Modeling and Optimization of Parallel Disassembly Line Balancing Problem With Parallel Workstations

Wei Liang, Zeqiang Zhang, Yanqing Zeng, Tao Yin, and Tengfei Wu

***Abstract*—To reasonably arrange disassembly facilities and plan enterprise space, we propose a parallel disassembly line balancing problem (PW-PDLBP) with parallel workstations. Additionally, a mixed-integer nonlinear programming (MINLP) model that minimizes the line length, number of workstations, idle time balancing index, and energy consumption is established based on the problem characteristics and is solved using the GUROBI optimizer. Furthermore, a multiobjective enhanced differential evolution algorithm (MEDE) is developed to obtain high-quality disassembly schemes for PW-PDLBP. The correctness of encoding and decoding and the solving performance of MEDE are verified by comparing with the MINLP model and four existing algorithms. Then, an instance consisting of two different types of end-of-life TVs is optimized. Finally, the effectiveness of PW-PDLBP in improving enterprise space utilization is validated by comparing it with the parallel line layout without parallel workstations.**

***Index Terms*—Mixed-integer nonlinear programming (MINLP), multiobjective enhanced differential evolution algorithm (MEDE), parallel disassembly line balancing, parallel workstations.**

I. Introduction

With the development of industry, the standardization of factories has increased dramatically. The disassembly mode of end-of-life (EOL) products has also been converted from manual disassembly to the standardized flow line process. Each production line needs to follow the "one stream" standard in the standardized production workshops. Each disassembly task has to be assigned appropriately to avoid product blockage on the line. Additionally, disassembly is also an essential means to protect the environment and enable resource recycling and reuse. Hence, the disassembly line balancing problem (DLBP) [1] is vital in recycling.

In current enterprises that recycle EOL products, two-sided disassembly lines are more suitable for disassembling large and symmetrical EOL products (e.g., automobiles). However, for nonlarge EOL products, such as computers, TVs, and printers, parallel layouts are more favored by companies [2]. In the original parallel layout, all workstations are sequentially arranged in series, which can waste space in the production workshop. Because high profits are what enterprises pursue, they need to utilize fewer workstations to reduce costs. In addition, it is also interesting to study the shortening of disassembly line length in the case of lesser workstations. A shorter line length can effectively improve space utilization and reduce costs. Moreover, additional production facilities can be deployed in an identical shop. Therefore, this article investigates the parallel layout with parallel workstations in order to achieve a reasonable utilization of workshop space. The parallel layout with parallel workstations is described in Section III. Notably, employing parallel workstations is an effective means to reduce line length. Furthermore, the idle time balancing index [3] is also necessary to achieve production line balancing and can effectively balance the work intensity of each workstation. Additionally, it is necessary to reduce energy consumption under the policy of green production.

Because the DLBP is an NP-hard problem [4], exact methods cannot obtain an effective disassembly scheme for large-scale instances in a limited time. However, one of the advantages of intelligent algorithms is the ability to obtain feasible solutions quickly. For example, the differential evolutionary algorithm (DE) [5], as a novel evolutionary algorithm, has solved many engineering optimization problems such as facility layout [6], drones [7], and job shop scheduling problems [8] since it was proposed. Therefore, we present a multiobjective enhanced differential evolutionary algorithm (MEDE) adapted to the proposed problem. In addition, a mixed-integer nonlinear programming (MINLP) model is developed to validate the correctness of the designed encoding and decoding, ensuring that the proposed problem can be solved effectively.

The rest of this article is organized as follows. In Section II, DLBP is reviewed from three aspects. Section III provides a description of the parallel DLBP with parallel workstations (PW-PDLBP). Additionally, a MINLP model is constructed. In

Manuscript received 15 November 2022; revised 28 December 2022; accepted 26 January 2023. Date of publication 1 February 2023; date of current version 19 September 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 51205328 and Grant 51675450, in part by the Youth Foundation for Humanities, Social Sciences of Ministry of Education of China under Grant 18YJC630255, and in part by the Sichuan Science and Technology Program under Grant 2022YFG0245 and Grant 2022YFG0241. Paper no. TII-22-4697. *(Corresponding author: Zeqiang Zhang.)*

The authors are with the Technology and Equipment of Rail Transit Operation and Maintenance Key Laboratory of Sichuan Province and School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China (e-mail: liangwei@my.swjtu.edu.cn; zhangzq@home.swjtu.edu.cn; zyq2017200281@163.com; amniyim@my.swjtu.edu.cn; wtenfi@gmail.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TII.2023.3241583.

Digital Object Identifier 10.1109/TII.2023.3241583

Section IV, the MEDE is presented. In Section V, we solve and validate the MINLP model and MEDE. In Section VI, we optimize and analyze an instance containing two different types of EOL TV sets. Finally, Section VII concludes this article.

The contributions and differences of this study are as follows:

a) The PW-PDLBP is proposed to improve space utilization in factories and is compared with the parallel line layout without parallel workstations.
b) A MINLP model of PW-PDLBP is construed and solved using the GUROBI optimizer.
c) The MEDE is developed to optimize PW-PDLBP. Additionally, the encoding and decoding of PW-PDLBP are designed.

II. Related Studies

Scholars have studied DLBP in multiple aspects. In this section, we summarize the related studies from three aspects: layout, disassembly time, and solution method.

A. Layouts

The disassembly layouts studied in the existing literature include four primary forms: linear, two-sided, U-shaped, and parallel. The linear disassembly line is a layout that is the most frequent and earliest studied form. Liang et al. [9] studied the linear layout considering the tool switch. He et al. [10] studied the linear layout with uncertainty in the number of parts. In addition, the two-sided layout is superior when disassembling large and symmetrical EOF products. As the two-sided disassembly line is functioned by paired workstations on both sides, symmetrical tasks can be performed collaboratively by the paired workstations, which is a unique advantage. Therefore, Liang et al. [11] modeled and optimized two-sided layouts by considering different conditions, such as destructive mode, partial disassembly, and energy consumption. Unlike linear and two-sided layouts, the tasks at the inlet and outlet of the U-shaped layout can be disassembled simultaneously in a workstation. Therefore, the U-shaped layout is a decent option in specific environments. Agrawal and Tiwari [12] studied both balancing and sequencing problems of U-shaped disassembly lines. Li and Janardhanan [13] studied profit-oriented partial disassembly and consideration of AND/OR relationship for U-shaped DLBP, respectively. The parallel layout is an upgrade from the linear layout. However, unlike a linear layout, workstations of the parallel line layout [14] can be responsible for the tasks of two lines on both sides. Moreover, parallel disassembly line has been applied extensively in resource recycling enterprises. However, the research on parallel DLBP (PDLBP) among existing studies on DLBP is scarce. Therefore, it is meaningful to promote the development of parallel layout in DLBP and its further application in enterprises.

