Environmental Science and Pollution Research
https://doi.org/10.1007/s11356-023-27081-3

OPPORTUNITIES AND CHALLENGES OF INDUSTRIAL SUSTAINABLE DEVELOPMENT IN THE POST-PANDEMIC WORLD

Hybrid evolutionary algorithm for stochastic multiobjective disassembly line balancing problem in remanufacturing

Guangdong Tian[1] · Xuesong Zhang[2] · Amir M. Fathollahi-Fard[3] · Zhigang Jiang[4] · Chaoyong Zhang[5] · Gang Yuan[6] · Duc Truong Pham[7]

Received: 30 December 2022 / Accepted: 13 April 2023

Abstract
With the development of the industrial economy and the accelerated renewal of products, many end-of-life products (EOL) have been generated to pollute our environment. This fact highlights the importance of recycling and remanufacturing EOL products as an active research topic. An efficient disassembly line is one solution for improving the remanufacturing and recycling processes of EOL products while reducing the environmental pollution. Although many optimization models and intelligent algorithms were developed to address the disassembly line balancing problem (DLBP), uncertainty was ignored by them. To alleviate the drawbacks of uncertainty for the disassembly operation, this study proposes a stochastic multi-objective optimization model for the DLBP minimizing the disassembly idle rate, smoothness, and energy consumption generated during the operation under uncertain operation time. Another novelty of this paper is to present an improved version of the northern goshawk optimization algorithm using a stochastic simulation method to solve the proposed model. The feasibility of the proposed model and the applicability of the developed algorithm are shown by two extensive examples. Finally, the performance of the proposed algorithm is revealed by a comparison with recent and state-of-the-art algorithms from the literature.

Keywords Remanufacturing · Disassembly line balancing · Green manufacturing · Disassembly planning

Responsible Editor: Philippe Garrigues

✉ Xuesong Zhang
zz980818@126.com

Guangdong Tian
tiangd2013@163.com

Amir M. Fathollahi-Fard
afathollahifard@uvic.ca

Zhigang Jiang
jiangzhigang@wust.edu.cn

Chaoyong Zhang
zcyhust@hust.edu.cn

Gang Yuan
yuangang1224@163.com

Duc Truong Pham
d.t.pham@bham.ac.uk

[1] School of Mechanical-Electrical and Vehicle Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, China

[2] Transportation College, Northeast Forestry University, Harbin 150040, China

[3] Peter B. Gustavson School of Business, University of Victoria, 1700, Victoria, BC V8P5C2, Canada

[4] Key Laboratory of Metallurgical Equipment and Control Technology, Ministry of Education, Wuhan University of Science and Technology, Wuhan 430081, China

[5] State Key Laboratory of Digital Manufacturing Equipment & Technology, Huazhong University of Science & Technology (HUST), Wuhan 430074, People's Republic of China

[6] China Institute for Agricultural Equipment Industrial Development, Jiangsu University, Zhenjiang 212000, China

[7] Department of Mechanical Engineering, University of Birmingham, Birmingham B15 2TT, UK

Published online: 28 April 2023

Introduction

Nowadays, manufacturing industries consume considerable non-renewable resources and create many end-of-life (EOL) products which are at the end of their useful life (Tian et al., 2022b, c). It is essential to recycle and remanufacture EOL products for reducing the number of wastes sent to landfills (Mojtahedi et al. 2021), conserving natural resources (Chen et al., 2018), and preventing pollution (Kizilay et al. 2022). The disassembly line balancing problem (DLBP) is a solution for recycling and remanufacturing EOL products (Zhang et al. 2021). This fact motivates our attempts to propose a stochastic multi-objective DLBP using an improved metaheuristic algorithm.

The base models for the DLBP refer to the disassembly sequence planning problem (Demello et al. 1990) where the main goal is to find the optimal sequence of disassembly tasks (Wu et al., 2022a, b). To address the product complexity and real-life constraints of disassembly operations, scholars focused on the modeling and optimization of DLBP (Tian et al., 2022a, b, c) where various methods like disassembly matrices (Zhou and Bian, 2022), AND\OR graphs (Tian et al. 2019), and hybrid graphs (Wang et al. 2020) were used in the literature. Among them, hybrid graphs were repeatedly used in the literature (Yin et al. 2022), and this paper also considers a hybrid graph to formulate a stochastic multi-objective DLBP.

The earliest studies focused on the optimization of cost or profitability in this research area (Demello and Sanderson, 1990). However, recent studies consider different optimization criteria simultaneously like costs, profits, smoothness, and energy consumption (Tian et al., 2022c). For example, Guo and Zhang (2020) solved their proposed bi-objective DLBP model using a decentralized search algorithm to minimize disassembly costs while maximizing the disassembly profits. Zhu et al. (2018) constructed a multi-objective DLBP model minimizing the number of workstations and the average of maximum hazard from the disassembly line while maximizing the smoothing rate. A Pareto-based firefly algorithm (FA) using random-key procedures was developed to solve their DLBP. Recently, Tian et al. (2022c) considered smoothness rate, disassembly cost, and energy consumption involved in the disassembly line where a cloud model is represented to consider the uncertainty, and an enhanced social engineering optimizer (SEO) was deployed for solving their multi-objective model.

Although uncertainty is one of the inherent parts of DLBP, there are a few papers that considered the uncertainty in the modeling of DLBP using stochastic (Altekin, 2017) or fuzzy theories (Altekin et al. 2013). For example, Altekin (2017) studied stochastic DLBP using two second-order cone programming models and five piecewise linear mixed integer programming methods. Bentaha et al. (2020) developed a stochastic DLBP maximizing the revenue while balancing the workloads under uncertainty where the processing time of tasks is assumed as a random variable with an unknown probability function. They combined stochastic programming with L-shaped and Monte Carlo simulation for addressing their optimization model. More recently, He et al. (2022) studied a stochastic DLBP minimizing the total cost of disassembly where component yield ratios and demands were uncertain. After formulating two-stage stochastic programming, a sample average approximation (SAA) and an L-shaped algorithm were proposed to solve their problem. The uncertainty of EOL products due to damage and deformation caused by use will have a large impact on their DLBP operation process, and it is an important issue to consider this uncertainty to reduce its negative impact on DLBP in order to arrange the disassembly operation scheduling more rationally. This fact motivates the research in this paper.

The DLBP is known an NP-hard optimization problems, and exact methods are not able for solving real and large instances (Tian et al., 2022c). In this regard, the literature review is very rich in the development of intelligent algorithms based on heuristics and metaheuristics (Bentaha et al., 2020; He et al. 2022; Fathollahi-Fard et al. 2018; Wu et al. 2022a; Guo et al. 2023), and the main contribution of many studies is to propose new algorithms (Zhang et al. 2021). For example, Wu et al. (2022b) proposed simplified discrete gravitational search algorithm (GSA) to solve DLBP and compared it with traditional metaheuristic algorithms. Zhang et al. (2015) constructed a parallel disassembly model for complex products and solved their proposed model using an improved genetic algorithm (GA) with new crossover operator. Ren et al. (2018) constructed an asynchronous parallel disassembly planning model considering operator constraints and tool number constraints and solved the model using a GA. In another paper (Ren et al. 2017), they studied an improved GSA with profit-oriented partial disassembly solutions. To improve the disassembly efficiency of obsolete agricultural machines, Yang et al. (2019b) proposed an improved multi-objective fruit fly optimization algorithm. At last but not least, Zhang et al. (2021) improved the SEO by incorporating crossover operators and rules of swap order into the algorithm.

In addition to the above algorithms, many other metaheuristics like ant colony optimization (ACO) (Zhu et al. 2014), artificial bee colony (ABC) (Wang et al. 2021), and flower pollination algorithm (Jiao et al. 2016) have been used. The main reason behind the development of a new algorithm refers to the no-free-lunch theory for optimization (Wolpert et al. 1997). This theory confirms that it is always possible for a new algorithm to solve an optimization

problem like DLBP, better than existing algorithms in the literature. This fact motivates us to propose the northern goshawk optimization algorithm (NGO) (Dehghani et al. 2021) which has been never applied to the literature of DLBP. Hence, this study proposes an improved NGO using stochastic simulation for a stochastic DLBP.

In conclusion, this study makes the following highlights:

- A stochastic DLBP is developed to minimize the disassembly line idle rate, the disassembly line smoothness, and the energy consumption generated during the disassembly process.
- An improved NGO using a stochastic simulation method is developed.
- Real-world industrial case studies are considered to show the applicability of this study.
- The proposed algorithm is compared with both recent and traditional algorithms from the literature.

The rest of the paper is organized as follows: the "Proposed DLBP" section describes the proposed stochastic DLBP and its formulation. The "Proposed solution algorithm" section reports the improved NGO with stochastic simulation procedures. Experimental results and sensitivity analyses are performed in the "Experimental results" section. The "Conclusions, discussions, and future works" section concludes the paper and presents the limitations of the paper and future research directions.

Proposed DLBP

The DLBP is a complex optimization problem which is the process of orderly assigning disassembly tasks to a series of workstations to achieve various disassembly objectives while satisfying the constraints for the relationships between the disassembly tasks and the upper limit of the beat of each workstation (Feng et al., 2019; Henrioud et al., 1991; Hu et al., 2020). For example, Fig. 1 shows an automobile disassembly line including three workstations.

