Available online at www.sciencedirect.com
ScienceDirect
IFAC PapersOnLine 52-13 (2019) 992–997

Type-1 U-shaped Assembly Line Balancing under uncertain task time

Yuchen Li *. Xiaofeng Hu. Xiaowen Tang.* Ibrahim Kucukkoc*****

**School of Economics and Management, Beijing University of Technology, 100124,China (e-mail: liyuchen@bjut.edu.cn;txw@bjut.edu.cn).*
***School of Mechanical Engineering, Shanghai Jiao Tong University, 200240, China, (e-mail: wshxf@sjtu.edu.cn)}*
****Department of Industrial Engineering, Balikesir University, 10145, Turkey, (e-mail: ikucukkoc@gmail.com)}*

Abstract: Recently, assembly line balancing problem with uncertain task time gains more and more attention in the literature. Task time uncertainty may overload workstations. Uncertain task time attributes were studied in the frameworks of the probability theory. In this paper, we use a new method, which is the uncertainty theory, to model the uncertain task time as the historical task time information is unavailable. We incorporate the uncertainty into the constraints of the type-1 U-shaped assembly line balancing problem. We derive some useful theorems related to the optimal solutions. Further, we develop an algorithm based on the branch and bound remember algorithm to solve the proposed problem. Finally, numerical studies are conducted to illustrate our model.

Keywords: Assembly line balancing; Uncertainty theory; Uncertain Programming; Uncertain task time attribute; Branch and bound remember.

1. INTRODUCTION

An assembly line is a manufacturing process in which parts are added as the semi-finished assembly moves from workstation to workstation where the parts are added in sequence until the final assembly is produced. The basic structure of the assembly line is that workers manufacture and assemble the product by performing a sequence of tasks along a moving device, e.g., conveyor belt. An example of data structure of an assembly line is presented in the following directed acyclic digraph (Figure 1). The number inside the node designates the task number and the number outside the node is the operation time (task time) for the task. The directed arcs show the precedence relationship between pairs of tasks. In Figure 1 there are 10 tasks with task time ranging from 2 to 9.

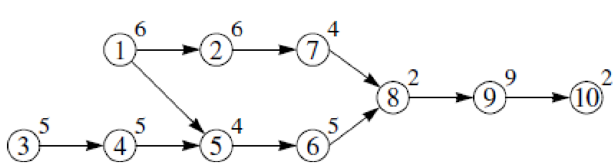

Figure 1 Precedence digraph of 10 tasks assembly line

Bryton (1954) first proposed the assembly line balancing problem and the first scientific research was done by Salveson (1955). Type-I balancing problem is to minimize the number of stations m given the cycle time c. There are several types of assembly lines including two-sided, U-shaped, parallel assembly lines and so on (Battaïa and Dolgui, 2013). In this paper, we focus on the U-shaped assembly line.

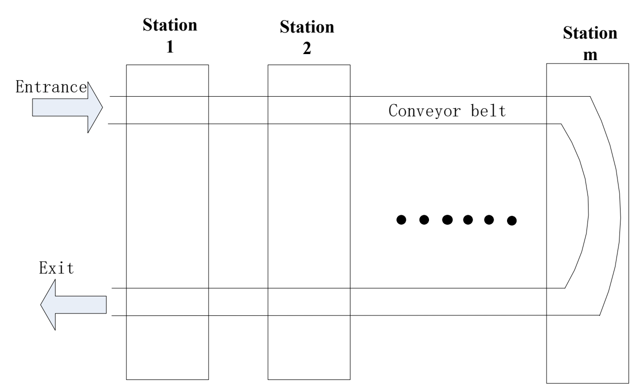

Figure 2 The basic structure of an U-shaped

The U-shaped line is proposed by Moden (1993). Figure 2 shows the basic structure of a U-shaped assembly line. The line (conveyor belt) is arranged like a "U" shape. A station can cross over the line and consists of two segments, entrance side and exit side. When a task is available to be assigned to the entrance/exit side of a station, all of its predecessors/successors must have been assigned before.