B. Disassembly Time

In existing studies on DLBP, disassembly time is mainly classified into certain task time [15] and uncertain task time [16]. Certain task time means that the specific disassembly time of each task is known before the instances are solved and have been studied extensively. McGovern and Gupta [17] studied DLBP with certain task time and optimized five objectives: the number of workstations, idle time balancing index, demand index, hazard index, and the number of direction changes. Kucukkoc [15] also performed a weighted single objective solution of two-sided DLBP with a certain task time considering AND/OR relationship. Zhang et al. [18] and Kalayci et al. [19] studied DLBP with task time as a triangular fuzzy number. Wang et al. [16] and Agrawal and Tiwari [12] studied DLBP with task time following the normal distribution. Although the DLBP with uncertain task time simulates the recycling state of products better, the disassembly schemes obtained with uncertain time are not applicable. The disassembly line will be blocked if the actual processing time of the tasks exceeds the formulated time in the disassembly schemes. As a result, disassembly schemes obtained under certain task time are more applicable to various recycling states.

C. Solution Methods

The existing solution methods include intelligent algorithms [20] and exact solutions [21]. With the development of computers, more and more researchers have built exact models instead of theoretical models. For example, Bentaha et al. [22] developed an exact model and employed a second-order cone programming and upper and lower bound approach based on convex segmented linear function approximation. Additionally, Pour-Massahian-Tafti et al. [23] developed three MILP models to solve the disassembly cost of dynamic DLBP. Because DLBP is an NP-hard problem, the solution space of the DLBP explodes with the increase in problem size. For small-scale DLBP, the exact model can obtain the optimal solution in a limited time. However, the exact model cannot obtain the optimal solution for medium-scale and large-scale DLBPs in a limited time. Nevertheless, the intelligent algorithm can provide decision-makers with a superior solution in a bound time while solving large-scale cases. Intelligent algorithms, such as the genetic algorithm [24], tabu search algorithm [25], simulated annealing algorithm [26], and ant colony algorithm [17], [27] have achieved great success in DLBPs. But the correctness of the encoding and decoding designed by the researchers cannot be tested when solving a new problem. The exact model cannot solve large-scale cases, but the exact model is an effective way to test the correctness of encoding and decoding by solving small-scale cases.

III. Problem Description and Modeling

A. Problem Description

Presently, the studies on parallel line layouts are all PDLBP without parallel workstations. However, in this study, we analyze PW-PDLBP. The two different layouts are shown in Fig. 1. For PDLBP without parallel workstations, all workstations are common workstations and turned ON in series. Uniquely, PW-PDLBP contains two types of workstations: parallel workstations and common workstations. A parallel workstation consists of a pair of independent workstations. The common workstations are responsible for removing tasks on both disassembly

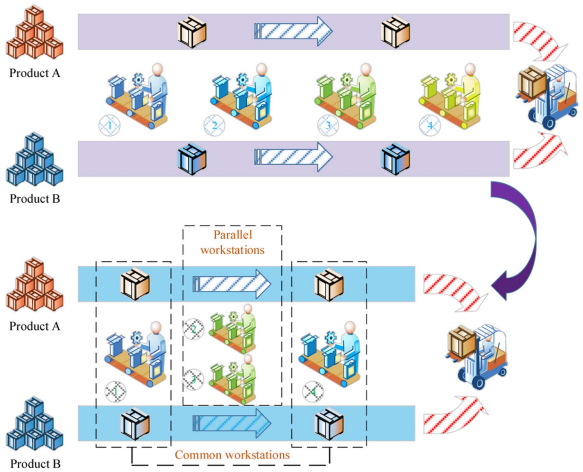

Fig. 1. Layouts of PDLBP and PW-PDLBP.

lines, whereas independent workstations are only responsible for removing tasks on the nearest disassembly line. As a result, PW-PDLBP has the advantages of parallel line layout and linear layout. Additionally, PW-PDLBP can improve space utilization effectively.

B. MINLP Disassembly Time

Symbol Definition

i, j, s	Task indexes.
r	Index of workstation columns.
$\boldsymbol{R}$	Set of parallel workstation columns; $\boldsymbol{R} = \{1, 2\}$.
k	Workstation numbers.
n	Number of disassembly tasks.
m	Number of available workstations.
l, u	Disassembly line indexes.
$\boldsymbol{L}$	Set of disassembly line indexes; $\boldsymbol{L} = \{1, 2\}$.
h_i^l	Hazard attribute. If task i on line l is a hazard task, $h_i^l = 1$; otherwise, $h_i^l = 0$.
$\boldsymbol{I}$	Set of disassembly task indexes; $\boldsymbol{I} = \{1, 2, \ldots, n\}$.
$\boldsymbol{M}$	Set of workstation numbers; $\boldsymbol{M} = \{1, 2, \ldots, m\}$.
t_i^l	Disposal time of task i on the line l.
CT	Cycle time.
TP_{ij}^l	Precedence relationship matrix. If task i on line l is a predecessor task of task j on line l, $TP_{ij}^l = 1$.
α	Large number; $\alpha = 2 \cdot CT \cdot n$.
e_w	Energy consumption of workstation on state.
e_a	Auxiliary energy consumption.
e_d	Protective energy consumption for hazardous tasks.

Variables

w_i^l	Start disassembly time of task i on line l.
x_{ik}^{lr}	Assignment variable of tasks. If task i on line l is assigned to workstation k at the rth column, $x_{ik}^{lr} = 1$; otherwise, $x_{ik}^{lr} = 0$.
z_{ijk}^{ulr}	Disassembly sequence variable. If task i on line u and task j on line l are assigned to workstation k at the rth column simultaneously, and task i is assigned before task j, $z_{ijk}^{ulr} = 1$; otherwise, $z_{ijk}^{ulr} = 0$.
S_k^r	Workstation switching variable. If the workstation k at the rth column is on state, $S_k^r = 1$; otherwise, $S_k^r = 0$.

Objective functions

$$F_1 = \min \sum_{k=1}^{m} S_k^1 \tag{1}$$

$$F_2 = \min \sum_{r=1}^{\|\mathbf{R}\|} \sum_{k=1}^{m} S_k^r \tag{2}$$

$$F_3 = \min \sum_{r=1}^{\|\mathbf{R}\|} \sum_{k=1}^{m} \left(S_k^r \cdot CT - \sum_{l=1}^{\|\mathbf{L}\|} \sum_{i=1}^{n} x_{ik}^{lr} \cdot t_i^l \right)^2 \tag{3}$$

$$F_4 = \min \left(\sum_{r=1}^{\|\mathbf{R}\|} \sum_{k=1}^{m} S_k^r \cdot CT \cdot (e_w + e_a) + \sum_{l=1}^{\|\mathbf{L}\|} \sum_{i=1}^{n} \left(d_i^l \cdot \left(w_i^l + t_i^l \right) \cdot e_d \right) \right). \tag{4}$$

There are four optimization objectives in this study. First, (1) aims to minimize the line length, which can effectively increase the space utilization of enterprises. Second, (2) aims to minimize the number of workstations, which can reduce the employment cost of enterprises. Third, (3) aims to minimize the idle time balancing index. F_3 is a crucial objective of DLBP, which can effectively balance each workstation's workload. Finally, (4) aims to minimize energy consumption. Energy consumption includes energy consumption of workstations on-state, auxiliary energy consumption (lighting and ventilation, etc.), and protective energy consumption for hazardous tasks. Hazardous parts are potentially hazardous to workers throughout the disassembly process. For example, the leakage of phosphor from discarded TV sets during disassembly may poison the workers. Therefore, precautions should be taken before disassembling hazardous parts, and hazardous parts should be removed as early as possible.

