

Chapter 37

Improved Heuristic Procedure for Mixed-Model U-line Balancing Problem with Fuzzy Times

Zeqiang Zhang and Wenming Cheng

Abstract Facing the increasingly competitive market environment, variety of customer needs, and rapidly changing market, more and more assembly lines are switching to U-line and mixed model produced due to the use of just-in-time production principles. Considering that some uncertainties are existing in practical problem, this paper focuses on the mixed-model U-line balancing problem (MMULBP) with processing time and cycle time as fuzzy numbers. An improved heuristic procedure is proposed to solve the problem. The procedures are based on the traditional ranked positional weight method, and some improvements are made on fuzzy number operation and two-direction search for U-line layout, in order to solve the complex practical problem. The results of an experimental study indicate that the proposed procedure is effective.

Keywords Mixed-model U-line balancing problem · Position weight method · Heuristic · Fuzzy

37.1 Introduction

Facing the increasingly competitive market environment, variety of customer needs, and rapidly changing market, the traditional straight assembly line cannot meet the demands of the modern society. More and more assembly lines are switching to

Z. Zhang (✉) · W. Cheng
School of Mechanical Engineering, Southwest Jiaotong University,
Chengdu 610031, China
e-mail: zhangzeqiang@gmail.com

W. Cheng
e-mail: wmcheng@swjtu.cn

U-line and mixed model produced due to the use of just-in-time production principles [1, 2].

For the importance of the simple assembly line balancing problem (ALBP), there are numerous researches on ALBP [3–5], but only few literatures focus on mixed-model U-line balancing problem (MMULBP). Sparling and Miltenburg [6] first presented MMULBP and the formulation for the problem, and an approximate solution algorithm for the MMULBP is also proposed. And then, some researches about exact algorithms are also conducted [7, 8], such as goal programming [9], mixed integer linear programming [10], and branch-and-bound algorithms [11]. The simple ALBP falls into the NP hard class of combinatorial optimization problems, and MMULBP is more complex for with larger search space. So, heuristics is an effective way to solve the problem [12].

There are numerous literatures for ALBP, and most of them assume constant and deterministic operation times, but only few studies are reported for solving the fuzzy line balancing problem. Nguyen Van [13] developed a heuristic solution for fuzzy mixed-model line balancing problem. Fonseca et al. [14] proposed modified COMSOAL and ranked positional weighting technique to solve the straight line balancing problem with a fuzzy representation of the time variables. Zhang et al. [15] proposed an effective heuristic for solving type 1 of the fuzzy ULBP. Tsujimura et al. [16] presented a genetic algorithm to solve the traditional straight ALBP with fuzzy operation time. Zacharia and Nearchou [17] presented a new multi-objective genetic algorithm for solving the fuzzy extension of the simple ALBP of type 2.

This paper presents a heuristic for the type 1 mixed-model U-shaped line balancing problem with fuzzy processing times. The rest of the paper is as follows. In Sect. 37.2, the fuzzy MMULBP-1 is discussed and mathematical model is presented. Next, Sect. 37.3 will develop the heuristic solution for fuzzy MMULBP-1. Then, an example problem is solved using the proposed method in Sect. 37.4. Finally, some conclusions are presented in Sect. 37.5.

37.2 Problem Statement and Formulation

37.2.1 Problem Statement

In order to reduce the production cost and provide customers with a variety of products in a timely manner, different products or models should be produced on the same line. Mixed-model assembly line balancing problem (MMALBP) can be described as follows: Given P models, the set of tasks associated with each model, the processing time of each task, and the set of precedence relations of tasks for each model, the problem is to assign the tasks to an ordered sequence of stations such that some constraints are satisfied and some performance measures are

optimized. The mixed-model U-shaped line balancing problem is an extension of MMALBP with respect to U-shaped line instead of straight line. Three versions of the problem can be divided as follows:

- (I) MMULBP-1 consists of assigning tasks to work stations such that the number of stations is minimized for a given cycle time.
- (II) MMULBP-2 is to minimize the cycle time for a given number of stations.
- (III) MMULBP-E is obtained by maximizing the line efficiency for varying production rates and numbers of stations.

37.2.2 Assumptions

1. Task processing time of each model is given and is a fuzzy number.
2. Precedence diagrams for each model are given and can be accumulated into a single combined precedence diagram.
3. WIP inventory buffer is not allowed between workstation.
4. Parallel workstation is not allowed.
5. Common tasks exist between models that must be assigned to the same workstation.