The paper addresses a practical problem that task time is uncertain and there is a lack of historical data of task times. Task time is uncertain during production due to the technology changes, environmental changes, and learning effect of the human workers (Li and Boucher, 2016). Most research utilizes the probability theory to model the task time uncertainty. However, we sometimes lack information for task times, especially when a new product is going to be assembled. In this regard, we should resort to some new methods other than probability theory to solve the problem.

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2019.11.324

The remainder of this paper is organized as follows. A brief literature review for recent studies is in Section 2. We propose an uncertain programming model which embodies the uncertainty of the task time in Section 3. An algorithm, based on branch and bound remember algorithm, is developed in Section 4. Numerical studies are conducted in Section 5. Section 6 concludes the paper. The notations are given in Table 1.

Table 1 Notations

n	the number of tasks
m	the number of stations
c	cycle time of the assembly line
t_i	task time for task i, y=1...n
x_{ij}	1, if task i is assigned to the entrance side of station j; 0, otherwise
y_{ij}	1, if task i is assigned to the exit side of station j; 0, otherwise
P	precedence relation, if $(v,o)\in$P, v is a predecessor of o
M(Λ)	the uncertain measure for event Λ which indicates the belief degree that event will occur
α	the required belief degree
Φ_i(x)	the uncertainty distribution for task i's task time, i=1…n
$\Phi_i^{-1}(\alpha)$	the inverse uncertainty distribution for task i's task time, i=1…n

2. LITERATURE REVIEW

U-shaped assembly line balancing with uncertain task time attribute has been studied in literature. Celik et al. (2014) used an ant colony method to rebalance U-lines with stochastic task times. Dong et al. (2014) minimized the expectation of overload time for U-shaped lines. Delice et al. (2016) proposed a novel stochastic two-sided U-shaped assembly line balancing problem and solved it by a genetic algorithm. Tiacci (2017) proposed a genetic approach to solve a mixed-model U-shaped assembly line considering stochastic task time.

There is a disadvantage of probability theory when it is utilized to model the uncertain task time because we need enough samples to derive the probability distribution. However, task time data is sometimes unavailable, especially when manufacturing new products. In this case, we should invite some experts of the subject matter to obtain *belief degrees* for these task times. Liu (2012) used some examples to show that belief degrees cannot be modeled by probability theory. In this paper, we modeled the uncertain task time under a new mathematical framework—Uncertainty theory to encompass the uncertain task time in assembly line balancing problem.

Uncertainty theory was found by Liu (2007) to model the belief degree and received a great deal of attention in academia. It has become an offshoot in mathematics for gauging the indeterminate phenomena. Later, Liu (2009) developed an uncertain programming model which is a mathematical programming pertaining to the uncertain variables. Then, uncertain programming was widely used to model the belief degree of some uncertain input in practical problems (Gao and Qin, 2016; Ke et al., 2015; Li and Liu, 2017; Wen et al., 2014). Uncertain programming becomes an efficient tool to handle the uncertainty in various combinatorial optimization problems, such as project scheduling, facility location-allocation, machine scheduling and so on. To this end, we employ the uncertain programming to optimize the type 1 U-shaped assembly line balancing problem (UALBP1) where the task times $\tilde{t}_i$ are treated as uncertain variables.

3. CHANCE-CONSTRAINED MODEL

In this section, we develop an uncertain programming model which is chance related. Chance-constrained programming was initiated by Charnes and Cooper (1961). It is a powerful tool to deal with a system with uncertainty. In a chance-constrained program, we optimize certain objective subject to some chance constraints where a fixed confidence interval α is specified. Task times t_i are modeled as independent uncertain variables with uncertainty distribution $\Phi_i(x)$. The chance constraints in our proposed problem are that the belief degrees for all stations not getting overloaded should be greater than or equal to α. We formulate the type-1 chance-constrained U-shaped assembly line balancing problem (CC-UALBP1) as follows. Constraint (1) describes a task can only be assigned to the entrance or exit side of one station, constraint (2)-(3) ensure that a task is available to be assigned when all of its predecessors or successors have been assigned. Constraint (4) is the chance constraint regarding the uncertainty of the task time t_i. The constraint (4) guarantees that the belief degree for a station not being overloaded is not less than α.