Constraints

$$\sum_{r=1}^{\|\boldsymbol{R}\|} \sum_{k=1}^{m} x_{ik}^{lr} = 1, \forall i \in \boldsymbol{I}; \forall l \in \boldsymbol{L} \tag{5}$$

$$\left\lceil \left(\sum_{l=1}^{\|\boldsymbol{L}\|} \sum_{i=1}^{n} t_i^l \right) \Big/ CT \right\rceil \leq \sum_{r=1}^{\|\boldsymbol{R}\|} \sum_{k=1}^{m} S_k^r \leq \|\boldsymbol{L}\| \cdot n \tag{6}$$

$$CT \cdot \left[\left(\sum_{r=1}^{\|\boldsymbol{R}\|} \sum_{k=1}^{m} x_{jk}^{lr} \cdot k \right) - 1 \right] + \sum_{r=1}^{\|\boldsymbol{R}\|} \sum_{k=1}^{m} z_{ijk}^{ulr} \cdot t_i^u \leq w_j^l, \forall i, j \in \boldsymbol{I}; \forall l, u \in \boldsymbol{L} \tag{7}$$

$$w_i^l + t_i^l \leq \left(\sum_{r=1}^{\|\boldsymbol{R}\|} \sum_{k=1}^{m} \left(x_{ik}^{lr} \cdot k \right) \right) \cdot CT, \forall i \in \boldsymbol{I}; \forall l \in \boldsymbol{L} \tag{8}$$

$$x_{ik}^{lr} + x_{jk}^{lr} - 1 \leq z_{ijk}^{llr} + z_{jik}^{llr}, 1 \leq i < j \leq n; \forall r \in \boldsymbol{R}; \forall k \in \boldsymbol{M}; \forall l \in \boldsymbol{L} \tag{9}$$

$$1/2 \cdot (x_{ik}^{lr} + x_{jk}^{lr}) \geq z_{ijk}^{llr} + z_{jik}^{llr}, 1 \leq i < j \leq n; \forall r \in \boldsymbol{R}; \forall k \in \boldsymbol{M}; \forall l \in \boldsymbol{L} \tag{10}$$

$$x_{ik}^{ur} + x_{jk}^{lr} - 1 \le z_{ijk}^{ulr} + z_{jik}^{lur}, \forall i, j \in \boldsymbol{I}; \forall r \in \boldsymbol{R};$$
$$\forall k \in \boldsymbol{M}; \forall l, u \in \boldsymbol{L} \tag{11}$$

$$1/2 \cdot (x_{ik}^{ur} + x_{jk}^{lr}) \ge z_{ijk}^{ulr} + z_{jik}^{lur}, \forall i, j \in \boldsymbol{I}; \forall r \in \boldsymbol{R};$$
$$\forall k \in \boldsymbol{M}; \forall l, u \in \boldsymbol{L} \tag{12}$$

$$\alpha \cdot (1 - z_{jik}^{ulr}) + w_i^l \ge w_j^u + t_j^u, \forall i, j \in \boldsymbol{I}; \forall r \in \boldsymbol{R};$$
$$\forall k \in \boldsymbol{M}; \forall l, u \in \boldsymbol{L} \tag{13}$$

$$w_j^l \ge w_i^l + t_i^l, \forall i, j \in \{\boldsymbol{I} | TP_{ij}^l = 1\}; \forall l \in \boldsymbol{L} \tag{14}$$

$$S_k^r \le \sum_{l=1}^{\|\boldsymbol{R}\|} \sum_{i=1}^{n} x_{ik}^{lr}, \forall r \in \boldsymbol{R}; \forall k \in \boldsymbol{M} \tag{15}$$

$$2 \cdot n \cdot S_k^r \ge \sum_{l=1}^{\|\boldsymbol{L}\|} \sum_{i=1}^{n} x_{ik}^{lr}, \forall r \in \boldsymbol{R}; \forall k \in \boldsymbol{M} \tag{16}$$

$$\alpha \cdot (1 - S_k^2) + S_k^1 \ge 1, \forall k \in \boldsymbol{M} \tag{17}$$

$$\sum_{i=1}^{n} x_{ik}^{1,2} = 0, \forall k \in \boldsymbol{M} \tag{18}$$

$$\alpha \cdot (1 - S_k^2) - \sum_{i=1}^{n} x_{ik}^{2,1} \ge 0, \forall k \in \boldsymbol{M} \tag{19}$$

$$S_k^1 \le S_{k-1}^1, \forall k \in \{2, \ldots, m\} \tag{20}$$

$$x_{ik}^{lr}, z_{ijk}^{ul}, S_k^r = \{0, 1\}, \forall i, j \in \boldsymbol{I}; \forall k \in \boldsymbol{M}; \forall l, u \in \boldsymbol{L}. \tag{21}$$

The complete disassembly mode is employed in this study (5) aims to constrain that all parts must be removed. Equation (6) aims to limit the number of theoretically available workstations. Equation (7) aims to constrain the start disassembly time of each task. Equation (8) aims to ensure that the total disassembly time of the assigned tasks at each workstation does not exceed CT. Equations (9)–(12) aim to ensure that the disassembly of all tasks at each workstation is performed sequentially and without overlap. Constraints (9) and (10) focus on tasks of the same line, and constraints (11) and (12) focus on tasks of different lines. Equation (13) aims to ensure that the disassembly time of tasks assigned to the same workstation does not overlap. Equation (14) aims to ensure that the predecessor tasks are removed before the successor task. Equation (15) aims to ensure that workstations are turned ON only when assigned tasks. Equation (16) aims to ensure that the workstation k at column r is turned ON if there is a task assigned to it. Equation (17) aims to ensure that independent workstations are turned ON in pairs. Equation (18) aims to constrain independent workstations responsible for disassembling an adjacent line's tasks. Equation (19) aims to ensure that the states of parallel and common workstations at each station do not conflict. Equation (20) aims to ensure that the workstations are turned ON sequentially. Constraint (21) lists that x_{ik}^{lr}, z_{ijk}^{ulr}, and S_k^r are 0-1 variables.