37.2.3 Combined Precedence Diagram

Different models in mixed-model assembly line usually exists some similar tasks; precedence relations of same tasks in different models are usually similar. According to this character, transform different models into an equivalent single model by taking the union of the nodes and the precedence relations of the diagrams of all the models.

Let G_p be the precedence diagram of model p . G_p has a set of nodes $N(p)$ and of arcs $L(p)$ where

$$N(p) = \{n(p)_1, n(p)_2, \dots\} \quad (37.1)$$

$$L(p) = \{l(p)_1, l(p)_2, \dots\} \quad (37.2)$$

The nodes and arcs represent tasks and precedence relations, respectively. If an arc $l(p)_h$ has initial node $n(p)_i$ and terminal node $n(p)_j$, then the task represented by $n(p)_i$ must be completed before the task $n(p)_j$.

Precedence relations for a set of models $\hat{P} = \{1, 2, \dots, P\}$ are represented by directed graphs G_1, G_2, \dots, G_P . Then, we can define a single graph $G_{\hat{P}}$ with nodes $N(\hat{P})$ and arcs $L(\hat{P})$ where

$$N(\hat{P}) = \{N(1) \cup N(2) \cup \dots \cup N(P)\} \tag{37.3}$$

$$L(\hat{P}) = \{L(1) \cup L(2) \cup \dots \cup L(P)\} \tag{37.4}$$

For different models, a task may be included in some models and not included in others, so define δ_{ip} as follows:

$$\delta_{ip} = \begin{cases} 1 & \text{if task } n(p)_i \text{ is part of model } p \\ 0 & \text{otherwise} \end{cases} \tag{37.5}$$

Then, the weighted average processing time of task i (\tilde{t}_i) in the planning horizon (\tilde{T}) is computed as follows:

$$\tilde{t}_i = \frac{\sum_{p=1}^P Q_p \delta_{ip} \tilde{t}_{ip}}{\sum_{p=1}^P Q_p} \tag{37.6}$$

where \tilde{t}_{ip} denotes the fuzzy processing time of task i for model p , $i = 1, \dots, N$, $p = 1, \dots, P$, where N and P are the total number of tasks and models in the problem, respectively. Q_p is the production quantity of model p in the planning horizon.

The cycle time is calculated as follows:

$$\tilde{C} = \frac{\tilde{T}}{\sum_{p=1}^P Q_p} \tag{37.7}$$

So if we use \tilde{t}_i as the processing of task i in combined precedence diagram, then we can use the method of single-model assembly line balancing to solve the problem.

37.2.4 Constraints

1. All tasks must be assigned to one workstation.
2. Each task is assigned only once, i.e., a task cannot be split among two or more stations.
3. The work content in any workstation for any model cannot exceed the cycle time of that model.
4. The precedence constraints are not violated on the U-line.

37.2.5 Problem Formulation

Based on the above assumptions, the MMULBP with fuzzy times of type 1 (MMULBP-1) is formulated as follows:

$$\min \sum_{p=1}^P \sum_{k=1}^K \left(\tilde{C}_p - \sum_{i=1}^N v_{ipk} \tilde{t}_{ip} \right) \tag{37.8}$$

subject to

$$\bigcup_{k=1}^K S_k = E \tag{37.9}$$

$$S_i \cap S_j = \phi \quad i, j = 1, \dots, K \quad i \neq j \tag{37.10}$$

$$\sum_{i=1}^N v_{ipk} \tilde{t}_{ip} \leq \tilde{C}_p \quad p = 1, 2, \dots, P \quad k = 1, 2, \dots, K \tag{37.11}$$

$$\begin{aligned} \forall x \in S_i, y \in S_j, \quad & \text{if } p_{xy} = 1, \text{ then } i \leq j; \text{ or} \\ \forall y \in S_j, z \in S_k, \quad & \text{if } p_{yz} = 1, \text{ then } k \leq j. \end{aligned} \tag{37.12}$$

where

E the set of tasks in the line, $E = \{1, 2, \dots, N\}$.

K the number of workstations.

S_k the set of tasks assigned to workstation k , $S_k = \{i | \text{task } i \text{ is assigned to workstation } k\}$.