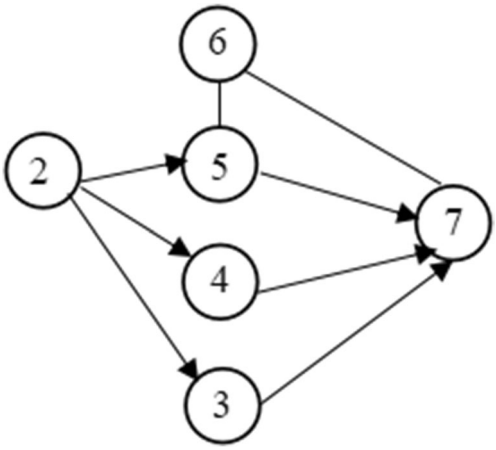

Fig. 2 A basic disassembly hybrid graph

Due to the complexity of disassembly, this paper uses a disassembly hybrid graph including a set of matrixes to describe the disassembly concept clearly (Hu et al. 2020; Issaoui et al. 2017). A disassembly hybrid graph can be represented by three factors including V, E, and DE in Eq. (1).

$$G = \{V, E, DE\} \tag{1}$$

where G denotes the disassembly hybrid matrix and V represents the product parts. If a product contains n parts, then $V = \{V_1, V_2, V_3, V_n\}$. E denotes the directed edges in the hybrid graph, represented by solid arrows, and DE denotes the undirected edges in the hybrid graph, represented by solid lines without arrows. To show an example for this graph, proposed basic disassembly hybrid graph is shown in Fig. 2.

In Fig. 2, the number of circles shows the number of tasks. The solid arrow is the directed edge E, which indicates that there is a direct priority disassembly constraint relationship between two tasks, such as task 2 and task 4 are connected by a solid arrow, then task 2 needs to be disassembled before task 4. The solid line without an arrow is the undirected edge DE, which indicates a direct contact relationship between

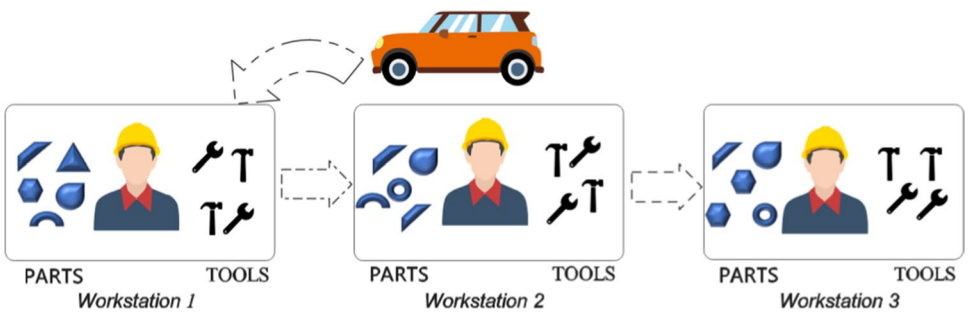

Fig. 1 An automobile disassembly line

the two tasks. As such, tasks 5 and 6 are connected by a solid line without arrows. Then, task 5 and task 6 are in a contacted relationship. According to the product disassembly hybrid diagram's directed edge E and undirected edge DE, the product disassembly priority constraint matrix P and direct contact matrix C are generated as follows:

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix} \quad C = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{bmatrix}$$

where

$$c_{ij} = \begin{cases} 1, & \text{part } i \text{ and } j \text{ are in direct contact;} \\ 0, & \text{part } i \text{ and } j \text{ are not in direct contact or } i = j; \end{cases}$$

$$p_{ij} = \begin{cases} 1, & \text{part } j \text{ must be disassembled before } i; \\ 0, & \text{else;} \end{cases}$$

When a part needs to be disassembled, it needs to be checked for disassembly based on the two matrices above, and when Eqs. (2) and (3) are satisfied, it meets the disassembly requirements.

$$\sum_{j=1}^{n} c_{ij} = 1 \tag{2}$$

$$\sum_{j=1}^{n} p_{ij} = 0 \tag{3}$$

where i denotes the parts for disassembly and n points to the total number of disassembly tasks.

Based on the above definitions and the disassembly rules, a random feasible disassembly sequence can be generated as follows: we first convert the disassembly mixture into P and C. Afterwards, the rows in P and C that are eligible for disassembly are found, and a random row is selected as the first task to be disassembled. Then, the row represented by this task is removed from P and C. Finally, the above steps are repeated until P and C are empty in order to generate a feasible disassembly sequence.

Combining the concept of disassembly and the characteristics of uncertain DLBP, the following mathematical model is established to deploy a stochastic DLBP in this paper. The notations for our formulation are summarized as follows:

Indices:

m: Index of disassembly tasks, $m = 1, 2, \ldots, M$

n: Index of workstations, $n = 1, 2, \ldots, N$

Parameters:

C_t: Cycle time of the disassembly line

t_m: Stochastic disassembly time for task m

t_t: Required time to change the disassembly tool

t_d: Required time to change the disassembly direction

e_m: Energy consumption for the process of task m

e_w: Energy consumption of workstation in the status of standby

g_m: Difficulty for removing a component in task m

Decision variables:

x_{mn}: 1, if task m is selected to be disassembled in workstation n, otherwise 0

y_m: if the disassembly tool of the mth task is different from the m-1th task; y_m=1, otherwise y_m=0. If task m is the first task of workstation n, then y_m=0

z_m: if the disassembly direction of the mth task is different from the m-1th task; z_m=1, otherwise z_m=0. If task m is the first task of workstation n, then z_m=0

The proposed stochastic DLBP formulation is as follows:

$$\min f_1 = E\left(\frac{\sum_{n=1}^{N} \left(Ct - \sum_{m=1}^{M} \left(t_m x_{mn} + t_t y_m x_{mn} + t_d z_m x_{mn} \right) \right)}{\sum_{n=1}^{N} Ct} \right) \tag{4}$$

$$\min f_2 = E\left(\sum_{n=1}^{N} \left(Ct - \sum_{m=1}^{M} \left(t_m x_{mn} + + t_t y_m x_{mn} + t_d z_m x_{mn} \right) \right)^2 \right) \tag{5}$$

$$\min f_3 = E\left(\sum_{n=1}^{N} \sum_{m=1}^{M} (1 + g_m) e_m t_m x_{mn} + e_w \sum_{n=1}^{N} \left(Ct - \sum_{m=1}^{M} t_m x_{mn} + t_t y_m x_{mn} + t_d z_m x_{mn} \right) \right) \tag{6}$$

s.t.

$$E\left(\frac{\sum_{n=1}^{N} \sum_{m=1}^{M} \left(t_m x_{mn} + t_t y_m x_{mn} + t_d z_m x_{mn} \right)}{Ct} \right) \leqslant N \leqslant M \tag{7}$$

$$E\left(\sum_{m=1}^{M} t_m x_{mn} + t_t y_m x_{mn} + t_d z_m x_{mn} \right) \leqslant Ct, \quad n = 1, 2, \cdots, N \tag{8}$$

$$\sum_{n=1}^{N} x_{mn} = 1, \quad m = 1, 2, \cdots, M \tag{9}$$

$$\sum_{j=1}^{M} c_{mj} = 1 \text{ and } \sum_{j=1}^{M} p_{mj} = 0 \tag{10}$$

$$x_{mn}, y_m, z_m = \{0, 1\} \tag{11}$$

The proposed model has three conflicting objectives as follows: Eq. (4) is the idle rate of the disassembly line considering the number of workstations. Equation (5) is the smoothness of the disassembly line, which is determined by the variance of the idle time. Equation (6) is the energy consumption for disassembly operations and standby workstations.

These objectives are limited by the constraint sets (7) to (10). Constraint set (7) indicates that the actual number of

workstations cannot exceed the maximum number of workstations. Constraint set (8) indicates that the working time within a workstation cannot exceed the cycle time. Constraint set (9) allows each disassembly task to be executed only once. Constraint set (10) ensures that each disassembly task satisfies the prerequisites. Finally, relation (11) are the definition of binary variables.

Proposed solution algorithm

As mentioned earlier, the basic version of DLBP is proven to be an NP-hard optimization problem (Jin et al. 2020; Kalayci et al. 2016) and our proposed stochastic DLBP using different objective functions is also known as an NP-hard problem. The main significant contribution of this paper is to propose an innovative solution algorithm, namely, improved NGO (INGO) using a stochastic simulation method for solving the proposed stochastic DLBP.

In the following, we first explain the main principles and inspirations of the NGO algorithm ("Main idea of NGO algorithm" section). Then, the multi-objective optimization framework for creating Pareto solutions is explained ("Multi-objective optimization procedure" section). The solution representation in our INGO is represented in the "Solution representation and the search space" section. Then, the search operators to find new solutions from the search space are explained in the "Search operators" section. Next, a stochastic simulation method is explained ("A stochastic simulation method" section). Finally, the final algorithmic framework is presented ("Final framework of the proposed INGO" section).

Main idea of NGO algorithm

The NGO algorithm proposed by Dehghani et al. (2021) simulates the hunting process of the northern goshawk, including two phases, i.e., (i) prey identification and attack and (ii) chase and escape. In the first phase, the northern goshawk randomly selects its prey and then quickly attacks it in a global search grade. The second phase is the chase and escape stage, which is a local search process in which the northern goshawk approaches the prey while trying to escape from the local optimum.

The NGO has certain superiority in its global search capability and can effectively deal with a variety of optimization problems (Dehghani et al. 2021). However, we need to customize and revise it for addressing complex optimization problems like our DLBP. The first issue is that the proposed DLBP is a multi-objective optimization problem. However, the NGO algorithm is suitable for single-objective problems. Hence, we need to extend the NGO for solving multi-objective optimization problems.

The second issue is that the search space of the NGO algorithm is continuous and cannot solve integer optimization problems. However, the feasible space of our DLBP is the sequence of tasks on each workstation, and therefore, we need a discrete search space for our NGO algorithm.