IV. MEDE

The MEDE is a stochastic heuristic and employs vector operation among individuals to update the population and approach the global optimum. The populations of MEDE are classified into elite evolutionary populations and random differential evolutionary populations. The two populations are performed using different operations for faster search and to avoid falling into local optima, respectively. In addition, MEDE employs the Pareto [28] and NSGA-II crowded distance mechanism [29] for screening individuals in the population. The flow chart of MEDE is shown in Fig. 2.

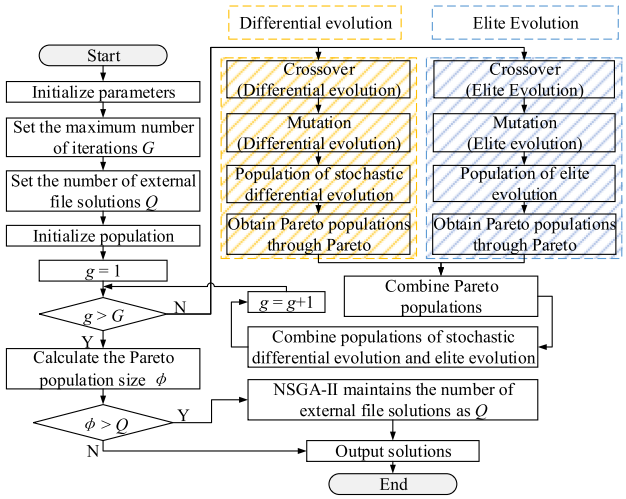

Fig. 2. Flow chart of MEDE.

A. Encoding and Decoding

The encoding in this study is obtained from literature [3] that addresses PDLBP. The pseudocode of decoding is shown in the following.

Input: Disassembly Sequence

Initialization of the current workstation number ($WS = 1$) and the remaining workable time of the current workstation ($RT = CT$).
For all tasks
  t = disassembly time of the currently assigned task.
  IF $t < RT$
    $RT = RT - t$;
    This task is assigned to the workstation WS and the assignment information of this task is saved.
    Else
    $WS = WS + 1$;
    $RT = CT$;
    This task is assigned to the workstation WS and the assignment information of this task is saved.
  End IF
End For
For all enabled workstations
  IF two consecutive workstations are assigned tasks that belong to their adjacent lines & and two adjacent lines are different
    parallel workstations are turned on.
  End If
End For

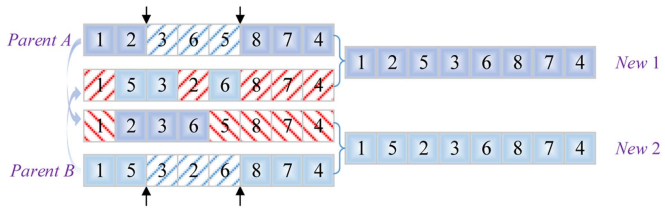

Fig. 3. Crossover operation.

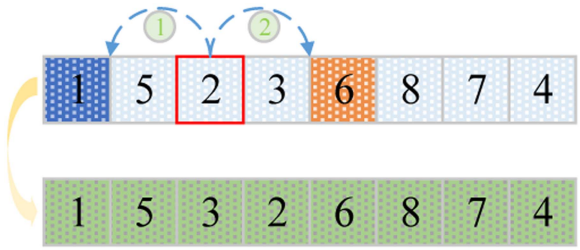

Fig. 4. Mutation operation.

B. Stochastic Evolutionary Population

Individuals of the stochastic evolutionary population are crossed by the operation, as shown in Fig. 3. Two identical points on two individuals (Parent A and Parent B) are selected. The selected tasks of Parent A are rearranged in the order they are listed in Parent B, resulting in a new individual. The crossover operation of Parent B is the same as that of Parent A.

The mutation operation is shown in Fig. 4. A task is randomly selected among an individual. The selected task is inserted somewhere between its predecessor and successor tasks. For example, task 2 is selected, and it can be inserted into one of the locations between tasks 2 and 5 and between tasks 3 and 6.

C. Elite Evolution Differential Population

The proportion of the elite population is pa of the total population count. The crossover shown in (22) is employed for the elite evolution differential population.

$$u_{i,G+1} = \begin{cases} v_{i,G+1}, & \text{if rand}\,(i) \le CR \\ x_{i,G+1}, & \text{otherwise} \end{cases}. \tag{22}$$

$*\text{Rand}(i)$ is a stochastic number between [0, 1] and CR is a crossover operation.

According to the characteristics of PW-PDLBP, the mutation operation of the stochastic differential evolution population adopts the operation shown in (23), which seeks optimal by integer downward for the difference vectors.

$$v_{i,G+1} = \text{floor}\left[x_{r1,G} + F \cdot (x_{r2,G} - x_{r3,G})\right]. \tag{23}$$

V. MEDE and MINLP Model Validations

The solving software of the MINLP model is GUROBI 9.5.1. The solving software of MEDE is MATLAB R2022a. All optimization software used in this study was run on a personal computer with the following configuration: Intel(R) Xeon(R) CPU E5-1620 v3 @ 3.50 GHz, 32 GB RAM, Windows 10.

TABLE I
COMPUTATIONAL RESULTS OF MINLP MODEL AND MEDE

Case	MINLP					MEDE				
	F_1	F_2	F_3	F_4	Time/s	F_1	F_2	F_3	F_4	Time/s
P8 +	6	–	–	–	4.57	**6**	12	1532	76.15	0.45
P10	–	10	–	–	353.48	7	**10**	252	62.11	
	–	–	240	–	1690.44	8	**10**	**240**	62.11	
	–	–	–	61.89	164.03	8	**10**	252	**61.89**	

The best values of each objective obtained by MEDE.

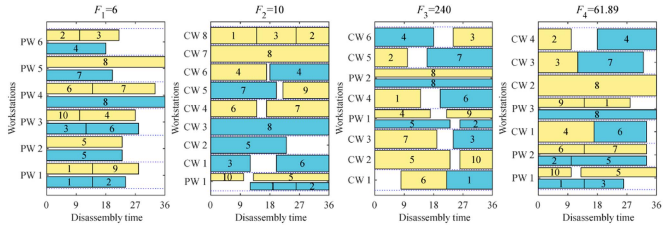

Fig. 5. Gantt charts of optimal value of each objective.

A. MINLP Model Solving

To verify the validity of the MINLP model and the mechanism of encoding and decoding, we applied MINLP and MEDE to solve a hybrid case consisting of P8 [25] and P10 [30].

For this case, CT is set as 36. Energy consumption is set as: $e_w = 0.12$ kJ/s, $e_a = 0.05$ kJ/s, and $e_d = 0.01$ kJ/s. The computational results of the MINLP model and MEDE are shown in Table I. The best values obtained by MEDE for each objective are written in bold.