\tilde{C}_p cycle time of model p .

$$v_{ipk} = \begin{cases} 1 & \text{if task } i \text{ of model } p \text{ is assigned to station } k \\ 0 & \text{otherwise} \end{cases} \tag{37.13}$$

$$p_{ij} = \begin{cases} 1 & \text{if task } i \text{ is performed before task } j \\ 0 & \text{otherwise} \end{cases} \tag{37.14}$$

37.2.6 The Character of U-Shaped Layout

U-shaped layouts with the straight line are mainly distinct in that the entrance and exit are located in the same direction. So we can add or remove workers in assembly line according to the demands. Taking the following problem as example [8], the problem has seven tasks and the precedence diagram is given in Fig. 37.1a. The numbers within and above the nodes represent tasks and the associated task

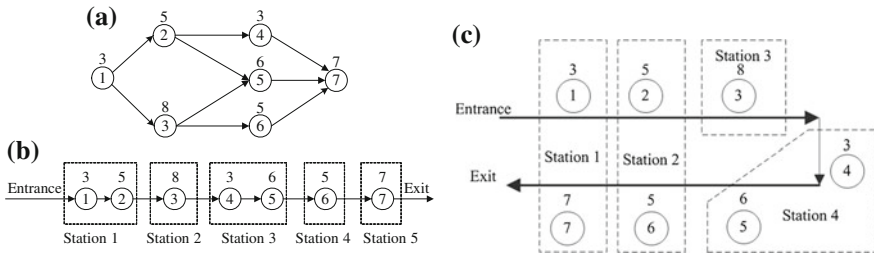


Fig. 37.1 Precedence diagram (a), traditional (b) and U-shaped (c) line balancing results

times, respectively. The directed arrow connecting the nodes specifies the precedence relations. When assuming a cycle time of 10, it can be seen that all tasks are performed at five workstations (shown in Fig. 37.1b, the idle time is 13 and the efficiency is 74 %) in traditional assembly line, whereas all tasks are performed at four workstations (shown in Fig. 37.1c, the idle time is 3 and the efficiency is 92.5 %) in U-shaped assembly line. We can see that U-shaped layout can eliminate excess idle time and achieve more balancing situation in this example. Taking tasks 1 and 7 in Fig. 37.1c as example, although tasks 1 and 7 are located at the beginning and end of the line, respectively, they have been processed by the same workstation (workstation 1). This means that the U-shaped layout has great assignment flexibility and balancing efficiency.

According to the character of U-shaped assembly line, the main difference in assigning the tasks compared to the traditional line is as follows: Straight line only allows assigning tasks according to the arcs direction in the precedence diagram. However, U-shaped line permits assigning tasks according to the arcs direction, or reverse direction, or both arcs direction and reverse direction. So, the U-shaped line balancing problem is much more complicated than the traditional line balancing problem due to the increased search space.

37.3 The Heuristic Procedure for MMULBP

For the content about arithmetics and ranking fuzzy numbers, the readers can refer to the paper [18].

37.3.1 Nomenclature

Notation for fuzzy RPWT procedure is given below:

- $\tilde{t}(i)$ fuzzy processing time of task $i(i = 1, \dots, n)$, defined as a triplet (a_1, a_2, a_3)
- \tilde{W}_i fuzzy ranked positional weight of task i

- $\tilde{t}(S_k)$ fuzzy time required to complete all tasks assigned to workstation S_k ($k = 1, \dots, m$)
- \tilde{c} fuzzy cycle time after assignments
- \tilde{I}_k fuzzy idle time for workstation S_k , ($k = 1, \dots, m$)
- \tilde{t}_{sum} total processing time in all workstations
- \tilde{C}_{max} permitted fuzzy cycle time
- $\tilde{\eta}$ fuzzy balance efficiency

According to the characters of the problem, we can have the following result:

$$\tilde{c} = \max\{\tilde{t}(S_k)\}. \tag{37.15}$$

$$\tilde{t}(S_k) = \sum_{j \in S_k} \tilde{t}(j), \quad k = 1, \dots, m. \tag{37.16}$$

$$\tilde{I}_k = \tilde{C}_{\text{max}} - \tilde{t}(S_k). \tag{37.17}$$

$$\tilde{t}_{\text{sum}} = \sum_{k=1}^m \tilde{t}(S_k), \quad k = 1, \dots, m. \tag{37.18}$$

$$\tilde{\eta} = \frac{\tilde{t}_{\text{sum}}}{m \times \tilde{c}}. \tag{37.19}$$