The last issue is that the original NGO is not able for solving stochastic optimization problems. For addressing the proposed stochastic DLBP, this study combines a stochastic simulation method. Based on these drawbacks of the NGO algorithm, this study proposes an INGO for solving the proposed stochastic DLBP with conflicting objectives.

Multi-objective optimization procedure

Since we have more than objective and our objectives usually have some conflicts, it is impossible to find a solution which is favorable for all the objectives (Tian et al. 2022a, c). Assume that *S1* and *S2* are two random solutions from the search space. Their values are respectively $\{f_1\ f_2\ f_3\}$ and $\{f'_1\ f'_2\ f'_3\}$. Solution *S1* dominates *S2*, if we have $\{f_1\ f_2\ f_3\} \preceq \{f'_1\ f'_2\ f'_3\}$ and $\{f_1\ f_2\ f_3\} \neq \{f'_1\ f'_2\ f'_3\}$. Otherwise, these solutions are Pareto solutions. In this regard, we classify all the solutions into different Pareto fronts, and the best front is our non-dominated solutions (Tian et al., 2022a, b, c).

To make these non-dominated solutions generated by the algorithm at each iteration, this paper sets up an external archive to preserve the non-dominated solutions and uses a crowding distance system to filter the non-dominated solutions within the external archive to maximize their solution quality. The calculation of the crowding distance system can be found in the literature (Zitzler et al. 2003).

Solution representation and the search space

Metaheuristics iteratively find a new set of solutions randomly from the search space (Fathollahi-Fard et al., 2018). We need to revise the NGO algorithm for exploiting and exploring discrete solutions from the search space which is defined by the disassembly sequence. For example, the solution string (1-3-5-7-8-9-10) in Fig. 3 represents the disassembly sequence

Fig. 3 Sequence of tasks for the solution representation

where we must assign this sequence of tasks to workstations to form a disassembly line balancing scheme.

This scheme consists of the following steps: (1) Input the obtained feasible disassembly sequence X, disassembly beat Ct, and turn on the first workstation FA, so that $FA = 1$, and the remaining assignable disassembly beat of the current workstation is ST, then $ST = Ct$. (2) Determine whether the time t_m of task m in sequence X is greater than ST, if yes, then $ST = Ct$ and turn on a new workstation with $FA = FA + 1$; otherwise, $ST = ST - t_m$. (3) Loop through step 2 until the tasks in sequence X are completely assigned to the workstations and output the disassembly solution. To help with understanding, it should be noted that the above process does not take into account the disassembly direction and tool change. However, in this study, it is necessary to consider the time required for these changes in the decoding process. If the total available workstation time is less than the working time required after accounting for the change time, then it is necessary to open the next workstation.

Search operators

To find new solutions at each iteration, we have redefined the search operators of NGO with new ones to consider the sequence of tasks while satisfying the constraints. In this regard, we have discrete recombination operators, procedures for the phase of prey identification and attack and the phase chase and escape.

Discrete recombination operators

Since the initial solutions are generated randomly, this study proposes initial solution discrete recombination operators to improve their quality. After a random sequence is generated, this work randomly combines all the disassembly sequence numbers to form $M/2$ combinatorial pairs, after which the pairs are selected to be crossed or muted according to Eq. (12).

$$\mu = (M/2) \times \delta \tag{12}$$

where μ is the number of pairs selected, M is the total number of disassembly tasks, $M/2$ is the number of combined pairs, and δ is the set parameter.

1) *Crossover operator*

Calculating μ according to Eq. (12), μ combinations of pairs are randomly selected for internal crossover, as shown in Fig. 4.

2) *Mutation operator*

Calculating μ according to Eq. (12), μ combinations of pairs are randomly selected for the mutation as shown in Fig. 4.

After we have improved the initial solutions, we must apply the concept of Pareto domination to define the Pareto fronts based on the crowding distance to find non-dominated solutions.

Phase of prey identification and attack

We consider new procedures for the phase of prey identification and attack. This paper names them T1 and T2, which are selected with an adaptively varying probability, where this selection probability is calculated as shown in Eq. (13)

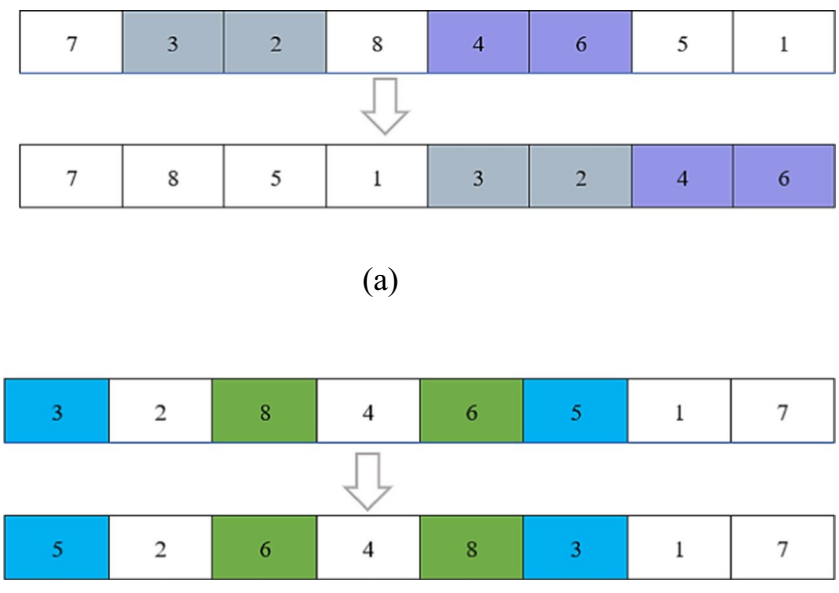

Fig. 4 Discrete recombination operators. (a) Crossover operator; (b) mutation operator

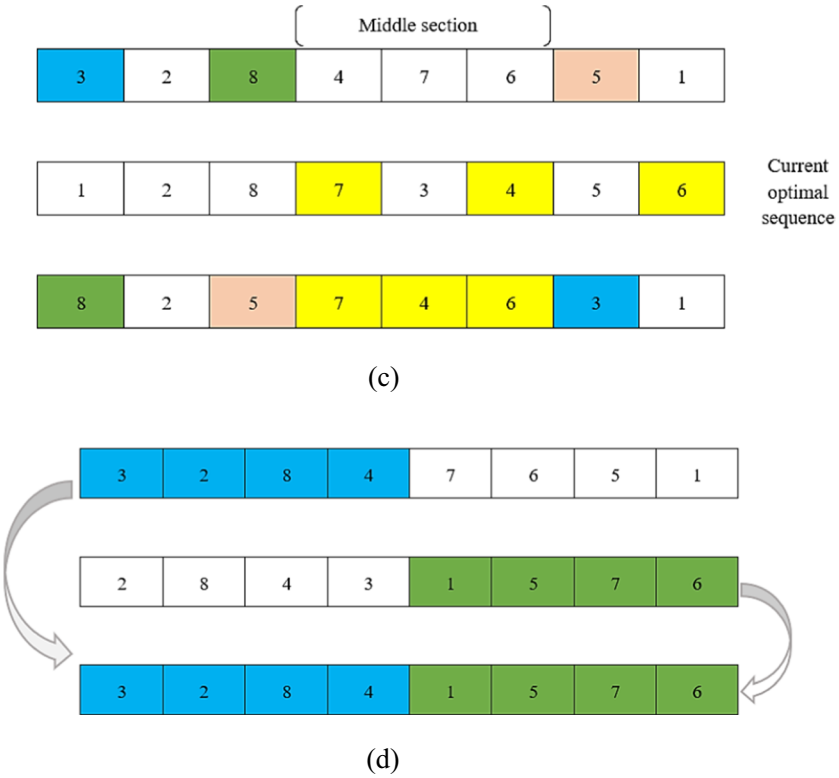

Fig. 5 T1 and T2 procedures. (c) T1 procedure; (d) T2 procedure

$$\chi = |\lambda - \lambda \times (Y/MaxIt)| \tag{13}$$

where χ is the probability of choosing the first method T1, λ is the set parameter, Y is the current number of iterations, $MaxIt$ is the maximum number of iterations, and the probability of choosing the second method T2 is $1-\chi$, if $\chi = 0$, then $\chi=\lambda$.

1) *T1 procedure*

The first task and two other random tasks in the disassembly sequence to be optimized are selected and divided into three parts, with the middle part being rearranged according to the order of disassembly in the optimal disassembly sequence selected in the previous step. In addition, the three selected tasks are randomly rearranged. If the sequence is not separated after the selection of the other two disassembly tasks or if there is only one disassembly task in the middle part, the two disassembly tasks are reselected, as depicted in Fig 5.

2) *T2 procedure*

A part of the disassembly sequence to be optimized is selected, another sequence is randomly found to select the remaining part, and these two parts are combined into a new disassembly sequence, as portrayed in Fig. 5.

Phase of chase and escape

In this paper, a local search strategy is used to optimize the original NGO for reformulating the phase of chase and escape. A random number H is first generated, after which H sequences are randomly selected for search optimization. If $H = 6$, then 6 disassembly sequences are randomly selected for search optimization. The search method is to

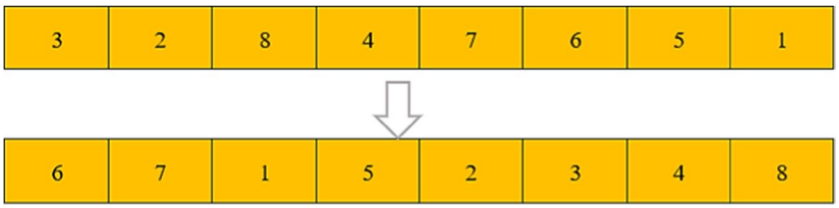

Fig. 6 Local search strategy

generate the corresponding sequence of the original disassembly sequence, i.e., the number of disassembly tasks plus one minus the task number of the original sequence; Fig. 6 represents the corresponding sequence generation method for a disassembly sequence with a total task number of 8.