Min m (CC-UALBP1)

$s.t.$

$$\sum_{j=1}^{m}(x_{ij}+y_{ij})=1, \qquad \forall i=1,2,...,n \qquad (1)$$

$$\sum_{j=1}^{m} jx_{vj} \le \sum_{j=1}^{m} jx_{oj},\ \forall v,o=1,2,...,n,(v,o)\in P \quad (2)$$

$$\sum_{j=1}^{m} jy_{oj} \le \sum_{j=1}^{m} jy_{vj},\ \forall v,o=1,2,...,n,(v,o)\in P \quad (3)$$

$$M\{\sum_{i=1}^{n} t_i(x_{ij}+y_{ij}) \le c\} \ge \alpha.\ \ \forall j=1,2,...,m \quad (4)$$

In CC-UALBP1, the production manager can control the belief degree in advance to guarantee the chance for the line not getting overloaded is below the predetermined cycle time. The CC-UALBP1 is not a deterministic model which requires a great amount of computational effort to solve it, let alone its NP-hard attribute. Next, we propose a theorem to transform the CC-UALBP1 model into a crisp model, where the uncertain measure is removed.

Theorem 3.1 Let $t_1, t_2, ..., t_n$ be independent uncertain variables with regular uncertainty distributions $\Phi_1(x)$, $\Phi_2(x)$, … ,$\Phi_n(x)$ respectively. Then, $M\{\sum_{i=1}^{n} t_i(x_{ij}+y_{ij}) \le c\} \ge \alpha$ is equivalent to

$$\sum_{i=1}^{n} \Phi_i^{-1}(\alpha)(x_{ij}+y_{ij}) \le c.$$

Proof:
For any station j, let $S_j = t_i(x_{ij}+y_{ij})$. Obviously, S_j is an increasing function in t_i. According to inverse uncertainty distribution theorem (Liu, 2010), for any $0 \le \alpha \le 1$, we have

$$\Psi_j^{-1}(\alpha) = \sum_{i=1}^{n} (x_{ij}+y_{ij})\Phi_i^{-1}(\alpha),$$

where Ψ_j is an uncertainty distribution of S_j.
That is to say,

$$M\{S_j \le \sum_{i=1}^{n} (x_{ij}+y_{ij})\Phi_i^{-1}(\alpha)\} = M\{S_j \le \Psi_j^{-1}(\alpha)\} = \alpha \cdot$$

Thus,

$$M\{S_j \le c\} \ge \alpha,$$

$$\Leftrightarrow M\{S_j \le c\} \ge M\{S_j \le \Psi_j^{-1}(\alpha)\},$$

$$\Leftrightarrow M\{S_j \le c\} \ge M\{S_j \le \sum_{i=1}^{n} (x_{ij}+y_{ij})\Phi_i^{-1}(\alpha)\}.$$

We also know uncertainty distribution is an increasing function, this fact tells us that

$$\sum_{i=1}^{n} (x_{ij}+y_{ij})\Phi_i^{-1}(\alpha) \le c.$$

The theorem is proved. ■

Therefore, we can employ Theorem 3.1 to transform the constraint (4) into constraint (5), and a crisp model is obtained

$$\sum_{i=1}^{n} \Phi_i^{-1}(\alpha)(x_{ij}+y_{ij}) \le c. \quad \forall j = 1,2,...,m \qquad (5)$$

We can also find the relationship between α and m by the following corollary.

Corollary 3.1 Let $t_1, t_2, ..., t_n$ be independent uncertain variables with regular uncertainty distributions $\Phi_1(x)$, $\Phi_2(x)$, … , $\Phi_n(x)$ respectively. Then, the optimal m for CCALBP1 is nondecreasing in α.

Proof: Suppose we have α_1 and α_2, $\alpha_1 \geqslant \alpha_2$. We would like to prove the optimal solution of CC-UALBP1 (α_1) is also a feasible solution for CC-UALBP1 (α_2). Let $y^1 = \{y_{11}^1, …, y_{ij}^1, …, y_{mn}^1\}$ and $x^1 = \{x_{11}^1, …, x_{ij}^1, …, x_{mn}^1\}$ denote the optimal solution for CC-UALBP1 (α_1) and m_1 is the optimum. We plug $\{x^1, y^1\}$ into the CC-UALBP1 (α_2) to check whether the constraints are satisfied.