In Table I, both the MINLP model and MEDE can obtain the optimal values of four objectives within a limited time. However, the running time of MEDE is significantly shorter than that of the MINLP model. In addition, Gantt charts of the optimal value of each objective are drawn, as shown in Fig. 5. In Fig. 5, the yellow and blue boxes represent the tasks of P10 and P8, respectively. PW and CW represent the parallel workstation and common workstation, respectively. It can be observed from Fig. 5 that the precedence relationships of tasks are correct, and no task is overlapping. Thus, the mechanism of encoding and decoding is correct, and MEDE can solve PW-PDLBP effectively.

B. Performance of MEDE

MEDE is applied to solve a P52 case [28] to objectively verify its solving performance. Furthermore, the four existing algorithms, namely hybrid migrating birds optimization (HMBO) [31], multiobjective discrete fruit fly optimization algorithm (MDFOA) [32], flower pollination algorithm (FPA) [33], and genetic simulated annealing algorithm (GASA) [34], are compared with MEDE. After orthogonal experimental designs, parameters of MEDE are set as follows: N (population size) = 350, $maxcycle$ (maximum number of iterations) = 300, F (mutation operation) = 0.8, CR (crossover operation) = 0.1, and pa (proportion of elite individuals) = 0.4. The computational results of MEDE are shown in Table II.

In Table II, MEDE obtained 10 Pareto solutions. The numbers of HMBO (9), MDFOA (9), and FPA (8) are less than that of MEDE. As the values of F_{Idle} are the same for the

TABLE II
COMPUTATIONAL RESULTS OF MEDE SOLUTION FOR P52

No.	F_{Idle}	F_{Smooth}	F_{Cost}	*NPS*	R_{NPS}
1	0.0579	**1.0000**	143.988		
2	0.0579	0.6658	**124.686**		
3	0.0579	0.9343	128.232		
4	0.0579	0.9987	133.068		
5	0.0579	0.9986	128.268	6	6/13
6	0.0579	0.9997	140.304		
7	0.0579	0.9310	127.146		
8	0.0579	0.9995	135.600		
9	0.0579	0.8785	125.922		
10	0.0579	0.8086	124.800		

*NPS is the number of Pareto solutions; RNPS is the proportion of Pareto solutions.

The bolded values represent the best values obtained by MEDE for F_{Smooth} and F_{Cost}.

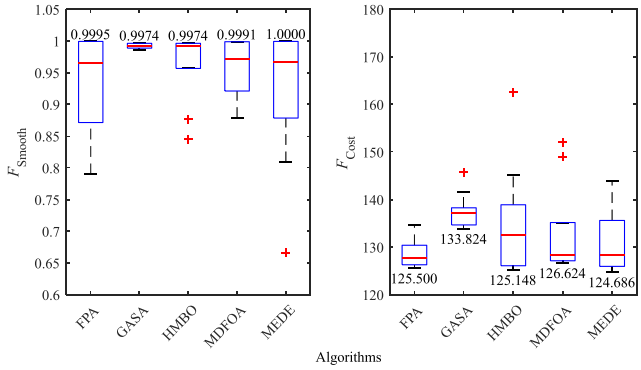

Fig. 6. Boxplots of F_{Smooth} and F_{Cost}.

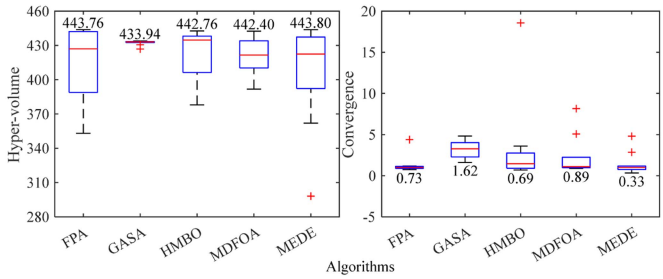

Fig. 7. Boxplots of *HV* and *C* metrics.

five algorithms, the boxplots of only F_{Smooth} and F_{Cost} are drawn in Fig. 6. In Fig. 6, MEDE obtained the best values of F_{Smooth} and F_{Cost}. The optimal values of F_{Smooth} and F_{Cost} obtained by MEDE are 1 and 124.686, whereas that of the other four algorithms are 0.995 and 125.148, respectively. Moreover, the median of F_{Cost} for MEDE outperforms that of the other four algorithms. However, owing to the nonpositive correlation between F_{Smooth} and F_{Cost}, the median of F_{cost} for MEDE is inferior to that of GASA and HMBO. The comparisons of the hyper-volume (HV) [35] and convergence (C) [36] metrics are shown in Fig. 7. It can be seen that the optimal algorithm for both *HV* and *C* metrics in Fig. 7 is MEDE. The Pareto solutions of F_{Smooth} and F_{Cost} of the five algorithms are drawn, as shown in Fig. 8. There are 13 groups of nondominated solutions among the five algorithms. The *NPS* of MEDE is 6 (Nos. 1, 2, 5, 6, 9, and 10), which contributes to 46.15% of the total number of nondominated solutions. *NPS* accounts for 6/10 of the number of solutions of MEDE. Thus, MEDE can solve DLBP effectively.

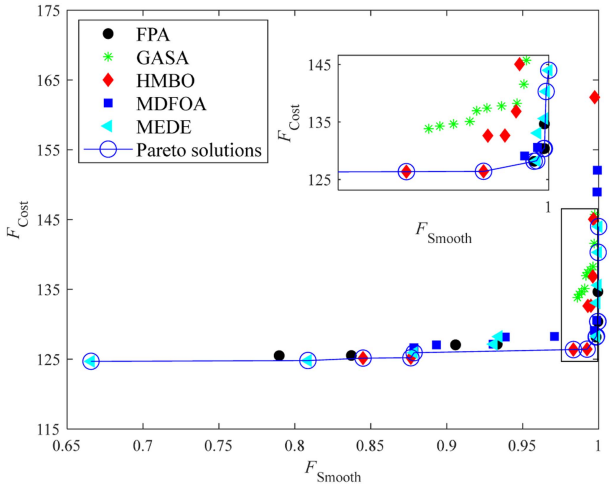

Fig. 8. Pareto solutions of F_{Smooth} and F_{cost}.

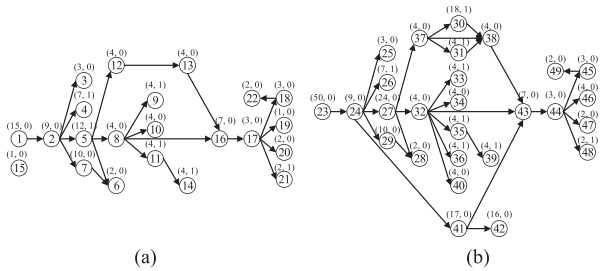

Fig. 9. Information of P22 and P27. (a) CRT Television A. (b) CRT Television B.