A stochastic simulation method

In this work, a stochastic simulation method is used to calculate the objective function values for the optimized disassembly solution. Stochastic simulations are also known as Monte Carlo methods or statistical testing methods (Li et al. 2002; Liu et al. 2020; Liu et al., 2021a, b). The calculation method is based on probability and statistical theory (Liu et al., 2021a, b; Sun et al. 2021; Fathollahi-Fard et al. 2022). In the model constructed in this paper, the objective functions given in Eqs. (4) to (6) contain random variables for the disassembly time. Therefore, the disassembly time of each task is used as a sample for a stochastic simulation method in this paper, and the disassembly time of each task is formed according to a uniform distribution, and the average value is taken as the objective function value after a large number of repeated calculations. The procedure for the stochastic simulation method is shown below:

Step 1: Initialize the disassembly sequence and the number of repetitions l.

Step 2: Generate random simulation samples, i.e., the disassembly task time t_m.

Step 3: Decode the disassembly sequence based on the generated samples, check the constraints, and perform the calculation of the objective function values.

Step 4: Perform l repetitions of the second and third steps.

Step 5: Return the average objective function values of l samples to obtain the desired objective function values.

Final framework of the proposed INGO

INGO uses the maximum number of iterations of the algorithm, $MaxIt$, as the termination condition, the current iteration number, Y, as the flag, and initially $Y = 1$, the algorithm stops when $Y \geq MaxIt$, the main idea of which is described above. In conclusion, the procedures of the improved NGO algorithm are as follows:

Step 0: Set the maximum number of iterations $MaxIt$, the population size P, the external archive capacity NEA, the number parameter δ for calculating the selection of combinatorial pairs, and the probability parameter λ for calculating the selection of T1, the number of replications of the stochastic simulation method l, and other problem parameters.

Step 1: Generate the initial disassembly scheme based on the disassembly sequence generation method proposed in the "Proposed DLBP" section and the encoding and decoding methods in the "Solution representation and the search space section".

Step 2: Apply the proposed operators' computation rules to optimize the initial disassembly scheme. The optimal disassembly scheme is also selected based on Pareto domination and crowding distance calculation.

Step 3: Apply the rules for the prey identification and attack phase, and update the disassembly sequence again using the optimal disassembly scheme in the current stage.

Step 4: Apply the rules of the chase and escape phase to perform a local search for the solution.

Step 5: Perform Pareto solution calculation on the solution after the local search is completed, and save the non-dominated solution to the external archive, or filter the external archive based on the crowding distance if the external archive is full.

Step 6: Perform crowding distance calculation on the solutions in the external archive, and update the optimal disassembly solution.

Step 7: Determine whether the maximum number of iterations is reached, i.e., whether $Y \geq MaxIt$, and if the main INGO algorithm is terminated, output the set of non-dominated solutions. Otherwise, go to step 3.

Note that all objective function values are calculated in the INGO step using the stochastic simulation method presented in the "A stochastic simulation method" section as a benchmark). Figure 7 depicts the flow chart of INGO.

Experimental results

To test the feasibility of the model and evaluate the performance of the INGO, this section presents different numerical examples as two case studies for analyzing both small and large sizes. Finally, the proposed algorithm is compared with successful recent and traditional algorithms from the literature. The experimental program of the proposed algorithm was developed using MATLAB R2020b on a computer with an Intel(R) Core (TM) i5-7400 CPU @ 3.00.

Small-scale case study

In this section, a worm reducer with 25 parts (P25) from the literature (Zhang et al. 2021; Yang et al. 2019a; Tian et al., 2022a, b, c; Zhou and Bian, 2022; Tian et al. 2019) is introduced as an example analysis to evaluate the performance of the algorithm and model under small to medium-scale disassembly operations. The worm reducer disassembly time interval is given in the Appendix.

Since our metaheuristic algorithm has a set of parameters, we must first tune them to improve the quality and

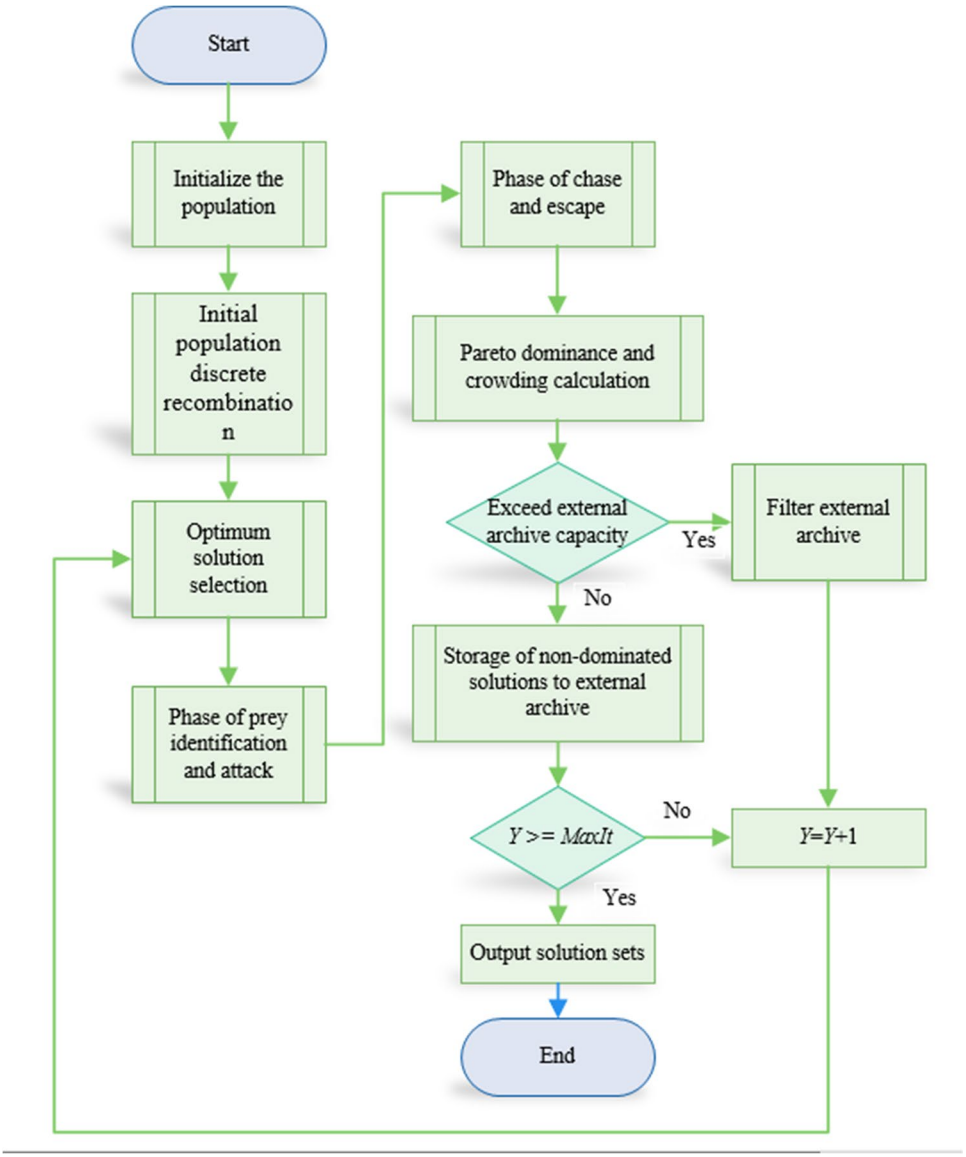

Fig. 7 INGO flow chart

efficiency of their solutions (Li et al. 2002; Liu et al. 2020; Liu et al., 2021a, b). Without a calibration of parameters, our metaheuristic algorithm is not efficient (Sun et al. 2021; Fathollahi-Fard et al. 2022; Wang et al., 2019a, b). Our INGO has four parameters, respectively, the maximum number of iterations *MaxIt*, the population size P, the parameter δ for calculating the selection of combinatorial pairs, and the probability parameter λ for calculating the selection of T1 procedure. Based on our experience and tests, we have considered four candidate values for each parameter. These values are the levels as reported in Table 1.

Table 1 List of parameters for our INGO algorithm

Parameter	Levels			
	1	2	3	4
MaxIt	20	30	40	50
P	5	15	30	50
δ	0.2	0.4	0.6	0.8
λ	0.2	0.4	0.6	0.8

Based on all cases from Table 1, there are $3^4 = 256$ tests. To save our time, we apply the Taguchi method (Taguchi et al. 2002) instead of full factorial method. This method suggests a set of orthogonal arrays to use a number of selected tests instead of all ones. For example, based on the levels and the number of parameters reported in Table 2, the Taguchi method suggests L_{16} including 16 selected tests from 256 tests for analyzing the calibration of our metaheuristic algorithm.