It is obvious that (1), (2) and (3) are satisfied because they are independent of α. Because the inverse uncertainty distribution is increasing in α, we have

$$\Phi_i^{-1}(\alpha_2) \le \Phi_i^{-1}(\alpha_1),$$

$$\sum_{i=1}^{n} \Phi_i^{-1}(\alpha_2)(x_{ij}^1+y_{ij}^1) \le \sum_{i=1}^{n} \Phi_i^{-1}(\alpha_1)(x_{ij}^1+y_{ij}^1) \le c, \ \forall j = 1,2,...,m_1,$$

(5) is satisfied. Therefore, $\{x^1, y^1\}$ is a feasible solution for CC-UALBP1 (α_2) which output m_1 stations. Since m_1 is feasible, m_1 becomes an upper bound for the optimum m_2 of CC-UALBP1 (α_2), and $m_1 \geqslant m_2$. The corollary is proved. ■

4. THE BRANCH AND BOUND REMEMBER ALGORITHM

Sewell and Jacobson (2012) invented a branch and bound remember algorithm (BBR) which uses memory to eliminate dominated solutions, and BBR is the most efficient exact method for solving traditional line balancing problem. In this section, we propose an algorithm based on the framework of BBR to solve our problems.

Bounds

Lower and upper bounds are utilized to alleviate the complexity of the problem, i.e. reduce the number of iterations of the BBR. The lower bound tells us the best theoretical optimum we can achieve before a subproblem is solved. We modify the three lower bounds proposed in Sewell and Jacobson (2012) by considering the uncertain task time attribute. The first two lower bounds are as follows. Suppose we have a partial solution $O_k = (A, U, E_1, E_2, …, E_k)$, where E_j is the task assignment to station j. A is the set of tasks assigned to one of the k stations and U is the set of unassigned tasks. Then,

$$\text{LB1} = k + \left\lceil \frac{\sum_{i \in U} \Phi_i^{-1}(\alpha)}{c} \right\rceil,$$

$$\text{LB2} = k + \left| \{ i \mid \Phi_i^{-1}(\alpha) > c \}, i \in U \} \right| + \left\lceil \frac{\left| \{ i \mid \Phi_i^{-1}(\alpha) = c/2 \}, i \in U \} \right|}{2} \right\rceil.$$

$$A = \bigcup_{j=1}^{k} E_j$$

We assign weight w_i to each task so as to compute the third lower bounds (LB3).

$$w_i = \begin{cases} 1 & \text{if } \Phi_i^{-1}(\alpha) > 2c/3 \\ 2/3 & \text{if } \Phi_i^{-1}(\alpha) = 2c/3 \\ 1/2 & \text{if } c/3 < \Phi_i^{-1}(\alpha) < 2c/3 \\ 1/3 & \text{if } \Phi_i^{-1}(\alpha) = c/3 \end{cases}$$

$$\text{LB3} = k + \sum_{i \in U} w_i.$$

Therefore, the lower bound at station for partial solution $O_k = (A, U, E_1, E_2, …, E_k)$ is LB = max (LB1, LB2, LB3).

The upper bound reflects the current best outcome (m) as BBR is running. It can be obtained in branching process as mentioned below. If the lower bound of a subproblem is greater than or equal to the upper bound, that subproblem is fathomed meaning that there is no need to continue the branching process and, instead, we move back to the first station and branch an unexplored subproblem.

Branching

The branching process refers to finding a task sequence. It has two purposes in BBR: 1) find a feasible task assignment for a new station; 2) find an upper bound UB for the current subproblem. Assume there is a partial solution $O_k = (A, U, E_1, E_2, …, E_k)$. The subproblem for this current original problem is to assign the tasks in U in order to find the minimum number of stations. In the branching process, a task i is assigned to a station, and then the constraints are examined.