VI. APPLICATION AND ANALYSIS OF AN INSTANCE

An instance consisting of two types of TV sets (P25 and P27) is applied to PW-PDLBP. Fig. 9 shows the information of P25 and P27. In Fig. 9, the number in the circle represents the task index, the first number above the circle represents the disposal time of the task, and the second number indicates whether the current task is harmful ("0" suggests that the task is harmless and "1" suggests it is hazardous). For this instance, the cycle time is set to 60, and parameters of energy consumption are set as follows: $e_w = 0.12$ kJ/s, $e_a = 0.05$ kJ/s, and $e_d = 0.01$ kJ/s. Finally, the results are shown in Table III, where disassembly schemes are shown in the sixth column. Additionally, to verify the effectiveness of PW-PDLBP in enabling enterprises to lower site costs (shorten line length), this study decodes the results of PW-PDLBP according to PDLBP [3] without the parallel workstation. The line length (F_1) and the number of workstations (F_2) of PDLBP are also presented in Table III.

There is no positive correlation between F_1 and both F_3 and F_4, as per Table III. With parallel workstations turned ON, independent workstations can only disassemble products on the adjacent line. Therefore, there is some risk that F_1 will deteriorate F_3 and F_4. Furthermore, there is a conflicting relationship between F_3 and F_4. F_4 cannot be optimal if F_3 is optimal and vice versa. However, there is a positive correlation between F_2 and both F_3 and F_4. F_3 and F_4 are only optimal with a minimum number of workstations. Therefore, the optimal number of workstations in the results is always

TABLE III
SOLUTION RESULTS OF P22 AND P27

No.	PW-PDLBP					PDLBP	
	F_1	F_2	F_3	F_4	Schemes	F_1	F_2
1	4	6	258	84.89	{{1,2,5,12,8,13,16},{23,24}}→{{17,18,7,11,6,9,3,4,14,10,21,22},{27,37,32,35,36,29,28,31}}→{30,38,41,43,44,45,15,19}→{42,34,20,49,48,40,46,47,33,26,25,39}	6	6
2	4	6	300	83.97	{{1,2,5,12,8,11,13,16},{23,24}}→{{17,18,7,6,3,9,10,4,20,15,21,14,22},{27,37,32,35,36,29,28,31}}→{30,38,41,43,44,45}→{42,34,46,19,49,47,26,40,25,48,33,39}	6	6
3	5	6	224	82.98	{{1,2,5,12,8,13,16},{23,24}}→{17,11,18,7,6,27,37,32}→{29,35,28,36,31,3,9,30,38}→{41,43,44,45,14,4,34,22,21,10}→{42,20,19,47,15,33,39,40,49,46,26,48,25}	6	6
4	5	6	262	81.04	{{1,2,5,12,13,8,16,3},{17,18,11,23}}→{7,6,24,27,37,32}→{29,28,35,36,20,31,30,38,9}→{41,43,44,45,4,14,15,34,21,10}→{42,46,49,33,19,40,47,48,26,25,22,39}	6	6
5	5	6	362	80.34	{{1,2,5,12,13,8,16,3},{17,18,11,23}}→{7,6,24,27,37,32,35}→{36,29,28,31,30,41,20}→{38,43,44,45,25,4,14,34,15,9,10,21}→{42,19,40,39,22,46,47,49,48,33,26}	6	6
6	6	6	224	80.93	{1,2,5,12,13,8,16,17}→{18,11,23}→{7,6,24,27,37,32}→{29,28,35,36,30,31,3,38,9}→{41,43,44,45,4,14,21,20,10,34}→{42,39,15,22,48,40,26,19,49,46,25,33,47}	6	6
7	6	6	252	80.55	{1,2,5,12,8,11,13,16}→{17,18,23}→{7,6,24,27,37,32,35}→{36,29,28,3,15,31,22,30,38,9}→{41,43,44,45,4,14,10,34,21}→{42,19,20,26,49,33,39,46,40,47,48,25}	6	6
8	6	6	268	80.44	{1,2,5,12,8,13,16,17}→{18,11,23}→{7,6,24,27,37,32,35}→{36,29,28,30,31,41}→{4,38,43,44,45,22,34,15,9,10,14,21,49,20,19}→{42,3,26,25,33,46,48,40,47,39}	6	6
9	6	6	272	80.37	{1,2,5,12,8,11,13,16}→{17,18,23}→{7,6,24,27,37,32,35}→{36,29,28,31,41,30}→{4,38,43,44,45,9,3,10,22,14,34,21,20}→{42,39,19,26,49,46,40,33,15,47,25,48}	6	6
10	6	6	300	80.16	{1,2,5,12,8,11,13,16}→{17,18,23}→{7,6,24,27,37,32,35}→{36,29,28,31,30,20,41}→{38,43,44,45,4,14,34,22,15,10,21,9,3}→{42,46,48,40,19,25,39,47,49,33,26}	6	6

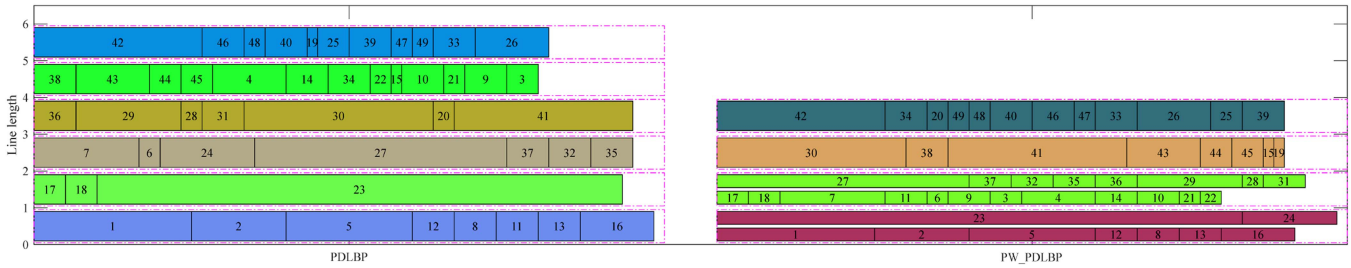

Fig. 10. Optimal layouts of PDLBP and PW-PDLBP.

6. For PDLBP, as its workstations are common workstations that are arranged in series, the line length of PDLBP is the number of workstations, i.e., the optimal values of F_1 and F_2 of PDLBP are 6. In Fig. 10, the layouts of the optimal value of the line length for PW-PDLBP and PDLBP are plotted. From Fig. 10, it is obvious that PW-PDLBP can effectively shorten the line length and improve the space utilization of enterprises. PW-PDLBP improves space utilization by 33.33% more than PDLBP. In addition, Table III also provides ten groups of Pareto schemes. Compared with the schemes provided by PDLBP, the decision-makers can also choose among the schemes provided by PW-PDLBP based on the space utilization of the enterprises. But this feature is not available in PDLBP. Besides, in the schemes provided by PW-PDLBP, schemes 1 and 2 are the most space-saving schemes, schemes 3 and 6 balance the workstation load best, and scheme 5 not only saves space but also consumes less energy.