Table 2 Experiments of calibration for the proposed INGO using the orthogonal array *L16*

Number of tests	Parameters and their levels				RPD
	MaxIt	*P*	δ	λ	
1	1	1	1	1	0.198965
2	1	2	2	2	0.168234
3	1	3	3	3	0.136852
4	1	4	4	4	0.123682
5	2	1	2	3	0.164523
6	2	2	1	4	0.123256
7	2	3	4	1	0.143653
8	2	4	3	2	0.116258
9	3	1	3	4	0.102516
10	3	2	4	3	0
11	3	3	1	2	0.083523
12	3	4	2	1	0.072516
13	4	1	4	2	0.118653
14	4	2	3	1	0.086352
15	4	3	2	4	0.089175
16	4	4	1	3	0.087986

The Taguchi method uses relative percentage deviation (RPD) to evaluate the algorithm parameters. To apply this tool, we have used the sum of the average objective function values for the RPD calculation, as shown in Eq. (14).

$$\mathrm{MH} = \frac{\sum \left(f_{r1} + f_{r2} + f_{r3}\right)^{*}}{\mathrm{NEA}} \varpi \; r = 1, 2, \ldots, \mathrm{NEA} \tag{14}$$

where MH is the sum of the mean objective function values, f_{r1}, f_{r2}, and f_{r3} are the objective function values, NEA is the number of external archives, and ϖ is a factor based on the program's running time.

Ultimately, the RPD is calculated as shown in Eq. (15).

$$\mathrm{RPD} = \frac{C_{\mathrm{MH}} - \mathrm{Min}_{\mathrm{MH}}}{\mathrm{Min}_{\mathrm{MH}}} \tag{15}$$

where C_{MH} is the MH of the solution obtained with the current parameter and $\mathrm{Min}_{\mathrm{MH}}$ is the minimum MH of all solutions obtained with each parameter

According to the results in Table 2 for the RPD and Fig. 8 for the average RPD, after 16 tests, the optimal parameters for this case are *MaxIt* = 30, *P*= 50, $\delta = 0.8$, $\lambda = 0.6$, NEA = 10. In addition, it should be noted that we set the time required to change the disassembly tool (t_t) = 4 s, the time required to change the disassembly direction (t_d) = 8s, the unit time energy consumption of task $m(e_m) = 0.8$, the unit time energy consumption of workstation standby (e_w) = 0.2, $l = 1000$, and $C_t = 120$. It should be noted that for l, too large an l would lead to a significant increase in algorithm running time, while a smaller l would not achieve the stochastic simulation effect; therefore, after literature analysis (Guo et al. 2020), l was set to 1000.

After solving the small-scale case study using the calibrated INGO, the non-dominated solutions are reported in Table 3. It should be noted that to make our results more reliable, we run the algorithm 10 times to make the results in Table 3.

Large-scale case study

To approve that our proposed INGO can solve large-scale examples efficiently, a large case study including 52 tasks

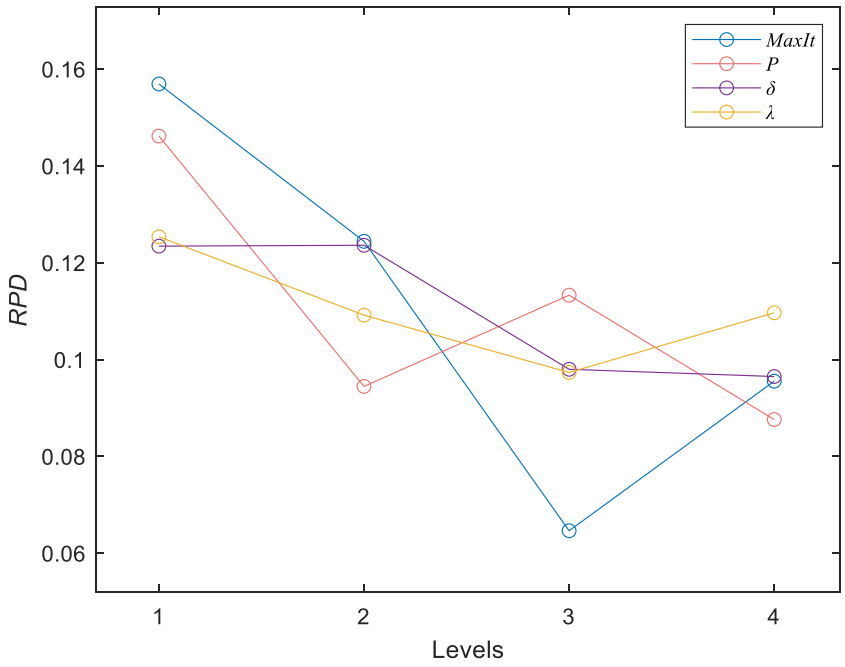

Fig. 8 Behavior of the INGO in the term of RPD metric

Table 3 Pareto optimal solution set for small-scale case study

Order	Line balance schemes	f_1	f_2	f_3
1	[2,25,24,14]→[4,23,13,15,21]→[5,12,16,19,6,17]→[11,18,3,22]→[10,7,20,9,8]	0.1364	2323	335.45
2	[2,4,5,15]→[6,7,16,17,18,19,14]→[25,24,23,22,13,12,21]→[20,3,11,10,8,9]	0.0856	2066	369.24
3	[25,2,24,14]→[4,23,5,22,21]→[15,6,16,7,17]→[18,13,12,11,19]→[10,8,20,9,3]	0.1687	2725	334.22
4	[25,14,24,2]→[13,21,4,3,12,23]→[5,15,6,11]→[10,16,22,19,9]→[7,17,8,18,20]	0.1418	1953	362.19
5	[2,25,4,24,21]→[15,16,19,5,6,23,22]→[7,14,13,17,18,20]→[12,3,11,10,8,9]	0.0486	381	375.06
6	[2,14,25,24,23]→[15,16,17,18,21,22]→[13,12,11,10,9,20,19]→[4,5,6,7,3,8]	0.1100	779	367.71
7	[25,14,13,12,24,21]→[2,11,3,10,23]→[9,22,4,5,6,7]→[8,15,16,19,17,18,20]	0.0981	734	372.79
8	[15,4,5,6,2]→[14,16,7,17,19,18]→[25,24,23,22,13,12,21]→[20,3,11,10,8,9]	0.0621	1994	369.46
9	[15,2,16,14]→[19,4,17,18,5,6]→[13,3,12,11,10,9,7]→[8,25,24,23,21,22,20]	0.0524	196	373.52
10	[4,25,24,15]→[2,14,16,21,13]→[17,5,3,23,6,19]→[7,12,18,22,20]→[11,10,8,9]	0.1548	2101	350.26

is provided from the literature (Tian et al., 2022c; Zhou et al. 2022; Tian et al. 2019). The information on the disassembly time interval is reported in the Appendix. In a same way as done for the small-scale case study, we must tune the parameters of INGO where the results of calibration are reported in Fig. 9. The final parameters are set to $C_t = 600$, $MaxIt = 50$, $P = 50$, $\delta = 0.6$, $\lambda = 0.4$, and the other parameters are the same as in the previous section. It should be noted that the disassembly tools and disassembly directions for this example were generated randomly.

After solving the case study based on the calibrated INGO, our non-dominated solutions are reported in Table 4.

Comparison with other algorithms

Due to the high complexity of our case studies in both small and large scales, no exact solver solves these instances. In this regard, we have considered two well-known algorithms, namely, non-dominated sorting genetic algorithm (NSGA-II) (Tian et al. 2019) and SEO (Fathollahi-Fard et al. 2018) to compare them with our proposed INGO. These two algorithms have been successfully developed and applied to the DLBP by scholars, while showing good solution accuracy and superior performance. The parameters of NSGA-II and SEO were set by the literature (Zhang et al. 2021; Tian et al., 2022a, b, c; Tian et al. 2019).