37.3.2 Fuzzification of RPWT for MMULBP

The proposed procedure for fuzzy MMULBP-1 is based on the RPWT method, and some modifications should be made. We should translate the MMULBP-1 to ULBP-1 based on the methods given in Sect. 37.2.3. Then, we can use the methods for single model to solve the problem. And the fuzzy positional weight for each task, \tilde{W}_i , can be calculated by summing the processing time of the current task with the processing times of all its successors or predecessors.

$$\tilde{W}_i = \max \left\{ \tilde{t}(k) + \sum_{i \in U_s(k)} \tilde{t}(i), \tilde{t}(k) + \sum_{j \in U_p(k)} \tilde{t}(j) \right\}. \tag{37.20}$$

where $U_p(k)$ [or $U_s(k)$] be the set of tasks which must precede (or succeed) task k , respectively.

The fuzzy operator will be utilized in this calculation for the processing times is fuzzy numbers. And the mean comparison method is used for the comparison [19].

The ranking of tasks based on their positional weights is also very important for the RPWT method in which the tasks are ranked in a decreasing order. And the average height method [18] is used for ranking of triangular fuzzy numbers. Compute the value H for tasks \tilde{A} and \tilde{B} where $H(\tilde{A}) = (a_1 + a_2 + a_3)/3$ and $H(\tilde{B}) = (b_1 + b_2 + b_3)/3$; if $H(\tilde{A}) > H(\tilde{B})$, then $\tilde{A} > \tilde{B}$.

Whenever a new task is added to a workstation, fuzzy processing times are accumulated using the fuzzy addition operator. And the mean comparison method is used to check whether the station time violates the permitted cycle time or not. If $\tilde{t}(S_k) + \tilde{t}(i) \leq \tilde{C}_{\max}$, then task i can be added to the current workstation. If $\tilde{t}(S_k) + \tilde{t}(i) > \tilde{C}_{\max}$ and task i cannot be assigned to the current workstation and a new workstation must be opened to accommodate the task.

37.3.3 Steps involved in the Procedure

The procedure of the proposed heuristic for MMULBP-1 is presented as follows.

- Step 1: Convert the mixed-model U-line balancing problem into the single-model U-line balancing problem (including the processing time, cycle time, and precedence diagrams) according to Sect. 37.2.3;
- Step 2: Based on the combined precedence diagram, establish each task's processing time given the cycle time as TFNs;
- Step 3: Establish combined precedence relationships for the problem;
- Step 4: Aggregate fuzzy cycle time and processing time;
- Step 5: Compute the positional weight for each task in the new combined precedence diagram according to Eq. (37.20);
- Step 6: Rank positional weight into List A, and an initial task assignment sequence is obtained. We use the average height method to rank the tasks in a descending order;
- Step 7: Assign aggregated tasks in List A to workstation 1 based on the precedence relations and their positional weight rank, and the fuzzy process times will be accumulated;
- Step 8: Continue to assign the next succeeding ranked task to the workstation, as long as the task fits the precedence relationships and the station times do not exceed the cycle time;
- Step 9: If the station times exceed cycle time with the addition of new task i , then create a new workstation;
- Step 10: Repeat steps 8–9 until all tasks are assigned;

Step 11: Determine the number of workstations and calculate idle time for each station after final assignment according to Eq. (37.17); compute fuzzy balance efficiency according to Eq. (37.19).

37.4 Numerical Example and Experiment

A numerical example is used to illustrate the proposed heuristic procedure. Three models, A, B, and C, with demand of $Q_1 = 400$, $Q_2 = 200$, and $Q_3 = 200$, respectively, each day, are simultaneously assembled in a line. For one day, the production time is fuzzy number $\tilde{T} = (480\ 493\ 507)$ min. There are 20 tasks in the production, and the precedence diagram for each model is given in Fig. 37.2. The task processing times for the three models are shown in Table 37.1.

According to Sect. 1.3, we can obtain the combined precedence diagram as shown in Fig. 37.3. By using Eq. (37.6), the weighted average processing time for each task i can be calculated; refer to the column \hat{t}_i in Table 37.2. And we also can calculate the fuzzy cycle time, $\tilde{C}_{max} = (36, 37, 38)$, according to the Eq. (37.7).

The fuzzy positional weight for the problem can be calculated by Eq. (37.20). The average height of the triangular fuzzy number also can be obtained according to the average height method, and the results are shown in Table 37.2.