VII. CONCLUSION

In this article, we constructed a MINLP model and developed a MEDE for PW-PDLBP. Moreover, the correctness of the MINLP model and the encoding and decoding is verified by solving a case consisting of P8 and P10. Then, the solving performance of MEDE was validated by comparing it with four algorithms, which are HMBO, MDFOA, FPA, and GASA. Finally, MEDE was applied to optimize a disassembly instance that consists of two types of EOL TV sets, and ten Pareto schemes were provided to decision-makers. Additionally, the line length of PW-PDLBP was compared with that of PDLBP to demonstrate the validity of the PW-PDLBP. The comparison results showed that PW-PDLBP could utilize the space more compactly than PDLBP for enterprises.

In future research, the suitability of multiparallel PW-PDLBP for the actual needs of enterprises will be analyzed, and extension of the AND/OR relationship to PDLBP is also interesting to study. For the mathematical model, performing a sensitivity analysis will explore more management significance. Moreover, recording and optimizing the actual size of workstations is a possible topic.

REFERENCES

[1] S. M. Gupta and A. Gungor, "Product recovery using a disassembly line: Challenges and solution," in *Proc. IEEE Int. Symp. Electron. Environ.*, 2001, pp. 36–40, doi: 10.1109/isee.2001.924499.

[2] F. Pistolesi and B. Lazzerini, "TeMA: A tensorial memetic algorithm for many-objective parallel disassembly sequence planning in product refurbishment," *IEEE Trans. Ind. Inform.*, vol. 15, no. 6, pp. 3743–3753, Jun. 2019, doi: 10.1109/TII.2019.2904631.