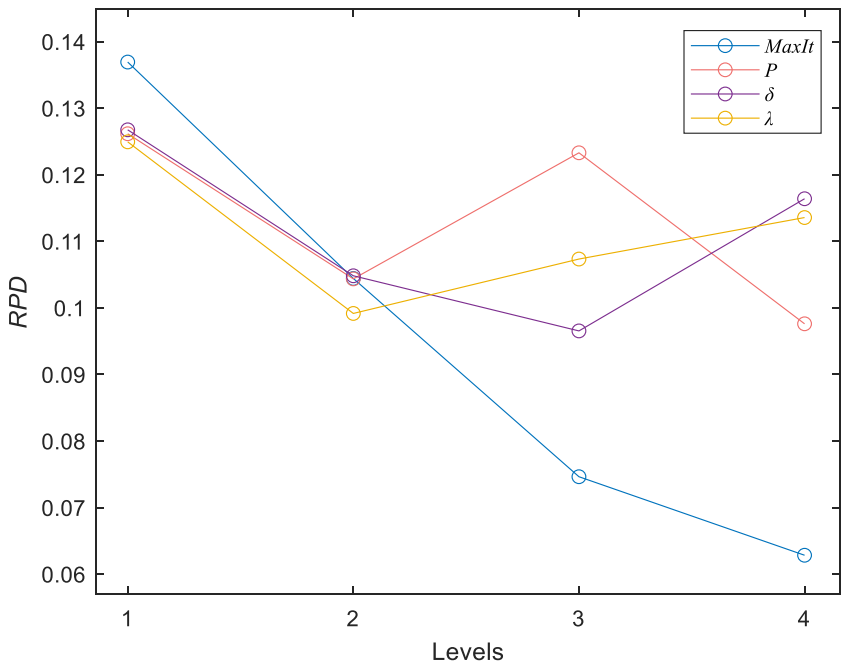

Fig. 9 Behavior of the INGO in the term of RPD metric

Table 4 Non-dominated solutions for our large-scale case study

Order	Line balance schemes	f_1	f_2	f_3
1	[28,26,33,18,42]→[4,31,15,27,25,52]→[21,29,19,49,45,32,36]→[47,3,46,30,1,16]→[51,43,37,44,2,12,14]→[34,35,39,40,41,22,9,7]→[24,8,13,50,11,48,20,23,38]→[10,6,5，17]	0.0930	30291	4218.6
2	[37,26,39,28,21,18]→[4,36,42,29,33,14]→[3,2,15,11,47,32,1,16]→[31,12,25,34]→[22,9,35,27,23]→[49,45,52,46,51,38,40,24,19,50]→[30,43,7,41,8,13,20,17]→[10,44,48,5,6]	0.0845	23052	4324.5
3	[28,26,33,18,42]→[4,31,15,27,25,52]→[21,36,32,47,3,46,2]→[43,49,37,16,51,30,40,50,45]→[44,41,39,19,1,12,48]→[29,14,11,35,24,8,9]→[23,34,38,22,7,17,13,20]→[10,5,6]	0.0898	37360	4206.9
4	[18,37,28,42,21,4]→[36,33,26,3]→[31,52,39,32,16,1,15,27]→[12,35,47,9,2]→[29,25,45,46,44,24,19]→[49,34,43,51,40,30,22]→[50,7,41,48,8,13,10,5]→[6,20,14,11,23,38,17]	0.0831	23738	4324.7
5	[28,26,33,18]→[42,4,31,15,27,25]→[52,21,36,32,47,3,46]→[30,1,29,43,2,14]→[11,16,45,12,35,9]→[49,37,34,19,39,23,38,51,44]→[24,8,22,7,13,20,40,17,41]→[10,50,6,5,48]	0.0833	24796	4250.0
6	[29,33,21,37,26]→[32,18,36,3,42,28]→[4,16,14,47,39,1,12]→[31,2,11,9,46,52,34,45,44]→[15,43,22,27,25,19]→[7,30,49,51,35,40,41,23]→[50,24,38,48,17,8,13,20,10]→[6,5]	0.0842	51743	4195.1
7	[4,26,33,42,29]→[37,28,31,25,39,36,45]→[3,1,14,9,2]→[47,12,15,11,23,27,19]→[34,46,22,43,52,35,44,30]→[7,49,38,24,51,40,50,48,41,8]→[18,13,20,32,16,21,17]→[10,5,6]	0.0857	36471	4230.9
8	[36,29,26,4]→[33,42,18,37,3,21]→[14,2,32,16,47,28,39,1,31]→[52,46,30,11,43,25,49]→[45,15,23,38,12,9]→[44,35,34,24,8,51,22,7,17]→[13,27,10,5,20]→[40,6,41,19,50,48]	0.0905	61389	4099.5
9	[18,28,37,26,21]→[42,36,15,32,39,33,47,46]→[43,29,14,16,4,11,31,23,27]→[3,25,38,19,30]→[44,2,17,45,1,52,12]→[34,49,22,9,7,35]→[51,40,41,24,50,48,8,13,20]→[10,5,6]	0.0858	33468	4226.4
10	[18,36,28,42,21,29]→[14,32,26,33,3]→[47,37,15,11,27,23,46,39,43,19]→[44,2,1,16,4,38,31]→[9,45,52,17,25]→[49,30,12,51,35]→[40,24,41,34,22,7,50,48,8,13,10]→[20,6,5]	0.0775	23740	4417.1

To evaluate the solutions obtained by these algorithms, this work considers the enhanced inverse generation distance (IGD-NS) and the hypervolume (HV) metrics as two popular metrics for solving multi-objective optimization problems (Fathollahi-Fard et al. 2021; Zitzler et al. 2003; Tian et al. 2019). Lower IGD-NS values reflect the better performance of the algorithms while higher HV values reflect better convergence and distribution of non-dominated solutions. It should be noted that the individual algorithms do not differ significantly in time, as the results of the solution times will not be compared in this paper.

Due to the random nature of these intelligent algorithms (Yang et al., 2019a, b; Fathollahi-Fard et al. 2021; Yuan et al. 2020), this work runs each algorithm ten times and takes its average value, the final solution results are shown in Table 5, and the best values for each metric in each test problem are shown in bold. According to the results in Table 5, the proposed INGO is generally superior in comparison with other intelligent algorithms for different-size instances.

To analyze these algorithms statistically, box plots based on the analysis of variance for both small and large scales are drawn respectively in Fig. 10. For the small-scale case study (Fig. 10e), SEO demonstrated good results on the IGD-NS metric and the proposed INGO achieved optimal values for the HV metric. Based on the results for the large-scale case (Fig. 10f), the proposed IGNO achieved better values for both IGD-NS and HV. Generally, both plots confirm that the proposed INGO has certain robustness and accuracy in comparison with NSGA-II and SEO.

INGO sensitivity analysis

Here, this study analyzes the effectiveness of the discrete recombination operators in INGO, and the HV is used as a measure, the larger the indicator, the better the

Table 5 Average results of algorithms based on evaluation metrics

Small-scale case study	INGO		NSGA-II		SEO	
	IGD-NS	HV	IGD-NS	HV	IGD-NS	HV
	68.83	**2710.68**	70.05	2621.35	**67.20**	2702.03
Large-scale case study	INGO		NSGA-II		SEO	
	IGD-NS	HV	IGD-NS	HV	IGD-NS	HV
	384.09	**1048.83**	400.50	890.12	396.68	993.77

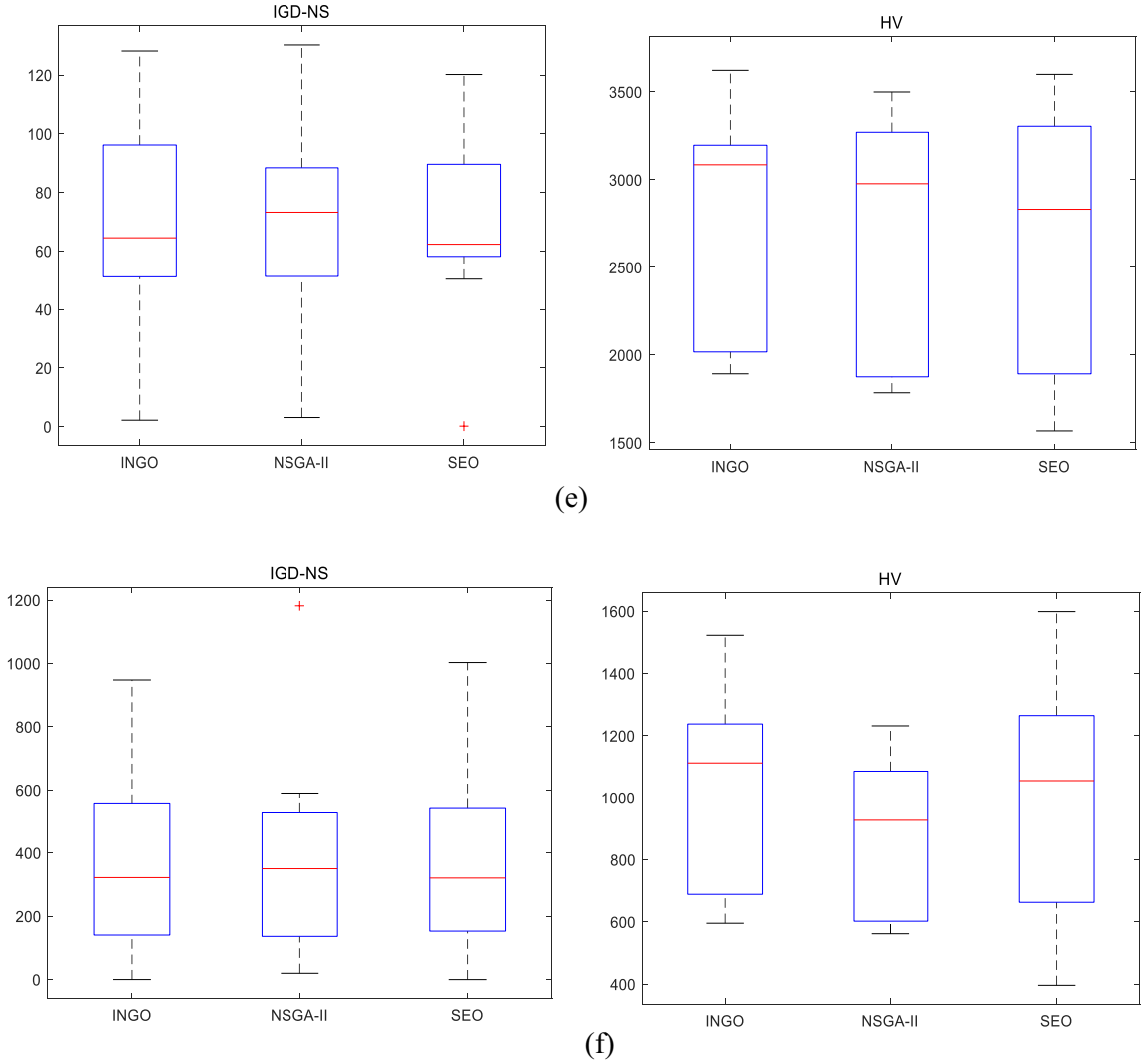

Fig. 10 Box plot for the comparison of algorithms in the two cases

performance of the algorithm, and the average results of the program after fifteen runs are shown in Table 6. Note that the population size P is set to 30 and 50, and the program runtime is limited to 1500 s.

According to the results in Table 6, INGO outperforms NGO in HV metrics for different population sizes and different size cases, proving its comprehensive performance superior to NGO and the effectiveness of the initial population discrete recombination operators.

