for his work includes IEEE Transactions on Reliability, J. Multivariate Analysis, Communications in Statistics, Probability in Engineering and Information Science, J. Applied Probability, Micro Electronics and Reliability, Yokohama Mathematical J, JUSE, Economic Quality Control, Int. J. Quality & Reliability Management, Calcutta Statistical Association Bulletin.

Debdip Khan is working as lecturer at Burdwan Raj College, Burdwan - 713101, India. He has been teaching sincerely for the last five years. He is actively engaged in research in the field of Management under The Department of Business Administration at The University of Burdwan. His areas of research and interest are assembly line balancing, reliability engineering, optimization and production planning and control.