Based on the above data, the solution can be found by using the algorithm proposed in Sect. 37.3.3, and the best solution is obtained for 6 workstations. The proposed procedure has the advantages of quick search by using positional weight ranking. For the detailed results, we can refer the readers to Table 37.3. After calculations, the practical fuzzy cycle is $\tilde{c} = (33, 36, 39)$, the total processing time in all workstations is $\tilde{t}_{sum} = (184.25, 199, 213.75)$ by Eq. (37.18), and the line efficiency is $(0.787, 0.921, 1)$ by Eq. (37.19). And we can see that this assembly line with high production rate can meet the demands of the production.

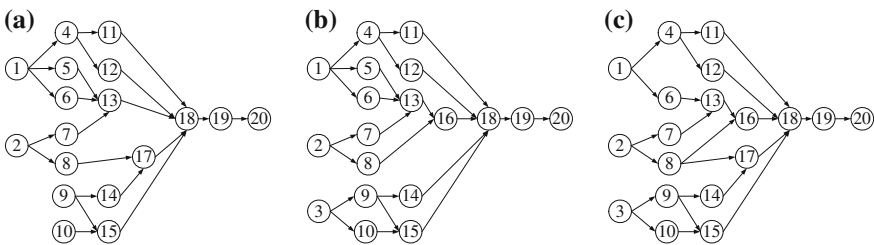


Fig. 37.2 Precedence diagram for models A (a), B (b), and C (c)

Table 37.1 Fuzzy processing time of every task in each model

Task no.	Processing time (s)			Task no.	Processing time (s)		
	\tilde{t}_{i1}	\tilde{t}_{i2}	\tilde{t}_{i3}		\tilde{t}_{i1}	\tilde{t}_{i2}	\tilde{t}_{i3}
1	(15 16 17)	(15 16 17)	(15 16 17)	11	(15 16 17)	(15 16 17)	(15 16 17)
2	(6.5 7 7.5)	(6.5 7 7.5)	(6.5 7 7.5)	12	(8.5 9 9.5)	(8.5 9 9.5)	(8.5 9 9.5)
3	(0 0 0)	(11 12 13)	(11 12 13)	13	(12 13 14)	(12 13 14)	(12 13 14)
4	(8.5 9 9.5)	(8.5 9 9.5)	(8.5 9 9.5)	14	(11 12 13)	(11 12 13)	(11 12 13)
5	(11 12 13)	(11 12 13)	(0 0 0)	15	(14 15 16)	(14 15 16)	(14 15 16)
6	(14 15 16)	(14 15 16)	(14 15 16)	16	(0 0 0)	(9 10 11)	(9 10 11)
7	(5.5 6 6.5)	(5.5 6 6.5)	(5.5 6 6.5)	17	(8.5 9 9.5)	(0 0 0)	(9 10 11)
8	(13 14 15)	(13 14 15)	(13 14 15)	18	(10 11 12)	(10 11 12)	(10 11 12)
9	(5.5 6 6.5)	(5.5 6 6.5)	(5.5 6 6.5)	19	(3.5 4 4.5)	(3.5 4 4.5)	(3.5 4 4.5)
10	(4.5 5 5.5)	(4.5 5 5.5)	(4.5 5 5.5)	20	(13 14 15)	(13 14 15)	(13 14 15)

Fig. 37.3 Combined precedence diagram for three models

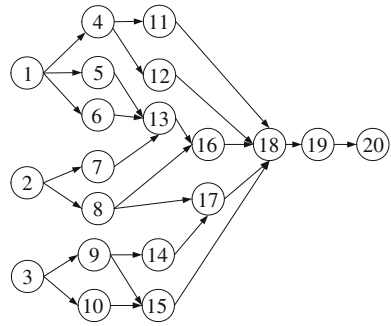


Table 37.2 Positional weight for the problem

Task no.	\tilde{W}_i	$H(\tilde{W}_i)$	Task no.	\tilde{W}_i	$H(\tilde{W}_i)$
1	(112.25 121 129.75)	121	11	(41.5 45 48.5)	45
2	(74.5 81 87.5)	81	12	(35 38 41)	38
3	(73.5 80 86.5)	80	13	(49.25 53 56.75)	53
4	(58.5 63 67.5)	63	14	(44 48 52)	48
5	(51.25 56 60.75)	56	15	(40.5 44 47.5)	44
6	(57 62 67)	62	16	(66.75 72 77.25)	72
7	(48.5 53 57.5)	53	17	(48 52 56)	52
8	(50.5 55 59.5)	55	18	(167.75 181 194.25)	181
9	(63.5 69 74.5)	69	19	(171.25 185 198.75)	185
10	(45 49 53)	49	20	(184.25 199 213.75)	199