Table 6 Average results of algorithms based on HV

Small-scale case study	HV	
	$P = 30$	$P = 50$
INGO	1568.35	1868.76
NGO	1322.56	1603.53
Large-scale case study	HV	
	$P = 30$	$P = 50$
INGO	898.62	986.86
NGO	803.25	889.58

Conclusions, discussions, and future works

For the recovery process of EOL products, the disassembly is a very important part, and this fact highlights the importance of the DLBP as a complex optimization problem. According to the literature on DLBP (Yang et al., 2019a, b; Fathollahi-Fard et al. 2021; Yuan et al. 2020; Zhang et al. 2017; Zheng et al. 2018; Zhu and Zhou, 2021; Zeng et al. 2022; Zhang et al. 2023), most of the current studies on the DLBP focus on the cost and efficiency perspectives, and a few studies are contributing to the uncertainty in the disassembly line. To fill this

research gap, this paper considered the impact of uncertainty in the disassembly operations and developed a stochastic SDLBP for minimizing the disassembly line idleness in the case of random disassembly operation time rate, smoothness, and energy consumption.

In addition to the development of a stochastic multi-objective DLBP, this study for the first time proposed a new extension of NGO as a multi-objective discrete INGO incorporating stochastic simulation methods to solve our optimization problem. To show the applicability of this research, both small and large numerical examples were solved, and the proposed algorithm was calibrated to improve its performance. Due to the Pareto optimization, it is difficult to achieve simultaneous optimization of the smoothness of the disassembly line, the idle rate, and energy consumption. The proposed INGO can directly achieve a good balance between these three goals and bring a better disassembly solution. Generally, this paper provided a new idea for solving the uncertain DLBP which can effectively improve the quality and efficiency of disassembly lines while promoting the development of green remanufacturing.

From our results, it can be observed that changes in disassembly tools and directions have a certain impact on the allocation of disassembly tasks. Taking them into account can arrange disassembly operations more reasonably. During the operation arrangement process, changes in direction and tools must be minimized, bringing unnecessary additional energy consumption. In addition, incorporating random disassembly operation time can minimize the uncertainty in the operation process and bring more space for decision-makers to choose disassembly solutions. Finally, the proposed stochastic DLBP allows decision-makers to better consider how the disassembly lines of different tasks can be better coordinated with the natural environment while minimizing their decision-making costs.

Although this paper presented a significant contribution to the literature on DLBP with the development of a stochastic multi-objective model and an improved metaheuristic algorithm, there are still many suggestions to extend this paper in the future. For example, more expressions for uncertain disassembly time, research on multi-product stochastic mixed flow disassembly, and combining big data are some suggestions to extend our DLBP formulation (Tian et al. 2023). At last but not least, the proposed algorithm can be improved by destruction and construction heuristics as well as adaptive memory search for solving the DLBP.

Appendix

See Table 7.

Table 7 Disassembly time interval for two case studies

Small-scale case			
Order	Time/s	Order	Time/s
1	/	14	U (24, 26)
2	U (4, 6)	15	U (24, 26)
3	U (24, 26)	16	U (7, 9)
4	U (9, 11)	17	U (5, 7)
5	U (7, 9)	18	U (14, 16)
6	U (14, 16)	19	U (4, 6)
7	U (7, 9)	20	U (7, 9)
8	U (7, 9)	21	U (4, 6)
9	U (29, 31)	22	U (14, 16)
10	U (14, 16)	23	U (5, 7)
11	U (7, 9)	24	U (7, 9)
12	U (9, 11)	25	U (24, 26)
13	U (17, 19)		
Large-scale case			
Order	Time/s	Order	Time/s
1	U (169.95, 175.95)	27	U (84.54, 89.54)
2	U (42.32, 48.32)	28	U (63.49, 67.49)
3	U (82.02, 88.02)	29	U (70.61, 75.61)
4	U (60.47, 66.47)	30	U (150.68, 156.68)
5	U (232.34, 238.34)	31	U (68.92, 74.92)
6	U (142.63, 148.63)	32	U (69.34, 72.34)
7	U (29.44, 35.44)	33	U (119.08, 123.08)
8	U (29.44, 35.44)	34	U (59.98, 64.98)
9	U (198.01, 212.01)	35	U (70.23, 73.23)
10	U (52.28, 58.28)	36	U (110.68, 113.65)
11	U (25.49, 32.49)	37	U (54.32, 58.32)
12	U (182.32, 188.32)	38	U (32.63, 37.63)
13	U (22.41, 28.41)	39	U (7.14, 10.14)
14	U (15.02, 20.02)	40	U (21.23, 24.23)
15	U (22.63, 27.63)	41	U (12.25, 16.25)
16	U (7.22, 13.22)	42	U (84.32, 87.32)
17	U (49.68, 55.68)	43	U (52.61, 55.61)
18	U (84.35, 90.35)	44	U (32.68, 37.68)
19	U (62.38, 69.38)	45	U (12.04, 16.04)
20	U (120.24, 125.24)	46	U (19.46, 23.46)
21	U (103.05, 110.05)	47	U (42.37, 46.37)
22	U (71.85, 77.85)	48	U (33.91, 37.91)
23	U (83.74, 89.74)	49	U (52.73, 55.73)
24	U (73.85, 77.85)	50	U (47.05, 50.05)
25	U (192.64, 196.64)	51	U (77.69, 79.69)
26	U (198.57, 203.57)	52	U (24.58, 27.58)

Author contribution Guangdong Tian: conceptualization, methodology, funding acquisition, resources, supervision, validation, and writing—original draft and editing. Xuesong Zhang and Amir M. Fathollahi-Fard: conceptualization, methodology, formal analysis, and writing—original draft. Zhigang Jiang and Chaoyong Zhang: conceptualization, supervision, methodology, and validation. Gang Yuan and Duc Truong Pham: conceptualization and supervision.

Data availability Not applicable.

Declarations

Ethical approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

Competing interests The authors declare no competing interests.

References

Altekin FT (2017) A comparison of piecewise linear programming formulations for stochastic disassembly line balancing. *Int J Prod Res* 55(24):7412–7434. https://doi.org/10.1080/00207543.2017.1351639

Avikal S, Jain R, Mishra PK (2014b) A Kano model, AHP and M-TOPSIS method-based technique for disassembly line balancing under fuzzy environment. *Appl Soft Comput* 25:519–529. https://doi.org/10.1016/j.asoc.2014.08.002

Avikal S, Mishra PK, Jain R (2014a) A Fuzzy AHP and PROMETHEE method-based heuristic for disassembly line balancing problems. *Int J Prod Res* 52(5):1306–1317. https://doi.org/10.1080/00207543.2013.831999

Bentaha ML, Voisin A, Marangé P (2020) A decision tool for disassembly process planning under end-of-life product quality. *Int J Prod Econ* 219:386–401. https://doi.org/10.1016/j.ijpe.2019.07.015

Chen W, Geng Y, Dong H, Tian X, Zhong S, Wu Q et al (2018) An emergy accounting based regional sustainability evaluation: a case of Qinghai in China. *Ecol Indic 88*:152–160

Dehghani M, Hubálovský Š, Trojovský P (2021) Northern goshawk optimization: a new swarm-based algorithm for solving optimization problems. *IEEE Access 9*:162059–162080

Demello LSH, Sanderson AC (1990) AND/OR graph representation of assembly plans. *IEEE Trans Robot Autom* 6(2):188–199. https://doi.org/10.1109/70.54734

Fathollahi-Fard AM, Hajiaghaei-Keshteli M, Tavakkoli-Moghaddam R (2018) The social engineering optimizer (SEO). *Eng Appl Artif Intell* 72:267–293. https://doi.org/10.1016/j.engappai.2018.04.009

Fathollahi-Fard AM, Hajiaghaei-Keshteli M, Tavakkoli-Moghaddam R, Smith NR (2021) Bi-level programming for home health care supply chain considering outsourcing. *Journal of Industrial Information. Integration* 100246

Fathollahi-Fard AM, Ahmadi A, Karimi B (2022) Sustainable and robust home healthcare logistics: a response to the COVID-19 pandemic. *Symmetry 14*:193

Feng YX, Gao YC, Tian GD, Li ZW, Hu HS, Zheng H (2019) Flexible process planning and end-of-life decision-making for product recovery optimization based on hybrid disassembly. *IEEE Trans Autom Sci Eng* 16(1):311–326. https://doi.org/10.1109/TASE.2018.2840348

Guo L, Zhang XF (2020) Remanufacturing parallel disassembly sequence planning method driven by multiple failures. *Journal of Zhejiang University Engineering Science* 54(11):2233–2246. https://doi.org/10.3785/j.issn.1008-973X.2020.11.019

Guo L, Zhang Z, Zhang X (2023) Human–robot collaborative partial destruction disassembly sequence planning method for end-of-life product driven by multi-failures. Adv Eng Inform 55:101821

He J, Chu F, Dolgui A, Zheng F, Liu M (2022) Integrated stochastic disassembly line balancing and planning problem with machine specificity. *Int J Prod Res* 60(5):1688–1708

Henrioud JM, Bourjault A (1991) LEGA: a computer-aided generator of assembly plans. In: *Computer-aided mechanical assembly planning*. Springer US, Boston, MA, pp 191–215. https://doi.org/10.1007/978-1-4615-4038-0_8

Hu ZB, He DF, Song W, Feng K (2020) Model and algorithm for planning hot-rolled batch processing under time-of-use electricity pricing. *Processes* 8(1):42. https://doi.org/10.3390/pr8010042

Issaoui L, Aifaoui N, Benamara A (2017) A model of mobility state of parts, the automation of feasibility test in disassembly sequence generation. *Proceedings of the Institution of Mechanical Engineers Part C-Journal of Mechanical Engineering* 231(20):3702–3714. https://doi.org/10.1177/0954406216654196