To ensure that we do not get any overlapping feasible task sequence, we should assign tasks in an ascending order of the task's number. As we know, the quality of a partial solution O_k has a negative relation with the cumulative idled time and a positive relation with the number of unassigned tasks. We end up with many feasible task assignments after the branching process is repeated. Therefore, we propose a criterion to select the partial solution to proceed. Set

$$b(O_k) = Id / k - 0.02\,|U|,$$

where

$$Id = kc - \sum_{i \in A} \Phi_i^{-1}(\alpha)$$

We always choose the partial solution with the smallest $b(O_k)$.

Remember

As BBR is ongoing, every subproblem is memorized (stored). Before branching on a subproblem, it checks whether the subproblem is dominated by a previous subproblem. A subproblem $O_k = (A, U, E_1, E_2, \ldots, E_k)$ is dominated by another subproblem $O_v = (A', U', E_1', E_2', \ldots, E_v')$ whenever $A' \subseteq A$ and $v \le k$. If so, then the best solution for O_v has no more stations than the best solution for O_k. Therefore, we sacrifice some computer memories to increase the computational speed. We could use a hash table to store subproblems in the program.

The BBR algorithm is presented by a flow chart below

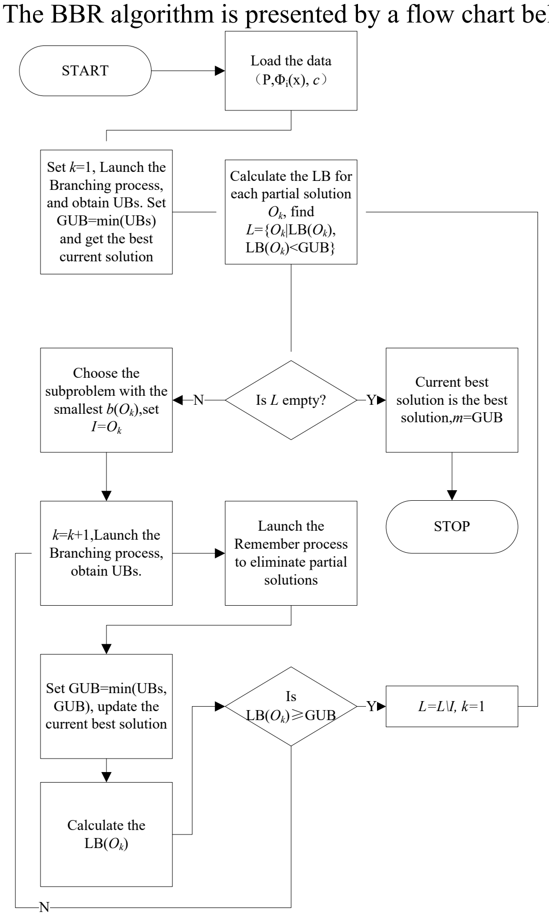

Figure 3: The flow chart of the proposed algorithm

5. NUMERICAL STUDIES

Consider an assembly line with 28 tasks. The precedence graph is shown in Figure 4 (it is an actual telephone assembly line (Scholl, 1993). Assume the task time of the 28 tasks are zigzag uncertain variables $t_i \sim Z(a_i, b_i, c_i)$ which are presented in Table 2. We let c = 135, α = 0.95. The optimum is 8, and the optimal task assignment is shown in Table 3.