[3] L. Zhu, Z. Zhang, and C. Guan, "Multi-objective partial parallel disassembly line balancing problem using hybrid group neighbourhood search algorithm," *J. Manuf. Syst.*, vol. 56, pp. 252–269, 2020, doi: 10.1016/j.jmsy.2020.06.013.
[4] K. Z. Gao, Z. M. He, Y. Huang, P. Y. Duan, and P. N. Suganthan, "A survey on meta-heuristics for solving disassembly line balancing, planning and scheduling problems in remanufacturing," *Swarm Evol. Comput.*, vol. 57, pp. 1–10, 2020, doi: 10.1016/j.swevo.2020.100719.
[5] R. Storn, "System design by constraint adaptation and differential evolution," *IEEE Trans. Evol. Comput.*, vol. 3, no. 1, pp. 22–34, Apr. 1999, doi: 10.1109/4235.752918.
[6] X. Wan, X. Zuo, and X. Zhao, "A differential evolution algorithm combined with linear programming for solving a closed loop facility layout problem," *Appl. Soft Comput.*, vol. 121, pp. 1–14, 2022, doi: 10.1016/j.asoc.2022.108725.
[7] H. Liu, Q. Chen, N. Pan, Y. Sun, Y. An, and D. Pan, "UAV stocktaking task-planning for industrial warehouses based on the improved hybrid differential evolution algorithm," *IEEE Trans. Ind. Inform.*, vol. 18, no. 1, pp. 582–591, Jan. 2022, doi: 10.1109/TII.2021.3054172.
[8] G.-G. Wang, D. Gao, and W. Pedrycz, "Solving multi-objective fuzzy job-shop scheduling problem by a hybrid adaptive differential evolution algorithm," *IEEE Trans. Ind. Inform.*, vol. 18, no. 12, pp. 8519–8528, Dec. 2022, doi: 10.1109/TII.2022.3165636.
[9] W. Liang, Z. Zhang, Y. Zhang, P. Xu, and T. Yin, "Improved social spider algorithm for partial disassembly line balancing problem considering the energy consumption involved in tool switching," *Int. J. Prod. Res.*, pp. 1–18, 2022, doi: 10.1080/00207543.2022.2069059.
[10] J. He, F. Chu, A. Dolgui, F. Zheng, and M. Liu, "Integrated stochastic disassembly line balancing and planning problem with machine specificity," *Int. J. Prod. Res.*, vol. 60, no. 5, pp. 1688–1708, 2022, doi: 10.1080/00207543.2020.1868600.
[11] J. Liang, S. Guo, B. Du, W. Liu, and Y. Zhang, "Restart genetic flatworm algorithm for two-sided disassembly line balancing problem considering negative impact of destructive disassembly," *J. Cleaner Prod.*, vol. 355, pp. 1–16, 2022, doi: 10.1016/j.jclepro.2022.131708.
[12] S. Agrawal and M. K. Tiwari, "A collaborative ant colony algorithm to stochastic mixed-model U-shaped disassembly line balancing and sequencing problem," *Int. J. Prod. Res.*, vol. 46, no. 6, pp. 1405–1429, 2008, doi: 10.1080/00207540600943985.
[13] Z. Li and M. N. Janardhanan, "Modelling and solving profit-oriented U-shaped partial disassembly line balancing problem," *Expert Syst. Appl.*, vol. 183, pp. 1–13, 2021, doi: 10.1016/j.eswa.2021.115431.
[14] S. Hezer and Y. Kara, "A network-based shortest route model for parallel disassembly line balancing problem," *Int. J. Prod. Res.*, vol. 53, no. 6, pp. 1849–1865, 2015, doi: 10.1080/00207543.2014.965348.
[15] I. Kucukkoc, "Balancing of two-sided disassembly lines: Problem definition, MILP model and genetic algorithm approach," *Comput. Oper. Res.*, vol. 124, pp. 1–17, 2020, doi: 10.1016/j.cor.2020.105064.
[16] K. Wang, X. Li, L. Gao, P. Li, and S. M. Gupta, "A genetic simulated annealing algorithm for parallel partial disassembly line balancing problem," *Appl. Soft Comput.*, vol. 107, pp. 1–19, 2021, doi: 10.1016/j.asoc.2021.107404.
[17] S. M. McGovern and S. M. Gupta, "Ant colony optimization for disassembly sequencing with multiple objectives," *Int. J. Adv. Manuf. Technol.*, vol. 30, pp. 481–496, 2006, doi: 10.1007/s00170-005-0037-6.
[18] Z. Zhang, K. Wang, L. Zhu, and Y. Wang, "A Pareto improved artificial fish swarm algorithm for solving a multi-objective fuzzy disassembly line balancing problem," *Expert Syst. Appl.*, vol. 86, pp. 165–176, 2017, doi: 10.1016/j.eswa.2017.05.053.
[19] C. B. Kalayci, A. Hancilar, A. Gungor, and S. M. Gupta, "Multi-objective fuzzy disassembly line balancing using a hybrid discrete artificial bee colony algorithm," *J. Manuf. Syst.*, vol. 37, pp. 672–682, 2015, doi: 10.1016/j.jmsy.2014.11.015.
[20] G. Tian, Y. Ren, Y. Feng, M. C. Zhou, H. Zhang, and J. Tan, "Modeling and planning for dual-objective selective disassembly using and/or graph and discrete artificial bee colony," *IEEE Trans. Ind. Inform.*, vol. 15, no. 4, pp. 2456–2468, Apr. 2019, doi: 10.1109/TII.2018.2884845.
[21] A. Koc, I. Sabuncuoglu, and E. Erel, "Two exact formulations for disassembly line balancing problems with task precedence diagram construction using an AND/OR graph," *IIE Trans.*, vol. 41, no. 10, pp. 866–881, 2009, doi: 10.1080/07408170802510390.
[22] M. L. Bentaha, O. Battaïa, and A. Dolgui, "An exact solution approach for disassembly line balancing problem under uncertainty of the task processing times," *Int. J. Prod. Res.*, vol. 53, no. 6, pp. 1807–1818, 2015, doi: 10.1080/00207543.2014.961212.
[23] M. Pour-Massahian-Tafti, M. Godichaud, and L. Amodeo, "New models and efficient methods for single-product disassembly lot-sizing problem with surplus inventory decisions," *Int. J. Prod. Res.*, vol. 59, no. 22, pp. 6898–6918, 2021, doi: 10.1080/00207543.2020.1829148.
[24] X. Guo, M. Zhou, S. Liu, and L. Qi, "Multiresource-constrained selective disassembly with maximal profit and minimal energy consumption," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 2, pp. 804–816, Apr. 2021, doi: 10.1109/TASE.2020.2992220.
[25] C. B. Kalayci and S. M. Gupta, "A tabu search algorithm for balancing a sequence-dependent disassembly line," *Prod. Plan. Control*, vol. 25, no. 2, pp. 149–160, 2014, doi: 10.1080/09537287.2013.782949.
[26] C. B. Kalayci, S. M. Gupta, and K. Nakashima, "A simulated annealing algorithm for balancing a disassembly line," in *Proc. Des. Innov. Value Towards Sustain. Soc.: Proc. Eco-Des.: 7th Int. Symp. Environ. Conscious Des. Inverse Manuf.*, 2012, pp. 714–719, doi: 10.1007/978-94-007-3010-6_143.
[27] S. M. McGovern and S. M. Gupta, "Uninformed and probabilistic distributed agent combinatorial searches for the unary NP-complete disassembly line balancing problem," *Proc. SPIE*, vol. 5997, 2005, pp. 81–92, doi: 10.1117/12.629121.
[28] L. P. Ding, J. R. Tan, Y. X. Feng, and Y. C. Gao, "Multiobjective optimization for disassembly line balancing based on Pareto ant colony algorithm," *Comput. Integr. Manuf. Syst.*, vol. 15, no. 7, pp. 1406–1413, 2009, doi: 10.13196/j.cims.2009.07.160.dinglp.005.
[29] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002, doi: 10.1109/4235.996017.
[30] S. M. McGovern and S. M. Gupta, "2-opt heuristic for the disassembly line balancing problem," in *Proc. 3rd Int. Conf. Environ. Conscious Manuf.*, 2004, vol. 5262, pp. 71–84, doi: 10.1117/12.516155.
[31] L. Zhu, Z. Zhang, Y. Wang, and N. Cai, "On the end-of-life state oriented multi-objective disassembly line balancing problem," *J. Intell. Manuf.*, vol. 31, no. 6, pp. 1403–1428, 2020, doi: 10.1007/s10845-019-01519-3.
[32] C. Ning, Z. Zeqiang, Z. Binsen, and L. Liuke, "Multi-objective disassembly line balancing optimization and analytic hierarchy process decision-making considering energy consumption," *Comput. Integr. Manuf. Syst.*, vol. 25, no. 1, pp. 125–136, 2019, doi: 10.13196/j.cims.2019.01.012.
[33] Y. Zeng, Z. Zhang, Y. Zhang, S. Liu, and Y. Li, "Pareto flower pollination algorithm for multi-objective bucket brigade mixed-model disassembly line balancing problem," *Comput. Integr. Manuf. Syst.*, vol. 26, no. 3, pp. 760–774, 2020, doi: 10.13196/j.cims.2020.03.018.
[34] K. Wang, Z. Zhang, L. Zhu, and B. Zou, "Pareto genetic simulated annealing algorithm for multi objective disassembly line balancing problem," *Comput. Integr. Manuf. Syst.*, vol. 23, no. 6, pp. 1277–1285, 2017, doi: 10.13196/j.cims.2017.06.013.
[35] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Comput.*, vol. 19, no. 1, pp. 45–76, 2011, doi: 10.1162/EVCO_a_00009.
[36] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999, doi: 10.1109/4235.797969.

Wei Liang was born in Sichuan, China, in 1998. He received the B.S. degree in material forming and control engineering from Southwest University of Science and Technology, Mianyang, China, in 2020. He is currently working toward the Ph.D. in engineering with the School of Mechanical Engineering, Southwest Jiaotong University, Chengdu, China.

His research interests include production line balancing and intelligent optimization.

Zeqiang Zhang was born in Zhejiang, China, in 1978. He received the Ph.D. degree in mechanical design and theory from Southwest Jiaotong University, Chengdu, China, in 2006.

He is currently a Professor and Doctoral Tutor of the School of Mechanical Engineering, Southwest Jiaotong University. His current research interests include manufacturing systems and intelligent optimization.

Yanqing Zeng was born in Sichuan, China, in 1994. She received the B.S. degree in industrial engineering from Southwest University of Science and Technology, Mianyang, China, in 2020. She is currently working toward the Ph.D. in engineering with the School of Mechanical Engineering, Southwest Jiaotong University, Chengdu, China.

Her research interests include production line balancing and intelligent optimization.

Tao Yin was born in Shaanxi, China, in 1990. He received the B.S. degree in civil engineering from Northwest A&F University, Xianyang, China, in 2014. He is currently working toward the Ph.D. in engineering with the School of Mechanical Engineering, Southwest Jiaotong University, Chengdu, China.

His research interests include the optimization algorithm and intelligent manufacturing system.

Tengfei Wu was born in Sichuan, China, in 1991. He received the B.S. degree in agricultural mechanization and its automation from Sichuan Agricultural University, Ya'an, China, in 2016. He is currently working toward the Ph.D. in engineering with the School of Mechanical Engineering, Southwest Jiaotong University, Chengdu, China.

His research interests include production line balancing and intelligent optimization.