Jiao QL, Xu D, Li C (2016) Product disassembly sequence planning based on flower pollination algorithm. *Comput Integr Manuf Syst* 22(12):2791–2799. https://doi.org/10.13196/j.cims.2016.12.007

Jin LL, Zhang CY, Fei XJ (2020) Realizing energy savings in integrated process planning and scheduling. *Processes* 7(3):120. https://doi.org/10.3390/pr7030120

Kalayci CB, Polat O, Gupta SM (2016) A hybrid genetic algorithm for sequence-dependent disassembly line balancing problem. *Ann Oper Res* 242(2):321–354. https://doi.org/10.1007/s10479-014-1641-3

Kizilay D (2022) A novel constraint programming and simulated annealing for disassembly line balancing problem with AND/OR precedence and sequence dependent setup times. *Comput Oper Res* 146:105915

Li JR, Khoo LP, Tor SB (2002) A novel representation scheme for disassembly sequence planning. *Int J Adv Manuf Technol* 20(8):621–630. https://doi.org/10.1007/s001700200199

Liu M, Liu X, Chu F, Zheng FF, Chu CB (2020) Robust disassembly line balancing with ambiguous task processing times. *Int J Prod Res* 58(19):5806–5835. https://doi.org/10.1080/00207543.2019.1659520

Liu M, Liu X, Chu F, Zheng FF, Chu CB (2021a) An exact method for disassembly line balancing problem with limited distributional information. *Int J Prod Res* 59(3):665–682. https://doi.org/10.1080/00207543.2019.1704092

Liu X, Chu F, Zheng FF, Chu CB, Liu M (2021b) Distributionally robust and risk-averse optimisation for the stochastic multi-product disassembly line balancing problem with workforce assignment. *International Journal of Production Research Advance online publication*. https://doi.org/10.1080/00207543.2021.1881648

Mojtahedi M, Fathollahi-Fard AM, Tavakkoli-Moghaddam R, Newton S (2021) Sustainable vehicle routing problem for coordinated solid waste management. *J Ind Inf Integr 23*:100220

Ren YP, Yu DY, Zhang CY, Tian GD, Meng LL, Zhou XQ (2017) An improved gravitational search algorithm for profit-oriented partial disassembly line balancing problem. Int J Prod Res 55(24):7302–7316. https://doi.org/10.1080/00207543.2017.1341066

Ren Y, Zhang C, Zhao F, Xiao H, Tian G (2018) An asynchronous parallel disassembly planning based on genetic algorithm. Eur J Oper Res 269(2):647–660

Sun Y, Ning Z, Lodewijks G (2021) An autonomous vehicle interference-free scheduling approach on bidirectional paths in a robotic mobile fulfillment system. *Expert Syst Appl* 178:114932. https://doi.org/10.1016/j.eswa.2021.114932

Taguchi G, Jugulum R (2002) *The Mahalanobis-Taguchi strategy: a pattern technology system*. John Wiley & Sons

Tian GD, Ren YP, Feng YX, Zhou MC, Zhang HH, Tian JR (2019) Modeling and planning for dual-objective selective disassembly using AND/OR graph and discrete artificial bee colony. *IEEE Trans Ind Inform* 15:2456–2468. https://doi.org/10.1109/TII.2018.2884845

Tian G, Fathollahi-Fard AM, Ren Y, Li Z, Jiang X (2022a) Multi-objective scheduling of priority-based rescue vehicles to extinguish forest fires using a multi-objective discrete gravitational search algorithm. *Inf Sci 608*:578–596

Tian G, Yuan G, Aleksandrov A, Zhang T, Li Z, Fathollahi-Fard AM, Ivanov M (2022b) Recycling of spent lithium-ion batteries: a comprehensive review for identification of main challenges and future research trends. *Sustainable Energy Technologies and Assessments 53*:102447

Tian G, Zhang C, Fathollahi-Fard AM, Li Z, Zhang C, Jiang Z (2022c) An enhanced social engineering optimizer for solving an energy-efficient disassembly line balancing problem based on bucket brigades and cloud theory. *IEEE Transactions on Industrial Informatics*. https://doi.org/10.1109/TII.2022.3193866

Tian G, Lu W, Zhang X, Zhan M, Dulebenets MA, Aleksandrov A, ... Ivanov M (2023) A survey of multi-criteria decision-making techniques for green logistics and low-carbon transportation systems. Environ Sci Pollut Res 1–23

Wang WY, Mo DY, Wang Y, Tseng MM (2019a) Assessing the cost structure of component reuse in a product family for remanufacturing. *J Intell Manuf* 30(2):575–587. https://doi.org/10.1007/s10845-016-1267-1

Wang X, Du ZZ, Zhang YL, Wang JD, Wang JH, Sun W (2019b) Optimization of distillation sequences with nonsharp separation columns. *Processes* 7(6):323. https://doi.org/10.3390/pr7060323

Wang WJ, Tian GD, Zhang TZ et al (2020) Scheme selection of design for disassembly (DFD) based on sustainability: a novel hybrid of interval 2-tuple linguistic intuitionistic fuzzy numbers and regret theory. *J Clean Prod* 281:124724. https://doi.org/10.1016/j.jclepro.2020.124724

Wang KP, Li XY, Gao L, Li PG, Sutherland JW (2021) A discrete artificial bee colony algorithm for multiobjective disassembly line balancing of end-of-life products. *IEEE Transactions on Cybernetics* 1–12. https://doi.org/10.1109/TCYB.2020.3042896

Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. *IEEE Trans Evol Comput* 1(1):67–82. https://doi.org/10.1109/4235.585893

Wu P, Wang H, Li B, Fu W, Ren J, He Q (2022b) Disassembly sequence planning and application using simplified discrete gravitational search algorithm for equipment maintenance in hydropower station. *Expert Syst Appl 208*:118046

Wu T, Zhang Z, Yin T, Zhang Y (2022a) Multi-objective optimisation for cell-level disassembly of waste power battery modules in human-machine hybrid mode. Waste Manag 144:513–526

Yang DY, Xu ZG, Zhu JF, Su KY, Liu WM (2019a) Objective selective disassembly sequence planning considered product fault features. Journal of Harbin Institute of Technology 51(7):160–170. https://doi.org/10.11918/j.issn.0367-6234.201807074

Yang YS, Yuan G, Zhuang QW, Tian GD (2019b) Multi-objective low-carbon disassembly line balancing for agricultural machinery using MDFOA and fuzzy AHP. J Clean Prod 233:1465–1474. https://doi.org/10.1016/j.jclepro.2019.06.035

Yin T, Zhang Z, Zhang Y, Wu T, Liang W (2022) Mixed-integer programming model and hybrid driving algorithm for multi-product partial disassembly line balancing problem with multi-robot workstations. *Robot Comput Integr Manuf 73*:102251

Yuan G, Yang YS, Tian GD, Zhuang QW (2020) Comprehensive evaluation of disassembly performance based on the ultimate cross-efficiency and extension-gray correlation degree. *J Clean Prod* 245:118800. https://doi.org/10.1016/j.jclepro.2019.118800

Zeng Y, Zhang Z, Yin T, Zheng H (2022) Robotic disassembly line balancing and sequencing problem considering energy-saving and high-profit for waste household appliances. J Clean Prod 381:135209

Zhang XF, Wei G, Wang L (2015) Parallel disassembly sequence planning for complex products based on genetic algorithm. *Journal of Computer Aided Design & Computer Graphics* 27(7):1327–1333. https://doi.org/10.3969/j.issn.1003-9775.2015.07.024

Zhang B, Pan QK, Gao L, Zhang XL, Sang HY, Li JQ (2017) An effective modified migrating birds optimization for hybrid flowshop scheduling problem with lot streaming. *Appl Soft Comput* 52:14–27. https://doi.org/10.1016/j.asoc.2016.12.021

Zhang C, Fathollahi-Fard AM, Li J, Tian G, Zhang T (2021) Disassembly sequence planning for intelligent manufacturing using social engineering optimizer. Symmetry 13(4):663

Zhang Y, Zhang Z, Yin T, Liang W (2023) Mathematical formulation and an improved moth–flame optimization algorithm for parallel two-sided disassembly line balancing based on fixed common stations. Journal of Computational Design and Engineering 10(1):233–249

Zheng FF, He JK, Chu F, Liu M (2018) A new distribution-free model for disassembly line balancing problem with stochastic task processing times. *Int J Prod Res* 56(24):7341–7353. https://doi.org/10.1080/00207543.2018.1430909

Zhou B, Bian J (2022) Multi-mechanism-based modified bi-objective Harris Hawks optimization for sustainable robotic disassembly line balancing problems. *Eng Appl Artif Intell 116*:105479

Zhu ZW, Zhou XH (2021) A multi-objective multi-micro-swarm leadership hierarchy-based optimizer for uncertain flexible job shop scheduling problem with job precedence constraints. *Expert Syst Appl* 182(November):115214. https://doi.org/10.1016/j.eswa.2021.115214

Zhu XT, Zhang ZQ, Zhu XM (2014) An ant colony optimization algorithm for multi-objective disassembly line balancing problem. *China Mechanical Engineering* 25(8):1075–1079. https://doi.org/10.3969/j.issn.1004-132X.2014.08.016

Zhu L, Zhang Z, Wang Y (2018) A Pareto firefly algorithm for multi-objective disassembly line balancing problems with hazard evaluation. *Int J Prod Res 56*(24):7354–7374

Zitzler E, Thiele L, Laumanns M, Fonseca CM, Da Fonseca VG (2003) Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Trans Evol Comput 7*(2):117–132