Table 37.3 Results for the problem

Station no.	Assigned tasks	$\tilde{t}(S_k)$	\tilde{I}_k
1	20, 19, 18, 2	(33 36 39)	(0 1 5)
2	1, 3, 16, 9	(30.5 33 35.5)	(0.5 4 7.5)
3	6, 5, 4	(30.75 33 35.25)	(0.75 4 7.25)
4	8, 7, 13	(30.5 33 35.5)	(0.5 4 7.5)
5	17, 10, 14, 12	(30.5 33 35.5)	(0.5 4 7.5)
6	11, 15	(29 31 33)	(3 6 9)
Line efficiency = (0.787, 0.921, 1)			

37.5 Conclusion

In this paper, an improved heuristic for MMULBP with fuzzy times is constructed. Computational experiments demonstrated the validity of the proposed procedure. In future studies, we hope to apply the improved heuristic method to extensions of the

MMULBP, such as fuzzy MMULBP with multiple objectives. The integration of intelligent methods, for example, heuristic method and ACO algorithms, to MMULBP is also an interesting topic in the future.

Acknowledgement This work was partially supported by the National Natural Science Foundation of China (No. 51205328), the Youth Foundation for Humanities and Social Sciences of Ministry of Education of China (No. 12YJCZH296), the PhD Programs Foundation of Ministry of Education of China (No. 200806131014), and the Research Center of Sichuan Province for Circular Economy (No. XHJJ-1205).

References

1. Cheng CH et al (2000) The effect of straight- and U-shaped lines on quality. *IEEE Trans Eng Manage* 47:321–334
2. Aase GR et al (2004) U-shaped assembly line layouts and their impact on labor productivity: an experimental study. *Eur J Oper Res* 156:698–711
3. Scholl A, Becker C (2006) State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *Eur J Oper Res* 168:666–693
4. Peeters M, Degraeve Z (2006) An linear programming based lower bound for the simple assembly line balancing problem. *Eur J Oper Res* 168:716–731
5. Zhang Z-Q et al (2007) Improved ant colony optimization for assembly line balancing problem. *Comput Integr Manuf Syst CIMS* 13:1632–1638
6. Sparling D, Miltenburg J (1998) The mixed-model U-line balancing problem. *Int J Prod Res* 36:485–501
7. Aase GR et al (2003) U-OPT: an analysis of exact U-shaped line balancing procedures. *Int J Prod Res* 41:4185–4210
8. Gokcen H et al (2005) A shortest route formulation of simple U-type assembly line balancing problem. *Appl Math Model* 29:373–380
9. Gokcen H, Agpak K (2006) A goal programming approach to simple U-line balancing problem. *Eur J Oper Res* 171:577–585
10. Kara Y, Tekin M (2009) A mixed integer linear programming formulation for optimal balancing of mixed-model U-lines. *Int J Prod Res* 47:4201–4233
11. Scholl A, Klein R (1999) ULINO: optimally balancing U-shaped JIT assembly lines. *Int J Prod Res* 37:721–736
12. Chiang W-C, Urban TL (2006) The stochastic U-line balancing problem: a heuristic procedure. *Eur J Oper Res* 175:1767–1781
13. Nguyen Van H (2006) A heuristic solution for fuzzy mixed-model line balancing problem. *Eur J Oper Res* 168:798–810
14. Fonseca DJ et al (2005) A fuzzy logic approach to assembly line balancing. *Mathware Soft Comput* 12:57–74
15. Zhang Z et al (2009) A heuristic approach for fuzzy U-shaped line balancing problem. In: 6th international conference on fuzzy systems and knowledge discovery, Tianjin, pp 228–232
16. Tsujimura Y et al (1995) Solving fuzzy assembly-line balancing problem with genetic algorithms. *Comput Ind Eng* 29:543–547
17. Zacharia PT, Nearchou AC (2012) Multi-objective fuzzy assembly line balancing using genetic algorithms. *J Intell Manuf* 23:615–627
18. Hong TP, Chuang TN (1999) A new triangular fuzzy Johnson algorithm. *Comput Ind Eng* 36:179–200
19. Gen M et al (1996) Fuzzy assembly line balancing using genetic algorithms. *Comput Ind Eng* 31:631–634