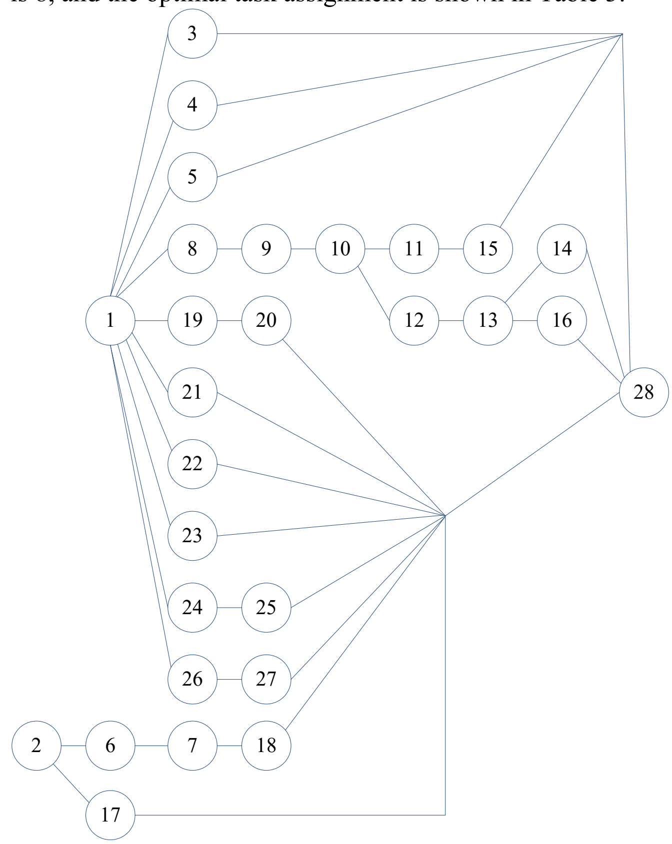

Figure 4 The precedence graph

Table 2 Dataset

Task #	Task time	Task #	Task time
1	Ƶ(60, 70, 80)	15	Ƶ(4, 5, 6)
2	Ƶ(42, 59, 75)	16	Ƶ(6, 8, 10)
3	Ƶ(25, 33, 36)	17	Ƶ(88, 97, 106)
4	Ƶ(5, 6, 7)	18	Ƶ(6, 8, 9)
5	Ƶ(0.7, 1, 1.5)	19	Ƶ(40, 47, 53)
6	Ƶ(22, 27, 34)	20	Ƶ(62, 67, 72)
7	Ƶ(14, 17, 20)	21	Ƶ(15, 17, 20)
8	Ƶ(50, 62, 70)	22	Ƶ(7, 8, 9)
9	Ƶ(25, 31, 37)	23	Ƶ(2, 3, 4)
10	Ƶ(50, 53, 55)	24	Ƶ(18, 21, 23)
11	Ƶ(17, 21, 25)	25	Ƶ(100, 107, 112)
12	Ƶ(16, 19, 24)	26	Ƶ(2.5, 3, 4)
13	Ƶ(94, 108, 115)	27	Ƶ(1, 2, 2.8)
14	Ƶ(45, 52, 60)	28	Ƶ(66, 72, 77)

Table 3 Optimal task assignments

	Station 1	Station 2	Station 3
Task assignment	(1,4,5,19,22,26)	(8,27,28)	(2,3,6,7)
	Station 6	Station 4	Station 5
Task assignment	(14,20)	(9,10,11,12,15,23)	(15,16,18)
	Station 7	Station 8	
Task assignment	(17,21,24)	(25)	

Further, we test the α in the range of [0.1, 0.9] with the step size equal to 0.1. The results are presented in Table 4. The results are consistent with corollary 3.1 that m is nondecreasing in α. Therefore, with other input being held constant, more stations lead to more belief reliabilities.

Table 4 Optimum under different α

α	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
m	7	7	8	8	8	8	9	9	9

We now turn our attention to the impact of task time variance on m. We increase the variance of the uncertain task time but maintain its expected value. We change the parameters of the zigzag distribution as follows.

$$a_i^{per} = a_i^0 \times (1 - per),\ c_i^{per} = c_i^0 \times (1 + per),$$

$$b_i^{per} = \frac{a_i^0 + 2b_i^0 + c_i^0 - a_i^{per} - c_i^{per}}{2},$$

where per indicates the degree of changes against the original zigzag uncertain distribution. The more per value is, the higher variance the uncertain task time has. We select 5 particular per values, 0, 0.05, 0.1, 0.15 and 0.2, respectively. The results are shown in Table 5. As can be seen, the number of stations increases in task time variance as the expected value of task time remains the same. In another word, as the expected values of task times are the same, we need to open more stations to satisfy the increasing volatility of task times.

Table 5 Optimum under different levels of variance

per	0	0.05	0.1	0.15	0.2
m	8	9	10	11	11

6. CONCLUSIONS

We address the Type-1 U-shaped Assembly Line Balancing with uncertain task time in this paper. When the historical task time information is not available, the probability theory is not suitable to estimate the uncertain task time. Alternatively, uncertainty theory which is used to measure the belief degree can be utilized. In this paper, we employ the uncertainty theory to evaluate the uncertain task time. An uncertain programming model is proposed (CC-UALBP1) in order to minimize the number of stations considering the uncertain task attribute. Further, we transform the uncertain models into deterministic models to reduce the computational effort. We develop an algorithm based on branch and bound remember algorithm to find the optimal solutions. We apply the proposed models to describe a 28 tasks assembly line with uncertain task time attribute. In the future, our model and algorithm can be extended to other assembly line problems (mixed-models, parallel stations, etc.) in assembly line balancing literature.

REFERENCES

Battaïa, O., & Dolgui, A. A taxonomy of line balancing problems and their solution approaches. International Journal of Production Economics,142(2) (2013), 259-277.

Bryton, B. Balancing of a Continuous Production Line. M.S. Thesis, Northwestern University, 1954

Charnes, A., &Cooper,W. Management models and industrial applications of linear programming.New York: Wiley, 1961

Celik, E., Kara, Y., and Atasagun, Y. "A new approach for rebalancing of U-lines with stochastic task times using ant colony optimisation algorithm." International Journal of Production Research 52.24(2014): 7262-7275

Delice, Y., Kızılkaya Aydoğan, E., & Özcan, U. Stochastic two-sided U-type assembly line balancing: a genetic algorithm approach. International Journal of Production Research, 54(11)(2016), 3429-3451.

Dong, J. , Zhang, L. , Xiao, T. , & Mao, H. "Balancing and sequencing of stochastic mixed-model assembly U-lines to minimise the expectation of work overload time." International Journal of Production Research 52.24(2014):7529-7548.

Gao, Y., & Qin, Z. A chance constrained programming approach for uncertain p-hub center location problem. Computers & Industrial Engineering, 102 (2016), 10-20.

Ke, H., Liu, H., & Tian, G. An uncertain random programming model for project scheduling problem. International Journal of Intelligent Systems, 30(1) (2015), 66-79.

Li, Y., & Boucher, T. O. Assembly line balancing problem with task learning and dynamic task reassignment. International Journal of Advanced Manufacturing Technology, 88(9-12) (2016), 1-9.

Li, R., & Liu, G. An uncertain goal programming model for machine scheduling problem. Journal of Intelligent Manufacturing, 28(3) (2017), 689-694.

Liu, B. Uncertainty Theory, Berlin: Springer, 2007

Liu, B. Theory and practice of uncertain programming (2nd ed). Berlin: Springer, 2009.

Liu, B. Uncertainty theory: a branch of mathematics for modeling human uncertainty. Berlin: Springer, 2010.

Liu, B. Why is there a need for uncertainty theory? Journal of Uncertain Systems, 6(1) (2012), 3–10.

Monden, Y., Toyota Productiont Systeml, (2nd Ed.), Industrial Engineering and Management Press, Institute of Industrial Engineers, Norcross, GA, 1993.

Salveson, M.E. The assembly line balancing problem. Journal of Industrial Engineering, 6.3 (1955), 18-25.

Scholl,A. data of assembly line balancing problem, http://alb.mansci.de/index.php?content=classview&content=classview&content2=classview&content3=classviewdlfree&content4=classview&classID=44&type=dl, 1993

Sewell, E. C., & Jacobson, S. H. A branch, bound, and remember algorithm for the simple assembly line balancing problem. INFORMS Journal on Computing, 24(3) (2012), 433-442

Tiacci, Lorenzo. "Mixed-model U-shaped assembly lines: Balancing and comparing with straight lines with buffers and parallel workstations." Journal of Manufacturing Systems (2017):S0278612517301073.

Wen, M., Qin, Z., & Kang, R. The alpha-cost minimization model for capacitated facility location-allocation problem with uncertain demands. Fuzzy Optimization and Decision Making, 13(3) (2014), 345-356.